

---

**Área Informática**

---

# Sistemas IV

---

Material elaborado por  
**Mgtr. Ing. Cecilia Savi**



Sistemas IV

2012

---

**INSTITUCION CERVANTES**





## Fundamentos de la materia

Dentro de la estructura del plan de estudios de la carrera de Analista de Sistemas de Computación, la materia se encuentra en el cuarto cuatrimestre, que corresponde al segundo año de estudios.

En esta asignatura, usted aprenderá a realizar actividades que le permitan dimensionar, analizar y diseñar sistemas de información en el contexto de las organizaciones o en casos de estudio. Acciones inherentes a la resolución de problemas a través del estudio y desarrollo de sistemas de información eficientes en áreas de la organización; además de proponer alternativas de solución a las necesidades y problemas detectados.

Para ello, el módulo proporciona herramientas de análisis, diseño, implementación y prueba, en el ámbito de los sistemas de información y una unidad práctica con actividades que van a potenciar las habilidades fundamentales que se requieren para diseñar sistemas de información de los negocios actuales.

### Organización del Módulo

- » Objetivos Generales
- » Contenidos
- » Mapa conceptual
- » Propuesta Metodológica
- » Formas de evaluación
- » Material Soporte de Información de todas las unidades
- » Evaluación Diagnóstica
- » Unidad 1: Introducción al Análisis
- » Unidad 2: Introducción al Diseño
- » Unidad 3: Cómo construimos los modelos
- » Unidad 4: Implementación
- » Unidad 5: Prueba
- » Anexo A: Integramos los Flujos de trabajo del PDUS.
- » Modelo de examen
- » Índice
- » Guía de actividades

## Objetivos Generales

---

- ▶ Reconocer el Proceso Unificado de Desarrollo de Software e identificar los flujos de trabajo, fases, artefactos o modelos básicos.
- ▶ Internalizar el significado de los procesos del Análisis, diseño, implementación y prueba del PDUS.
- ▶ Seleccionar los dispositivos de entrada salida más efectivo para el proyecto.
- ▶ Aplicar soluciones concretas, en los sectores manuales como automáticos del procesamiento de la información.
- ▶ Aplicar herramientas y propiedades del Análisis y Diseño.
- ▶ Identificar la mejor plataforma para la instalación.
- ▶ Interpretar los modelos de sistemas ya construidos, y los nuevos, como así también el balanceamiento de estos, en cada flujo de trabajo y en las fases.
- ▶ Aplicar los conceptos teóricos a ejemplos concretos.

## Contenidos

---

### **Unidad I Introducción al Análisis**

¿Qué es el análisis? - Esencia del Análisis. ¿Por qué hay que hacer Análisis? ¿Qué significa hacer Análisis con objetos?. Principales actividades del Proceso de Análisis. Organizar los objetos. Describir como interactúan los objetos. Especificar las operaciones de los objetos sin consideraciones de implementación. Definir la estructura interna de los objetos. Reusabilidad en el análisis. No debemos Confundir Análisis con Diseño.

El análisis en el proceso unificado de software, en pocas palabras. Descripción del análisis en términos estáticos. Artefactos del modelo de análisis.

Clases del análisis. Realización de casos de uso análisis. Paquetes de análisis. Descripción de la arquitectura.

Descripción del análisis en términos Dinámicos. Flujo de trabajo. Análisis de la arquitectura. Análisis de un caso de uso - Identificar clases de análisis. El papel del análisis en el ciclo de vida del software.

### **Unidad II Introducción al diseño**

¿Qué hacemos en el diseño?. Comparación entre el modelo de análisis y el modelo de diseño. Papel del Diseño en el Ciclo de vida Proceso Unificado. Diseño en el ciclo de vida (Visión Estática). Modelo de diseño (descripción de la arquitectura). Clases de diseño. Casos uso de diseño – Diagrama de transición de estados. Diagrama de clases. Diagrama interacción. Subsistema de diseño. Interfaz. Modelo de Despliegue (descripción de la arquitectura). Diagrama de despliegue. Ambiente Implementación

Diseño en el ciclo de vida (Visión Dinámica). Diseño arquitectura. Diseño de casos de uso. Diseño de clases. Diseño de subsistemas.

**Unidad III Cómo construimos los modelos**

Recordamos ¿Qué es entonces un modelo?. Recordamos Diagramas.

Caso Práctico. Enunciado

Modelado del Sistema de Negocio

Modelo de casos de uso del Sistema de Negocio (diagramas CU)

Descripciones de CU

Modelado del Sistema de Información

Modelo de casos de uso del Sistema de Información (diagramas de CU)

Descripciones de algunos de los casos de uso esenciales.

Listado de casos de uso.

Modelo de Objetos del Dominio

Diagrama de Clases

Modelo de Análisis

Realización de Análisis de los Casos de Uso (Diagramas de colaboración)

Refinamiento del Diagrama de Clase.

Modelo de Diseño

Diagrama de Transición de Estados.

Mapeo de Diagrama de Clase a Modelo Relacional.

Modelo de Despliegue.

Diagrama de Interacción (secuencia)

Interfaz

Subsistemas

**Unidad IV Implementación**

Papel de la **Implementación** en el Ciclo de vida Proceso Unificado. Modelo de Implementación. Subsistemas de Implementación. Descripción de la arquitectura – vista del Modelo de Implementación. Plan de integración de construcción – Planificación. Pruebas de Unidad

**Unidad V Prueba**

Papel de la **Prueba** en el Ciclo de vida Proceso Unificado. Modelo de Prueba. Casos de Prueba. Procedimientos de Prueba. Plan de Prueba. Realizar Pruebas. Fases de Evaluación. Tipos de test. Planificación de la evaluación.

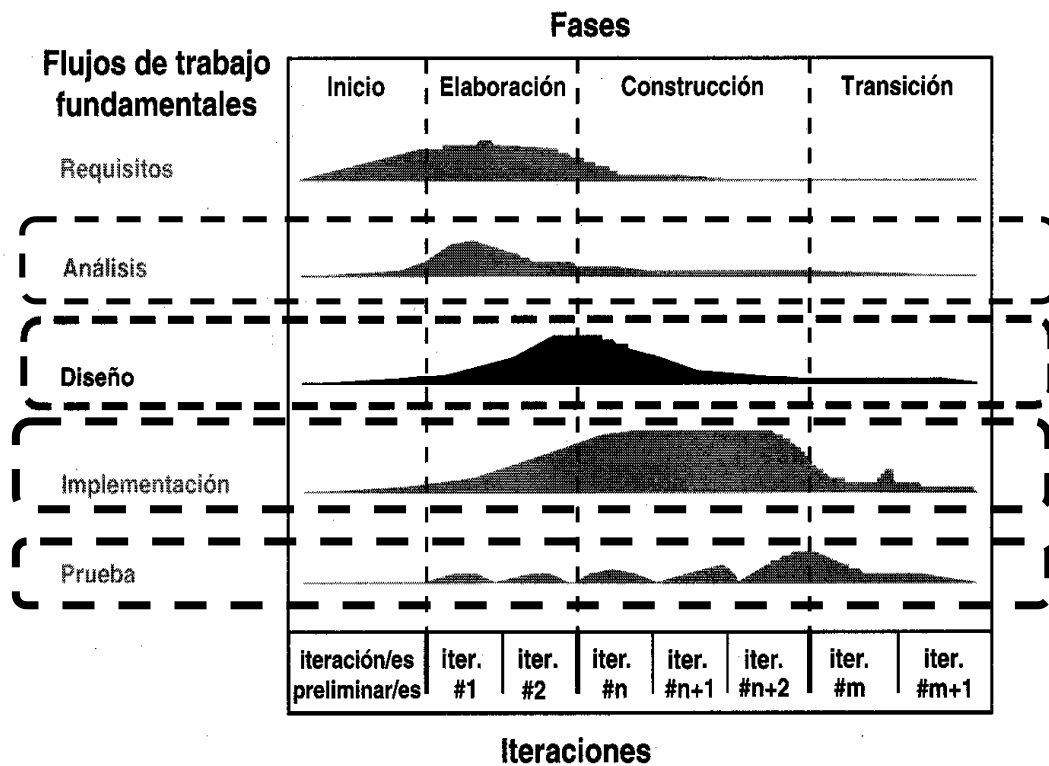
**Anexo A Integramos los Flujos de trabajo del PDUS.**

Visión integral del PDUS: Fases y flujos de trabajo

## Mapa Conceptual

### Proceso Unificado de Desarrollo de Software

#### Ciclo de Vida



## Propuesta Metodológica

En este apartado queremos plantear la organización de los temas del módulo, y recomendaciones sobre la forma de realizar su lectura.

El módulo se organiza en su primer capítulo realizando un repaso del flujo de trabajo fundamental: “Captura de requisitos” y se lo combina de un modo ajustado a los nuevos flujos “ANÁLISIS” (en la misma sección) y de “DISEÑO” (capítulo dos).

Cabe aclarar que se ha respetado la misma línea de trabajo que se desarrolló en el módulo de Sistemas III, ya que se considera que este es un complemento de aquel, ambos responden a una parte del Ciclo de vida del Proceso Unificado de desarrollo de software. Es decir se sintetizan los conceptos más significativos con su correspondiente ejemplificación, a través de una visión ESTÁTICA y luego DINÁMICA del Proceso.

En la tercera unidad se construyen modelos y diagramas, en un caso práctico. Esto permitirá alcanzar una visión integradora de lo que es desarrollar cualquier proyecto de sistemas.

Con respecto al ítem autoevaluación que se presenta al final de la unidad I , II y III los casos que se presentan fueron trabajados parcialmente en Sistemas III (los diagramas y modelos correspondientes a los Modelos de Negocios y de Captura de Requisitos ), la idea es completarlos con las nuevos diagramas aprendidos (pertenecientes al Análisis y al Diseño), y obtener así, dos casos MÁS, resueltos completamente, para que le sirvan al alumno de referencia, al momento de emprender el desarrollo de un sistema de Información.

El capítulo IV se basa en la implementación de un proyecto, en ella se destaca el vínculo de esta etapa, con la fase diseño, la distribución de la propuesta a componentes de hardware, y la capacidad de probar los componentes y enlazarlos para llevar a cabo la comprobación e implementación de un sistema.

Por último, en el apartado V se verifican los resultados de la implementación, probando cada construcción, así como versiones finales a ser entregadas. Se destaca también la planificación de las pruebas y sus distintos tipos.

En lo que llamamos Anexo se pretende dar una visión general, abarcativa del PDUS y de la relación flujos y fases, destacando las ventajas y desventajas de este proceso, los riesgos y los criterios de evaluación que deben considerarse.

## Formas de evaluación

---

Según la modalidad de regularización definida por la materia Sistemas IV, usted deberá rendir exámenes parciales escritos (cualquiera sea la modalidad) de manera presencial o domiciliaria. En estas instancias se evalúan contenidos teóricos y prácticos.

Las actividades de autoevaluación, se desarrollan al cierre de cada unidad. Las correspondientes claves de corrección, son alojadas por el docente tutor, en el área de descargas de autogestión del alumno, en forma oportuna según el avance de cada unidad.

En el examen final, la forma de evaluación es, en primera instancia, un escrito con un caso práctico para resolver, y en segunda instancia un escrito u oral (según lo determine el tribunal), sobre aspectos teóricos complementarios o que fundamenten lo realizado en la actividad práctica.

## Material soporte de información de todas las unidades

---

### **Obligatoria**

- ▶ JACOBSON, Ivar, Grady BOOCH, James RUMBAUGH, *El proceso Unificado de Desarrollo de Software*, Edición 1999 o superior.
- ▶ BOOCH, Grady, James RUMBAUGH, Ivar JACOBSON, *El lenguaje Unificado de Modelado*, Edición 2000 o superior
- ▶ RUMBAUGH, James, Ivar JACOBSON, Grady BOOCH, *El Lenguaje Unificado de Modelado. Manual de Referencia* Edición 2000 o superior

### **Sugerida**

- ▶ SCHMULLER Joseph, *Aprendiendo UML en 24 hs*
- ▶ MARTIN, James, James J. Odell, *Análisis y Diseño Orientado a Objetos*



## Evaluación Diagnóstica

1. Enuncie algún beneficio de trabajar con las tecnologías orientadas a objetos.
2. Elabore un esquema, en el que se identifiquen los flujos de trabajo y las fases que recuerde, en el ciclo de vida del PDUS.
3. Defina con sus palabras, las diferencias que se establecen entre un diagrama de casos de uso del negocio y el diagrama de casos de uso del sistema de información.
4. Dé un ejemplo de una clase, un objeto, un atributo, un método. Desarrolle un concepto de cada uno de ellos.
5. ¿Qué relación encuentra al momento de diseñar un prototipo de interfaz, con el diagrama de clases elaborado en la etapa del análisis?
6. ¿Qué herramienta o artefactos son indispensables construir para determinar los requisitos funcionales de un proyecto informático?
7. Mencione y explique algún concepto que haya sido significativo para usted en la etapa anterior y que considere relevante, al momento de continuar con el desarrollo de un proyecto informático.
8. Elabore un esquema o gráfico que sintetice lo aprendido en Sistemas III

## Iconos de Referencia



**Recomendación**



**Lectura en bibliografía**



**Conceptos para recordar**



# ► Unidad I

## Introducción al Análisis

### Objetivos

- Comprender de modo mas preciso la Captura de Requisitos.
- Comprender el significado de las fases e iteraciones del análisis.
- Identificar las herramientas y conocer su funcionalidad en el modelo
- Ejercitar los conocimientos adquiridos en la unidad anterior.

### Temas de la Unidad

Introducción

Recordamos: Ciclo de vida de PDUS. Fases dentro del ciclo, Iteraciones, Ciclo de desarrollo, Flujos de trabajo del proceso, Modelos. Recordamos: UML

¿Qué es el análisis? - Esencia del Análisis

¿Por qué hay que hacer Análisis?

¿Qué significa hacer Análisis con objetos?

Principales actividades del Proceso de Análisis

- Encontrar los objetos y sus responsabilidades

- Organizar los objetos

- Especificar las operaciones de los objetos sin consideraciones de implementación

- Definir la estructura interna de los objetos

Reusabilidad en el análisis

No debemos Confundir Análisis con Diseño.

El análisis en el proceso Unificado de software, en pocas palabras

Descripción del análisis en términos estáticos

- Artefactos del Modelo de análisis

- Clases Análisis

- Paquetes de Análisis

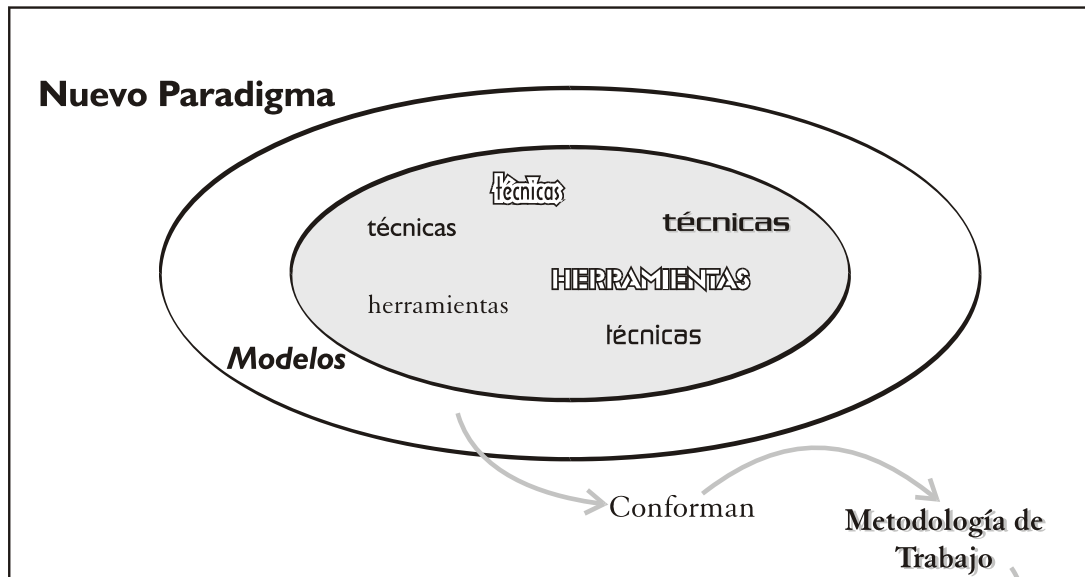
Descripción del análisis en términos Dinámicos

- Análisis de un caso de uso - Identificar clases de análisis

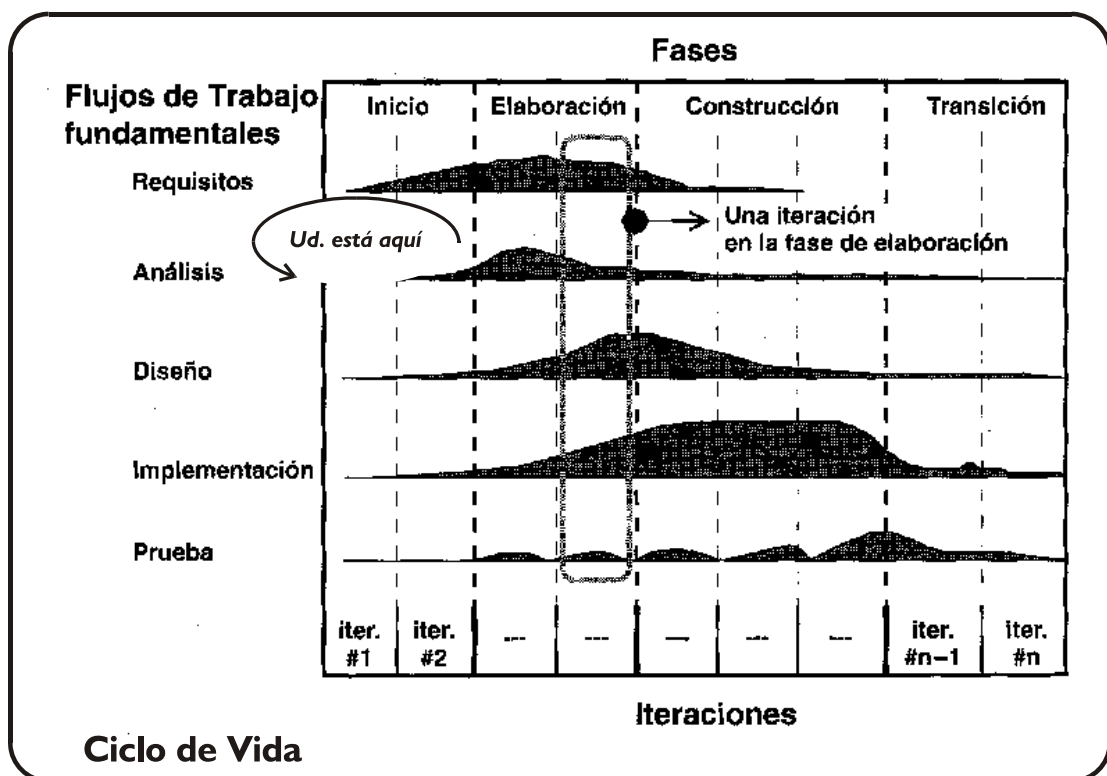
El papel del análisis en el ciclo de vida del software

Autoevaluación

## Mapa Conceptual



## Proceso Unificado de Desarrollo de Software



## Introducción

Durante el análisis, examinaremos los requisitos que se describieron en la captura de requisitos, refinándolos y estructurándolos. El objetivo de hacerlo, es conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea fácil de mantener y que nos ayude a estructurar el sistema entero incluyendo su arquitectura.

En consecuencia, podremos razonar más sobre los aspectos internos del sistema, y por tanto resolver aspectos relativos a la interferencia de casos de uso. También lograremos utilizar un lenguaje más formal para apuntar detalles relativos a los requisitos del sistema. Llamaremos a esto "refinar los requisitos".

Es decir, en esta unidad, se presentará una explicación más detallada de qué entendemos por análisis. Una breve "puesta en situación" conceptual del análisis y después la descripción del papel del análisis en las diferentes fases del ciclo de vida del software.

## RECORDAMOS: Ciclo de Vida del Proceso Desarrollo Unificado

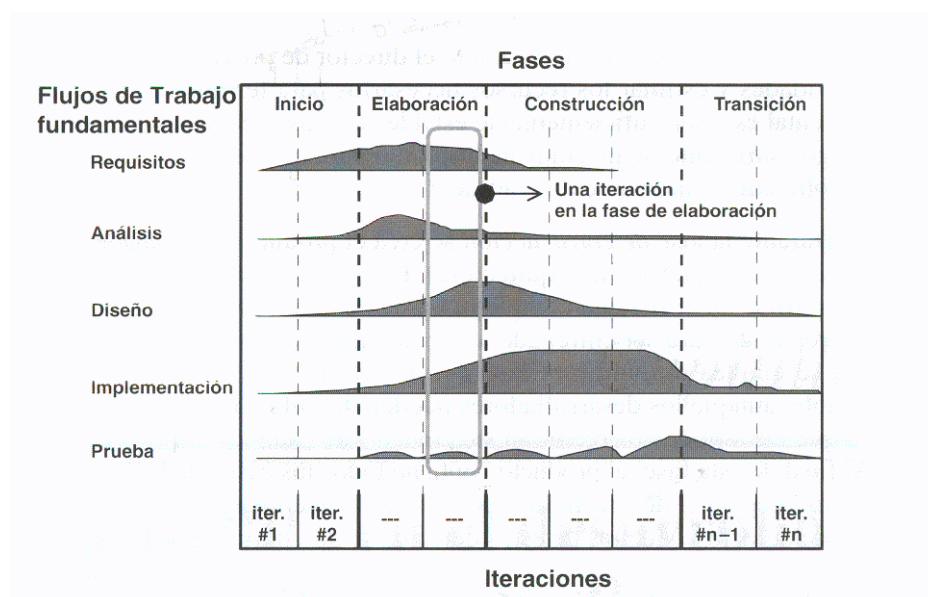


Un proceso es un conjunto de pasos ordenados parcialmente para alcanzar un objetivo. En la ingeniería del software, el objetivo es entregar un producto software que satisfaga las necesidades del usuario, de forma eficiente y predecible.

El objetivo del Proceso Unificado es permitir la producción de un software de la mayor calidad, que satisfaga las necesidades de los usuarios finales, dentro de planificaciones y presupuestos predecibles.

El Proceso unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema.

Cada ciclo consta de cuatro fases: inicio, elaboración, construcción y transición. Cada fase se subdivide a su vez en iteraciones.



### Fases dentro de un ciclo

Cada ciclo se desarrolla a lo largo del tiempo. Este tiempo, a su vez, se divide en cuatro fases. A través de una secuencia de modelos se visualiza lo que está sucediendo en esas fases. Dentro de cada fase, los desarrolladores pueden descomponer adicionalmente el trabajo en iteraciones con sus incrementos resultantes.

La Figura anterior muestra en la columna izquierda los flujos de trabajo: requisitos, análisis, diseño, implementación, y prueba. Las curvas son una aproximación, de hasta dónde se llevan a cabo los flujos de trabajo en cada fase. Recuerden que cada fase se divide normalmente en iteraciones o miniproyectos.

Una iteración típica pasa por los cinco flujos de trabajo, como se muestra en la Figura.



Una fase es el intervalo de tiempo entre dos hitos importantes del proceso durante la cual se cumple un conjunto bien definido de objetivos, se completan artefactos y se toman decisiones sobre si pasar a la siguiente fase.

El proceso Unificado consta de cuatro fases:

1. Iniciación: Establece la planificación del proyecto.
2. Elaboración: Establece un plan para el proyecto y una arquitectura correcta.
3. Construcción: Desarrollo del sistema.
4. Transición: Proporciona el sistema a sus usuarios finales.

Si resumiéramos en pocas palabras el objetivo de cada **fase**, podríamos expresar:

- Durante **la fase de inicio**, se desarrolla una descripción del producto final a partir de una buena idea y se presenta el análisis de negocio para el producto.

Durante la fase de iniciación se establece la planificación del proyecto y se delimita su alcance. La planificación del proyecto incluye los criterios de éxito, la evaluación del riesgo, estimaciones de recursos que se necesitarán y un plan de fases que muestre la planificación de los hitos principales. Durante la iniciación, es frecuente crear un prototipo ejecutable que sirva para probar los conceptos. Al final de la fase de inicio se examinan los objetivos del ciclo de vida del proyecto y se decide si proceder con el desarrollo del sistema.

- Durante **la fase de elaboración** se especifican en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura del sistema. La relación entre la arquitectura del sistema y el propio sistema es primordial.

Los objetivos de la fase de elaboración son analizar el dominio del problema, establecer una base arquitectónica sólida, desarrollar el plan del proyecto y eliminar los elementos de más alto riesgo del proyecto. Las decisiones arquitectónicas deben tomarse con una comprensión del sistema global. Esto implica que se deben describir la mayoría de los requisitos del sistema. Para verificar la arquitectura, se implementa un sistema que demuestre las distintas posibilidades de la arquitectura y ejecute los casos de uso significativos.

Las fases de iniciación y elaboración incluyen las actividades de análisis, diseño del ciclo de vida del desarrollo; la construcción y la transición constituyen su producción.

- Durante **la fase de construcción** se crea el producto. En esta fase, la línea base de la arquitectura crece hasta convertirse en el sistema completo. La descripción evoluciona hasta convertirse en un producto preparado para ser entregado a la comunidad de usuarios.

El grueso de los recursos requeridos se emplea durante esta fase del desarrollo. Sin embargo, la arquitectura del sistema es estable, aunque los desarrolladores pueden descubrir formas mejores de estructurar el sistema, ya que los arquitectos recibirán sugerencias de cambios arquitectónicos de menor importancia.

Durante la fase de construcción se desarrolla, de forma iterativa e incremental, un producto completo que está preparado para la transición hacia la comunidad de usuarios. Esto implica describir los requisitos restantes y los criterios de aceptación, refinando el diseño y completando la implementación y las pruebas del software. Al final de la fase de construcción se decide si el software, los lugares donde se instalará y los usuarios están todos preparados para empezar a funcionar.

- La **fase de transición** cubre el período durante el cual el producto se convierte en versión beta. En la versión beta un número reducido de usuarios con experiencia prueba el producto e informa de defectos y deficiencias.

Los desarrolladores corrigen los problemas e incorporan algunas de las mejoras sugeridas en una versión general dirigida a la totalidad de la comunidad de usuarios. La fase de transición conlleva actividades como la fabricación, formación del cliente, el proporcionar una línea de ayuda y asistencia, y la corrección de los defectos que se encuentren tras la entrega.

Durante la fase de transición el software se despliega en la comunidad de usuarios. Una vez que el sistema ha sido puesto en manos de los usuarios finales, a menudo aparecen cuestiones que requieren un desarrollo adicional para ajustar el sistema, corregir algunos problemas no detectados o finalizar algunas características que habían sido pospuestas. Esta fase comienza normalmente con una versión beta del sistema, que luego será reemplazada con el sistema de producción.

Al final de la fase de transición se decide si se han satisfecho los objetivos del ciclo de vida del proyecto, y se determina si se debería empezar otro ciclo de desarrollo. Este es también un punto en el que se asimilan las lecciones aprendidas en el proyecto para mejorar el proceso de desarrollo, que será aplicado al próximo proyecto.

---

## ***Iteraciones***



Una iteración es un ciclo completo de desarrollo que produce una versión (interna o externa) de un producto ejecutable, además de constituir un subconjunto del producto final en desarrollo, que luego se irá incrementando de iteración en iteración hasta convertirse en el sistema final.

---

Cada iteración pasa a través de varios flujos de trabajo del proceso, aunque con un énfasis diferente en cada uno de ellos, dependiendo de la fase en que se encuentre. Durante la iniciación, el interés se orienta hacia el análisis y el diseño. Durante la construcción, la actividad central es la implementación, y la transición se centra en el despliegue.

---

## ***Ciclos de desarrollo***

El paso a través de las cuatro fases principales constituye un ciclo de desarrollo, y produce una generación de software. La primera pasada a través de las cuatro fases se denomina ciclo de desarrollo inicial. A menos que acabe la vida del producto, un producto existente evolucionará a la siguiente generación repitiendo la misma secuencia de inicio, elaboración, construcción y transición. Esta es la evolución del sistema, así que los ciclos de desarrollo después del ciclo inicial son los ciclos de evolución.

---

## ***Flujos de trabajo del proceso***

El Proceso Unificado consta de nueve flujos de trabajo.

- |                             |  |
|-----------------------------|--|
| <b>1. Mod. del negocio</b>  | Describe la estructura y la dinámica de la Organización.                           |
| <b>2. Requisitos</b>        | Describe el método basado en casos de uso para extraer los requisitos.             |
| <b>3. Análisis y diseño</b> | Describe las diferentes vistas arquitectónicas                                     |
| <b>4. Implementación</b>    | Tiene en cuenta el desarrollo de software, la prueba de unidades y la integración. |



- |                                    |  |
|------------------------------------|--|
| <b>5. Pruebas</b>                  | Describe los casos de pruebas, los procedimientos y las métricas para evaluación de defectos |
| <b>6. Despliegue</b>               | Cubre la configuración del sistema entregable  |
| <b>7. Gestión de configuración</b> | Controla los cambios y mantiene la integridad de los artefactos de un proyecto.              |
| <b>8. Gestión del Proyecto</b>     | Describe varias estrategias de trabajo en un proceso iterativo.                              |
| <b>9. Entorno</b>                  | Cubre la infraestructura necesaria para desarrollar un sistema.                              |

Dentro de cada flujo de trabajo del proceso hay un conjunto de artefactos y actividades relacionados.



Un *artefacto* es algún documento, informe o ejecutable que produce, se manipula o se consume.



Una *actividad* describe las tareas (pasos de concepción, realización y revisión) que llevan a cabo los trabajadores para crear o modificar artefactos, junto con las técnicas y guías para ejecutar las tareas, incluyendo quizá herramientas para ayudar a automatizar algunas de ellas.

## Modelos



Los modelos son el tipo de artefacto más importante en el Proceso Unificado. Un modelo es una simplificación de la realidad, creada para comprender mejor el sistema que se está creando.

En el Proceso hay nueve modelos que en conjunto cubren todas las decisiones importantes implicadas en la visualización, especificación, construcción y documentación de un sistema con gran cantidad de software.

- 1. Modelo del negocio:** Establece una abstracción de la organización
- 2. Modelo del dominio:** Establece el contexto del sistema.
- 3. Modelo de casos de uso:** Establece los requisitos funcionales del sistema.
- 4. Modelo de análisis (opcional):** Establece un diseño de las ideas.
- 5. Modelo de diseño:** Establece el vocabulario del problema solución.
- 6. Modelo de despliegue:** Establece la topología hardware sobre la cual se ejecutará el sistema.
- 7. Modelo de implementación** Establece las partes que se utilizarán para ensamblar y hacer disponible el sistema físico
- 8. Modelo de pruebas** Establece las formas de validar y verificar sistema.

Una vista es una proyección de un modelo. En el Proceso Unificado la arquitectura de un sistema se captura en forma de cinco vistas que interactúan entre sí, vista de diseño, la vista de procesos, la vista de despliegue, la vista de implementación y la vista de casos de uso.

## RECORDAMOS: UML

---



El UML (lenguaje Unificado de Modelado) es una herramienta, que permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas

---

Nos ofrece un modo estándar de visualizar, especificar, construir, documentar y comunicar los artefactos de un sistema muy basado en el software, un lenguaje como éste, debe utilizarse en el contexto de un proceso de software completo.



UML es un medio, y no un fin. El objetivo final es una aplicación software robusta, flexible y escalable.

---

El UML está compuesto de diversos diagramas gráficos, que se combinan para conformar diagramas. Debido a que UML es un lenguaje, cuenta con reglas para combinar tales elementos. La finalidad de los diagramas es presentar diversas perspectivas de un sistema al cual se conoce como modelo. El modelo UML de un sistema describe lo que supuestamente hará el sistema, pero no dice como implementar dicho sistema.



*Recuerda que en la plataforma, están a tu disposición los módulos de trabajo, que te permitirán repasar todos estos diagramas mencionados, y así poder enfrentar los nuevos flujos de trabajo de esta etapa.*

## Qué es el análisis?



*Análisis es el estudio y especificación de un determinado dominio de problema, dentro del contexto de objetivos y metas de negocio preestablecidas, y sin consideraciones de implementación.*

*Es mucho más sencillo de interpretar, si desmenuzamos cada párrafo de esta definición:*

- ▶ Todos podemos estar de acuerdo en que Análisis es investigar y especificar un problema. El análisis consiste básicamente en mirar una determinada realidad y especificarla, es decir, observarla y plasmarla en un modelo
- ▶ Lo fundamental es hacer esto, bajo el contexto de ciertas metas y objetivos de negocio que están preestablecidas. Haremos Análisis dentro de una organización o para una organización y por lo tanto ese análisis tiene que responder a problemas de dicha organización y esos problemas se deben básicamente a que una empresa tiene objetivos y metas claras. A partir de esas metas y objetivos se plantean los problemas y a partir de esos problemas es que se necesita tener un sistema de información automatizado que le permita llevar adelante el negocio.
- ▶ Por lo tanto cuando hacemos análisis no se debe perder de vista la visión de objetivos y metas, que vienen de la alta gerencia de la organización. En caso contrario, nuestro análisis no estará respondiendo a la problemática de la empresa, sino a una problemática aleatoria o a los que se nos pudiera llegar a ocurrir, y esa no es la idea.

Los sistemas son para la organización y por lo tanto deben estar de acuerdo con sus objetivos y metas.

## Esencia del Análisis

Hay tres puntos a tener en cuenta:

- ▶ El análisis es un proceso de observación y comprensión. Se observa un dominio de problema para poder comprenderlo y especificarlo
- ▶ El análisis no es un proceso creativo. Se debe plasmar una realidad tal cual es, adoptando una posición neutra, a lo sumo se pueden proponer cambios interactuando con la gente que puede producirlos.
- ▶ El resultado del análisis debe ser único.

## ¿Por qué hay que hacer Análisis?

Hay tres razones para hacer análisis

- » Hacer Análisis es comprender el problema, así que aunque más no sea desde ese punto de vista, se debe hacer Análisis para entender cual problema es el problema a solucionar.
- » Se tiene que poder especificar ese problema en términos de sistemas de información, ya que el objetivo es construir sistemas que funcionen en computadoras.
- » Se necesita tener algún medio de comunicación estándar entre los distintos roles que participan en el desarrollo del proyecto, que permita que entre ellos se entiendan. Para que todo esto resulte, el modelo de análisis debe ser simple. No se puede plantear una especificación tan árida y compleja que una persona que no tenga experiencia en construcción de software, no lo pueda entender, y por lo tanto no lo pueda validar.

*En cualquier paradigma se debe hacer Análisis. Probablemente hacer esto con objetos sea algo muy distinto a lo que se viene haciendo con técnicas estructuradas.*

## ¿Qué significa hacer Análisis con objetos?



Es expresar los resultados en términos de objetos cooperantes, cada uno de ellos con un conjunto bien definido de comportamientos.

El problema principal del Análisis Orientado a Objetos es precisamente encontrar los objetos. Esto no significa solamente identificarlos, sino también asignarles responsabilidades.

Existen diferentes modelos que coinciden en que se va a contener en un modelo de análisis. La diferencia está en cómo lograrlo, en cómo se encuentran los objetos y como se encara el desarrollo de las aplicaciones a partir del modelo obtenido.

### Unos dicen que

Para encontrar los objetos, la propuesta es buscar objetos tangibles a través de sustantivos y/o categorías, en una especificación de requerimientos.

- ▶ Aquellos sustantivos que parecen de peso, podrían llegar a ser los objetos de la aplicación.
- ▶ En cuanto a las categorías, se trata de una técnica que proviene de la modelización de datos y consiste en agrupar objetos en categorías que representen, por ejemplo,

lugares, roles, personas, etc. En, sí mismas, esas categorías no interesan, son solo ayuda para descubrir los objetos, son solo una herramienta intelectual.

El problema fundamental al hacer Análisis con objetos es precisamente identificar los objetos y sus responsabilidades. Estos objetos a encontrar son los constructores básicos del sistema, y siempre es difícil encontrar aquellas abstracciones que representan a los constructores básicos, a partir de los cuales se forma todo el resto. No se trata de un problema que surja por el cambio de paradigma, ocurre porque en cualquier ámbito lo difícil es, encontrar lo fundamental.

## Principales actividades del Proceso de Análisis

En el Caso de objetos, estas actividades en general son:

- I. Encontrar los objetos y sus responsabilidad
- II. Organizar los objetos:
- III. Especificar las operaciones de los objetos sin consideraciones de implementación
- IV. Definir la estructura interna de los objetos

### I Encontrar los objetos

Esta es la actividad básica del Proceso de Análisis.

Esos objetos esenciales serán los que van a tener que pertenecer al modelo independientemente de cualquier implementación que pueda elegirse a posteriori.

El factor decisivo para el éxito o fracaso del desarrollo con objetos es tener en mente que: para identificar los objetos, es fundamental concentrarse en los comportamientos esperados del sistema.

### II Organizar los objetos

Una vez identificados los objetos (aunque no necesariamente todos), los mismos se organizan. Esto significa establecer su vinculación mutua y que tipo de relaciones hay.

Hasta ahora se ha utilizado la palabra *objeto* como concepto general. Aquí quizás habría que diferenciar entre *objeto* y *clase*. Pero eso depende de cada método algunos hacen esa diferencia y otros no.

El paradigma establece que cada objeto debe ser instancia de alguna clase. Además, también se establece que las clases son objetos y como tales tienen un comportamiento determinado, que como mínimo debe incluir el crear instancias de si misma. Esto implica que de las clases vamos a tener una vista dual. Por un lado son clases (en el sentido de ser un molde, una especificación de software, de un conjunto de objetos), y por otro, son objetos (en el sentido de tener existencias y comportamientos).

Por lo tanto cuando hagamos un modelo, en algún momento habrá que especificar cuando estamos hablando de objetos (de instancias), cuando de clases y cuando miramos las clases como objetos.

Uno de los momentos en los que tiene que hacerse esa diferencia, es cuando se plantea el tema de las jerarquías. Cuando los objetos se organizan en jerarquías de herencia, estamos hablando de clases, ya que dicha relación se da exclusivamente entre clases.

A nivel de Análisis, especificaremos una relación de herencia entre dos clases si esa relación existe en el mundo real. Esto se dará cuando pueda aplicarse la relación "*es un*", que es totalmente esencial. No es una relación que tenga que ver con implementación.

### ***III Especificar las operaciones de los objetos***

Esto es distribuir las responsabilidades entre los objetos que conforman el sistema. A medida que se va realizando esta actividad se va especificando el protocolo de cada objeto, es decir, el conjunto de mensajes que el objeto puede recibir. Los nombres de esos mensajes forman el vocabulario del modelo, son al modelo de objetos, lo que los atributos al modelo de datos.

Y esto lleva a otro tema: el polimorfismo. Debemos nombrar los mensajes de una manera acorde a la acción del objeto cuando recibe dicho mensaje. Si tenemos objetos que semánticamente hacen la misma cosa, aunque lo hagan de manera diferente, lo razonable es que se use el mismo nombre para el mensaje que activa ese servicio. El polimorfismo debe incorporarse desde el análisis, ya que es una de las herramientas mas poderosas del paradigma, es la que permite generalidad, extensibilidad, etc.

### ***IV Definir la estructura interna de los objetos***

Lo que se analiza es que es "lo que sucede dentro del objeto" para que pueda funcionar. De un objeto se definen comportamientos, métodos, atributos y datos. Por lo tanto para poder especificar un objeto hay que definir como es ese comportamiento y que estructura de datos puede asociársele.

En relación a los métodos hay que recalcar que se esta hablando de aquellos que tienen que ver con la regla del negocio de la organización.

En definitiva el análisis identifica a las propiedades lógicas y el diseñador decide que hacer con ellas.

## Reusabilidad en el análisis

La reusabilidad puede verse a distintos niveles: reusabilidad de código, de Diseño y de Análisis:

Lo primero que se ha hecho siempre es poder re usar código porque los lenguajes de programación siempre son anteriores a las técnicas de diseño. Esto está bastante resuelto, hay mecanismos de herencia y composición que permiten que el re uso de código (lenguajes orientados a objetos especialmente en aquellos que son puros) sea bastante natural y simple.

También esta, el re uso de Análisis. Lo primero que habrá que hacer es definir que se entiende y que significa el re uso de Análisis, lo cual está aún en discusión. De todas formas, el re uso de análisis no es una herramienta de productividad, como ocurre en Diseño o codificación sino que es una herramienta conceptual para lograr que todos los modelos a construir sean coherentes y válidos.

## No debemos Confundir Análisis con Diseño

Por último, se verán las diferencias significativas de cada una de estas etapas:

Análisis	Diseño
El análisis es el proceso de determinar “que” se necesita hacer, antes de decidir “cómo” debe hacerse. El análisis es el acto del descubrimiento.	El diseño es el proceso de determinar “cuál” de muchas posibles soluciones es la mejor para lograr lo que se necesita hacer, respetando las restricciones tecnológicas y de presupuesto del proyecto. El diseño es un “cómo” específico para aplicarlo a un “que”. El diseño es el arte de compromiso.

## El análisis en el Proceso de Desarrollo Unificado de Software, en pocas palabras

- ▶ El lenguaje que utilizamos en el análisis se basa en un modelo de objetos conceptual, que llamamos *modelo de análisis*. El modelo de análisis nos ayuda a refinar los requisitos y nos permite razonar sobre los aspectos internos del sistema, incluidos sus recursos compartidos internos.
- ▶ El modelo de análisis también nos ayuda a estructurar los requisitos y nos proporciona una estructura centrada en el mantenimiento, en aspectos tales como la flexibilidad ante los cambios y la reutilización. Esta estructura no sólo es útil para el mantenimiento de los requisitos como tales, sino que también se utiliza como entrada en las actividades de diseño y de implementación.
- ▶ El modelo de análisis puede considerarse una primera aproximación al modelo de diseño, aunque es un modelo por sí mismo.

Dijimos en la introducción que durante el análisis, analizaremos los requisitos que se describieron en la captura de requisitos, refinándolos y estructurándolos. El objetivo de hacerlo es conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea fácil de mantener y que nos ayude a estructurar el sistema entero incluyendo su arquitectura.

## Descripción del análisis en términos estáticos

---

### ***I - Artefactos del Modelo de análisis***

#### *II Clases Análisis*

#### *12 Paquetes de Análisis*

#### *13 Descripción de la arquitectura*

### ***II Clases Análisis***

---



Una clase de análisis representa una abstracción de una o varias clases y/o subsistemas del diseño del sistema.

---

Esta abstracción posee las siguientes características:

- ▶ Una clase de análisis se centra en el tratamiento de los requisitos funcionales y pospone los no funcionales, denominándolos requisitos especiales, hasta llegar a las actividades de diseño e implementación subsiguientes.
- ▶ Una clase de análisis define atributos, aunque esos atributos también son de un nivel bastante alto. Normalmente los tipos de esos atributos son conceptuales y reconocibles en el dominio del problema, mientras que los tipos de los atributos en las clases de diseño y la implementación suelen ser tipos de lenguajes de programación. Además, los atributos identificados durante el análisis con frecuencia pasan a ser clases en el diseño y la implementación.
- ▶ Las clases de análisis siempre encajan en uno de los tres estereotipos básicos: de interfaz, de control o de entidad. Cada estereotipo implica una semántica específica (descrita brevemente a continuación), lo cual constituye un método potente y consistente de identificar y de describir las clases de análisis y contribuye a la creación de un modelo de objetos y una arquitectura robustos

Sin embargo, es importante hacer notar aquí que el modelo de análisis hace abstracciones y evita resolver algunos problemas y tratar algunos requisitos que pensamos que es mejor posponer al diseño y a la implementación. La razón por la cual esta "conservación de la estructura" no siempre tiene lugar en la práctica es sencillamente que el diseño debe considerar la plataforma de implementación: lenguaje de programación, sistemas operativos, marcos de trabajo prefabricados, sistemas heredados, y demás. Por economía, puede conseguirse una arquitectura mejor mediante la modificación de la estructura del modelo de análisis durante la transición al modelo de diseño, al ir dando forma al sistema.



## ► Clases de interfaz



Las clases de interfaz se utilizan para modelar la interacción entre el sistema y sus actores (es decir, usuarios y sistemas externos). Esta interacción a menudo implica recibir (y presentar) información y peticiones de (y hacia) los usuarios y los sistemas externos.

Las clases de interfaz modelan las partes del sistema que dependen de sus actores, lo cual implica que clarifican y reúnen los requisitos en los límites del sistema. Por tanto, un cambio en un interfaz de usuario o en un interfaz de comunicaciones queda normalmente aislado en una o más clases de interfaz.

Las clases de interfaz representan a menudo abstracciones de ventanas, formularios, paneles, interfaces de comunicaciones, interfaces de impresoras, sensores, terminales.

Obsérvese que es suficiente con que las clases de interfaz describan lo que se obtiene con la interacción (**es decir, la información y las peticiones** que se intercambian entre el sistema y sus actores). No es necesario que describan cómo se ejecuta físicamente la interacción, ya que esto se considerará en las actividades de diseño e implementación subsiguientes.



Cada clase de interfaz debería asociarse con al menos un actor, y viceversa.

## ► Clases de entidad



Las clases de entidad se utilizan para modelar información que posee una vida larga y que es a menudo persistente. Las clases de entidad modelan la información y el comportamiento asociado de algún fenómeno o concepto, como una persona, un objeto del mundo real, o un suceso del mundo real.

En la mayoría de los casos, las clases de entidad del Análisis se derivan directamente de una clase de entidad del negocio (o de una clase del dominio) correspondiente, tomada del modelo de objetos del negocio (o del modelo del dominio).

*Sin embargo, una diferencia fundamental entre clases de entidad del Análisis y clases de entidad del negocio*

<i>Clases de entidad del análisis</i>	<i>Clases de entidad del negocio</i>
Es que las primeras representan objetos manejados por el sistema en consideración.	Representan objetos presentes en el negocio (y en el dominio del problema) en general.
En consecuencia, las clases de entidad	

<i>Clases de entidad del análisis</i>	<i>Clases de entidad del negocio</i>
reflejan la información de un modo que beneficia a los desarrolladores al diseñar e implementar el sistema, incluyendo su soporte de persistencia.	Las clases de entidad del negocio (o clases del dominio), que por el contrario, describen el contexto del sistema, y por tanto pueden incluir información que el sistema no maneja en absoluto.

Un objeto de entidad no ha de ser necesariamente pasivo y puede tener en ocasiones un comportamiento complejo relativo a la información que representa.

## ► Clases de control



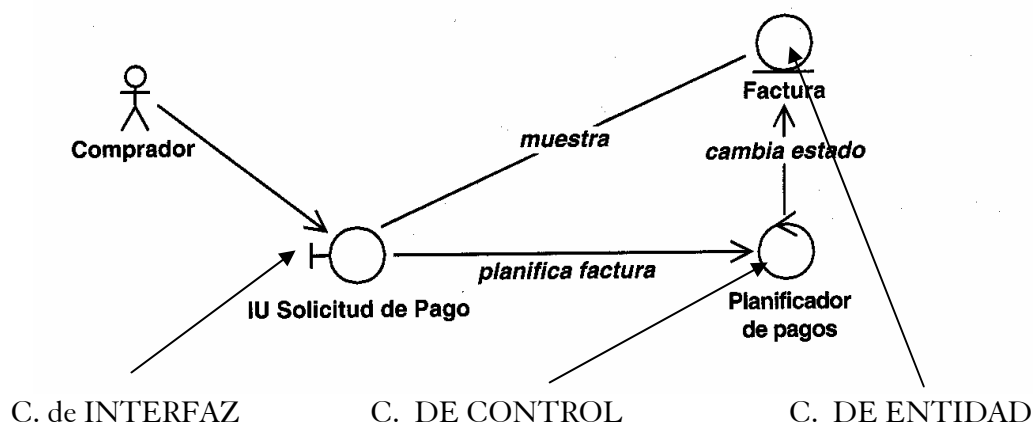
Las clases de control representan coordinación, secuencia, transacciones, y control de otros objetos y se basan con frecuencia para encapsular (unidad VII) el control de un caso de uso en concreto. Las clases de control también se utilizan para representar derivaciones y cálculos complejos, como la lógica del negocio, que no pueden asociarse con ninguna información concreta, de larga duración, almacenada por el sistema (es decir, una clase de entidad concreta).

Los aspectos dinámicos del sistema se modelan con clases de control, debido a que ellas manejan y coordinan las acciones y los flujos de control principales, y delegan trabajo a otros objetos (es decir, objetos de interfaz y de entidad).

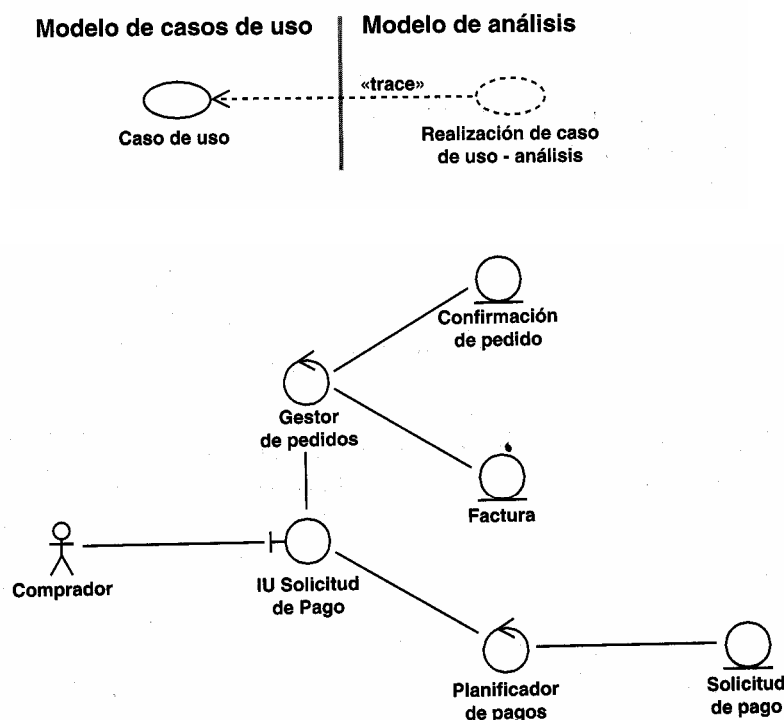
Obsérvese que las clases de control no encapsulan aspectos relacionados con las interacciones, con los actores, ni tampoco aspectos relacionados con información de larga vida y persistente gestionada por el sistema; esto lo encapsulan las clases de interfaz y de entidad, respectivamente.



En cambio, las clases de control encapsulan, y por tanto aíslan, los cambios del control, la coordinación, la secuencia, las transacciones ya veces la lógica del negocio que implica a varios otros objetos.



**Diagramas de clases** Una clase de análisis y sus objetos normalmente participan en varias realizaciones de casos de uso, y algunas de las responsabilidades, atributos, asociaciones de una clase concreta suelen ser sólo relevantes para una única realización de caso de uso. Por lo tanto es importante durante el análisis coordinar todos los requisitos sobre una clase y sus objetos que pueden tener diferentes casos de uso



## 12 Paquete del análisis

Los paquetes del análisis proporcionan un medio para organizar los artefactos del modelo de análisis en piezas manejables.



Un paquete de análisis puede constar de clases de análisis, de realizaciones de casos de uso, y de otros paquetes del análisis (recursivamente).

Los paquetes del análisis deberían ser cohesivos (es decir, sus contenidos deberían estar fuertemente relacionados), y deberían ser débilmente acoplados (es decir, sus dependencias de otros deberían minimizarse).

*Además, los paquetes del análisis tienen las siguientes características:*

- Los paquetes del análisis pueden representar una separación de intereses de análisis. Por ejemplo, en un sistema grande algunos paquetes del análisis pueden analizarse

de manera separada posiblemente de manera concurrente por diferentes desarrolladores con diferente conocimiento del dominio.

- ▶ Los paquetes del análisis deberían crearse basándonos en los requisitos funcionales y en el dominio del problema (es decir, la aplicación o el negocio), y deberían ser reconocibles por las personas con conocimiento del dominio.
- ▶ Los paquetes del análisis no deberían basarse en requisitos no funcionales o en el dominio de la solución.
- ▶ Los paquetes del análisis probablemente se convertirán en subsistemas en las dos capas de aplicación superiores del modelo de diseño.

### ▶ Paquetes de servicio

Aparte de proporcionar casos de uso a sus actores, todo sistema también proporciona un conjunto de servicios a sus clientes. Un cliente adquiere una combinación adecuada de servicios para ofrecer a sus usuarios los casos de uso necesarios para llevar a cabo su negocio:



Un servicio representa un conjunto coherente de acciones relacionadas funcionalmente –un paquete de funcionalidad– que se utiliza en varios casos de uso.

Un servicio es indivisible en el sentido de que el sistema necesita ofrecerlo o todo entero o nada en absoluto.

Los casos de uso son para los usuarios, y los servicios son para los clientes. Los casos de uso atraviesan los servicios, es decir, un caso de uso requiere acciones de varios servicios.



En el Proceso Unificado, el concepto de servicio está soportado por los paquetes de servicio. Los paquetes de servicio se utilizan en un nivel más bajo de la jerarquía (de agregación) de paquetes del análisis para estructurar el sistema de acuerdo a los servicios que proporciona.

Un paquete de servicio puede, por tanto, representar cierta funcionalidad "adicional" del sistema. Cuando un paquete de servicio queda excluido, también lo queda todo caso de uso cuya realización requiera el paquete de servicio.

Los paquetes de servicio pueden ser mutuamente excluyentes, o pueden representar diferentes aspectos o variantes del mismo servicio. Por ejemplo, "corrección ortográfica del inglés británico" y "corrección ortográfica del inglés americano" pueden ser dos paquetes de servicio diferentes que proporciona un sistema.

Los paquetes de servicio constituyen una entrada fundamental para las actividades de diseño e implementación subsiguientes, dado que ayudarán a estructurar los modelos de diseño e implementación en términos de subsistemas de servicio. En particular, los subsistemas de servicio tienen una influencia decisiva en la descomposición del sistema en componentes binarios y ejecutables.

Por tanto, los paquetes de servicio se revelan como nuestro instrumento fundamental para la reutilización durante el análisis. Esto tiene sus consecuencias tanto en el diseño como en la implementación del sistema, y finalmente obtiene como resultado componentes reutilizables.



Ingresa al Web site de la institución y realiza un análisis de los paquetes funcionales y de servicio que te brinda. Observa la agrupación de los casos de uso y realiza una síntesis de lo observado para comentarlo con el profesor.

## Descripción del análisis en término Dinámicos

### ***Análisis de un caso de uso – Identificar clases de análisis***

Identificar las clases del análisis cuyos objetos son necesarios para llevar a cabo el flujo de sucesos del caso de uso.

#### ***Identificación de clases del análisis***

En este paso, identificamos las clases de control, entidad, e interfaz necesarias para realizar los casos de uso y esbozamos sus nombres, responsabilidades, atributos y relaciones.

Los casos de uso descritos en los requisitos no siempre están suficientemente detallados como para poder identificar clases del análisis. La información sobre el interior del sistema normalmente no es de interés durante la captura de requisitos y por tanto puede haberse dejado de lado. Por tanto, para identificar las clases del análisis puede que tengamos que refinar las descripciones de los casos de uso en lo referente al interior del sistema. Este refinamiento debe reflejarse en la descripción textual de flujos de sucesos - análisis de la realización de los casos uso.

*Podemos utilizar las siguientes normas generales para identificar las clases del análisis:*

**a)** Identificar clases de entidad mediante el estudio en detalle de la descripción del caso de uso y de cualquier modelo del dominio que se tenga, y después considerar qué información debe utilizarse y manipularse en la realización del caso de uso. Sin embargo, debemos ser conscientes de la información que es mejor capturar como atributo, la que es preferible asociar a clases de interfaz o de control, o la que simplemente no es necesaria para la realización del caso de uso; las "informaciones" de estos tipos no deberían modelarse como clases de entidad.

**b)** Identificar una clase de interfaz central para cada actor humano, y dejar que esta clase represente la ventana principal del interfaz de usuario con el cual interactúa el

actor. Si el actor ya interactúa con una clase de interfaz, deberíamos considerar el reutilizarla para conseguir una buena facilidad de uso de la **interfaz de usuario** (Unidad VI). Además, esas clases de interfaz centrales normalmente se consideran agregados de clases de interfaz más primitivas.

**c)** Identificar una clase de interfaz primitiva para cada clase de entidad que hayamos encontrado anteriormente. Estas clases representan objetos lógicos con las cuales interactúa el actor (humano) en la interfaz de usuario durante el caso de uso. Estas clases de interfaz primitivas pueden refinarse después de acuerdo a diversos criterios de facilidad.

**d)** Identificar una clase de interfaz central para cada actor que sea un sistema externo, y dejar que esta clase represente la interfaz de comunicación. Recuérdese que un actor del sistema puede ser cualquier cosa desde unidades software o hardware que interactúan con nuestro sistema, tales como impresoras, terminales, dispositivos de alarma, sensores, etc. Si las comunicaciones del sistema se dividen en varios niveles de protocolo, puede ser innecesario que el modelo de análisis distinga algunos de esos niveles. Si esto es así, debemos identificar clases de interfaz separadas para cada nivel de interés.

**e)** Identificar una clase de control responsable del tratamiento del control y de la coordinación de la realización del caso de uso, y después refinar esta clase de control de acuerdo a los requisitos del caso de uso. Por ejemplo, en algunos casos el control se encapsula mejor dentro de una clase de interfaz, especialmente si el actor maneja gran parte del control. En estos casos la clase de control no es necesaria. En otros casos el control es tan complejo que es mejor encapsularlo en dos o más clases de control. En estos casos es necesario dividir la clase de control.



Debemos recoger en un diagrama de clases las clases del análisis que participan en un realización de caso de uso. Utilizaremos este diagrama para mostrar las relaciones que se utilizan en la realización del caso de uso.

**Descripción de interacciones entre objetos del análisis** Cuando tenemos un esbozo de las clases necesarias para realizar el caso de uso, debemos describir cómo interactúan sus correspondientes objetos del análisis. Esto se hace mediante diagramas de colaboración que contienen las instancias de actores participantes, los objetos del análisis, y sus enlaces. Si el caso de uso tiene flujos o subflujos diferenciados y distintos, suele ser útil crear un diagrama de colaboración para cada flujo. Esto contribuye a hacer más clara la realización del caso de uso, y también hace posible extraer diagramas de colaboración que representa interacciones generales y reutilizables.

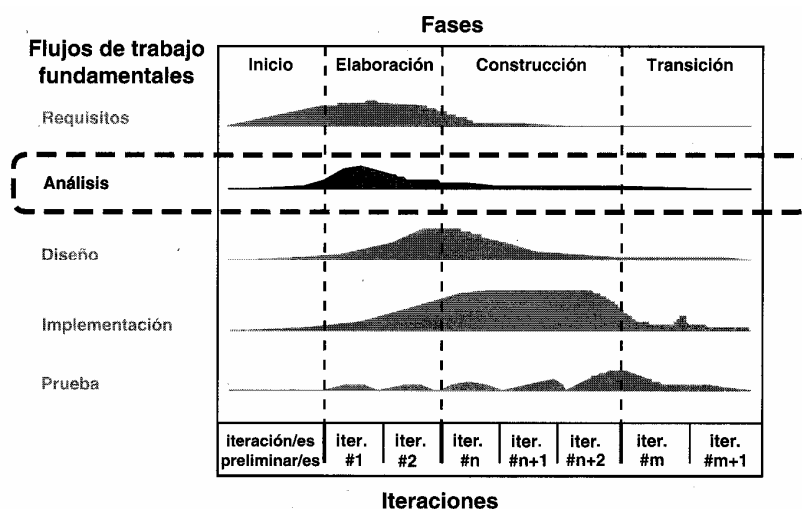
Un diagrama de colaboración se crea comenzando por el principio del flujo del caso de uso y continuando el flujo paso a paso, decidiendo qué interacciones de objetos del análisis e instancias de actor son necesarias para realizarlo. Normalmente los objetos encuentran su razón natural en la secuencia de interacciones de la realización del caso de uso.

Cada clase del análisis identificada en el paso anterior debería tener al menos un objeto que participe en un diagrama de colaboración. Si no lo tiene, la clase del análisis es superflua ya que no participa en ninguna realización de caso de uso.

Los mensajes no se asocian a operaciones debido a que no especificamos operaciones en las clases del análisis. En cambio, un mensaje debería reflejar el propósito del objeto invocante en la interacción con el objeto invocado. Este "propósito" es el origen de una responsabilidad del objeto receptor y podría llegar a ser incluso el nombre de una responsabilidad.

En algunos casos es apropiado complementar el diagrama de colaboración con descripciones textuales, especialmente si el mismo caso de uso tiene muchos diagramas de colaboración que describen o si hay diagramas que representan flujos complejos. Esas descripciones textuales deberían recogerse en el artefacto flujo de sucesos-análisis de la realización del caso de uso.

## El papel del análisis en el ciclo de vida del software



Las iteraciones iniciales de la elaboración se centran en el análisis. Eso contribuye a obtener una arquitectura estable y sólida y facilita una comprensión en profundidad de los requisitos. Más adelante, al término de la fase de elaboración y durante la construcción, cuando la arquitectura es estable y se comprenden los requisitos, el énfasis pasa en cambio al diseño y a la implementación.

El propósito y objetivo del análisis debe alcanzarse de algún modo en todo proyecto.

El proyecto utiliza el modelo de análisis, para describir los resultados del análisis, y mantiene la consistencia de este modelo a lo largo de todo el ciclo de vida del software.

El proyecto utiliza el modelo de análisis para describir los resultados del análisis pero considera a este modelo como una herramienta transitoria e intermedia -quizás de más interés durante la fase de elaboración-. Más adelante, cuando el diseño y la implementación están en marcha durante la fase de construcción, se deja de actualizar el

análisis. En su lugar, cualquier "tema de análisis" que aún quede, se resuelve como parte integrada dentro del trabajo de diseño en el modelo de diseño resultante

## Autoevaluación

---

1. Explique con sus palabras, qué diferencia establece entre las clases de Control, Interfaz y Entidad. Gráfiqelas.
2. ¿Definición de paquetes en el análisis es lo mismo que definición de subsistemas?
3. Después de haber comprendido los flujos de trabajo: Captura de Requisitos y Análisis, ¿qué conclusiones saca con respecto a la ARQUITECTURA DEL SISTEMA (en sus dos vistas)?



# ► Unidad II

## Introducción al Diseño

### Objetivos

- Relacionar y vincular contenidos vistos en el flujo de trabajo captura de requisitos y análisis, con los nuevos contenidos propuestos.
- Comprender el significado de las fases e iteraciones del diseño.
- Ser capaces de visualizar y reflexionar sobre el diseño, utilizando una notación común.
- Identificar las herramientas y conocer su funcionalidad en el modelo.

### Temas de la Unidad

¿Qué hacemos en el diseño?

Diseño en el ciclo de vida (Visión Estática)

Modelo de diseño (descripción arquitectura)

Clases de diseño – Diagrama de transición de estados

Casos uso de diseño

Diagrama de clases

Subsistema de diseño

Interfaz

Modelo de Despliegue (descripción de la arquitectura)

Diagrama de despliegue

Ambiente Implementación

Diseño en el ciclo de vida (Visión Dinámica)

Diseño arquitectura

Diseño de casos de uso

Diseño de clases

Diseño de subsistemas

Papel del Diseño en el Ciclo de vida Proceso Unificado

Comparación entre el Análisis y el Diseño

Autoevaluación

## ¿Qué hacemos en el diseño?

---



En el diseño modelamos el sistema y encontramos su forma (incluida la arquitectura) para que soporte todos los requisitos -incluyendo los requisitos no funcionales y otras restricciones- que se le suponen.

---

Una entrada esencial en el diseño es el resultado del análisis, esto es, el modelo de análisis. Recordemos que el modelo de análisis proporciona una comprensión detallada de los requisitos y lo que es más importante, impone una estructura del sistema que debemos esforzarnos por conservar lo más fielmente posible cuando demos forma al sistema.

En concreto, los propósitos del diseño son:

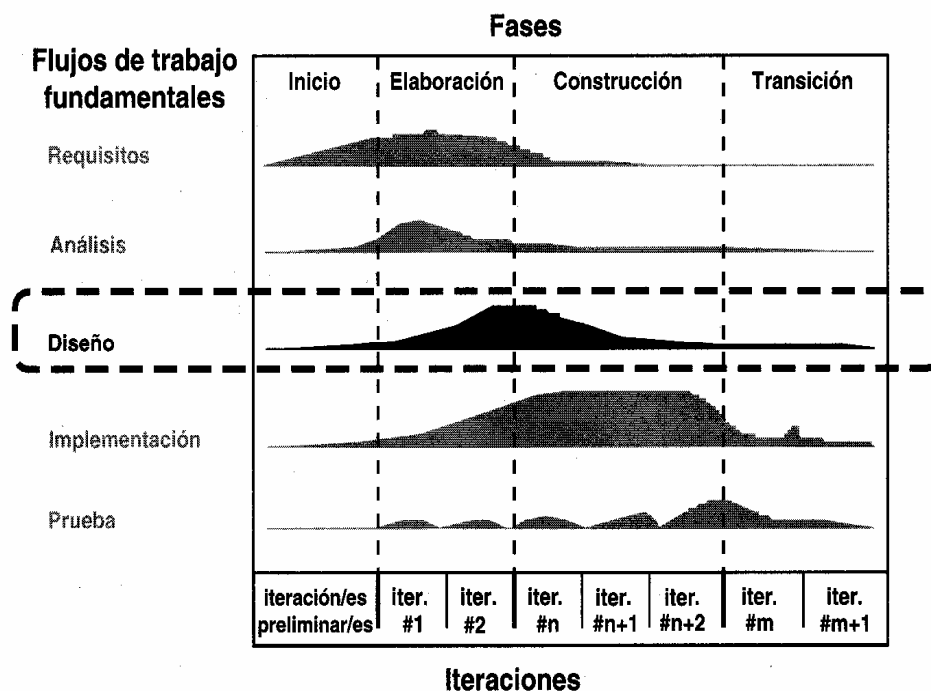
- ▶ Adquirir una comprensión, en profundidad, de los aspectos relacionados con los requisitos no funcionales (módulo Sistemas III) y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de distribución y concurrencia, tecnologías de interfaz de usuario, tecnologías de gestión de transacciones, etc.
- ▶ Crear una entrada apropiada y un punto de partida para actividades de implementación subsiguientes capturando los requisitos o subsistemas individuales, interfaces y clases.
- ▶ Ser capaces de descomponer los trabajos de implementación en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo, teniendo en cuenta la posible concurrencia. Esto resulta útil en los casos en los que la descomposición no puede ser hecha basándose en los resultados de la captura de requisitos (incluyendo el modelo de casos de uso) o análisis (incluyendo el modelo de análisis), (modulo Sistemas III).
- ▶ Capturar las interfaces entre los subsistemas. Esto ayuda cuando reflexionamos sobre la arquitectura y cuando utilizamos interfaces como elementos de sincronización entre diferentes equipos de desarrollo.
- ▶ Crear una abstracción de la implementación del sistema, en el sentido de que la implementación es un refinamiento directo del diseño que rellena lo existente, sin cambiar la estructura. Esto permite la utilización de tecnologías como la generación de código y la ingeniería de ida y vuelta entre el diseño y la implementación.

## Diseño en el ciclo de vida (vista estática)

El diseño esta conformado esquemáticamente por los siguientes Modelos y diagramas

	<b>Modelos</b>	<b>Diagramas</b>
<b>D I S E Ñ O</b>		Diagrama de clases de diseño
	Modelo	Diagrama De transición de estados
	Diseño	Subsistema
	Modelo	Diagrama de despliegue
	Despliegue	Ambiente Implementación

## El papel del “Diseño” en el ciclo de vida del Proceso Unificado



El diseño es el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción. Esto contribuye a una arquitectura estable y sólida y a crear

un plano del modelo de implementación. Más tarde, durante la fase de construcción, cuando la arquitectura es estable y los requisitos están bien entendidos, el centro de atención se desplaza a la implementación.

No obstante, el modelo de diseño está muy cercano al de implementación, lo que es natural para guardar y mantener el modelo de diseño a través del ciclo de vida completo del software. Esto es especialmente cierto en la ingeniería de ida y vuelta, donde el modelo de diseño se puede utilizar para visualizar la implementación y para soportar las técnicas de programación gráfica.

---

## Modelo de Diseño

---



El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar.

---

Además, el modelo de diseño sirve de abstracción de la implementación del sistema y es, de ese modo utilizado, como una entrada fundamental de las actividades de implementación.

En el modelo de diseño, los casos de uso son realizados por las clases de diseño y sus objetos. Esto se representa por colaboraciones en el modelo de diseño y denota *realización de caso de uso-diseño*. Adviértase que realización de caso de uso-diseño es diferente de la realización de casos de uso-análisis.

---

## Clases de diseño

---



Una clase de diseño es una abstracción de una clase o construcción similar en la implementación del sistema.

---

Presenta las siguientes características:

- ▶ El lenguaje utilizado para especificar una clase del diseño, es lo mismo que el lenguaje de programación. Consecuentemente, las operaciones, parámetros, atributos, tipos y demás son especificados utilizando la sintaxis del lenguaje de programación elegido.
- ▶ La visibilidad de los atributos y las operaciones de una clase de diseño se especifica con frecuencia. Por ejemplo, las palabras clave *public*, *protected*, *private* son usadas muy a menudo en C++.

- ▶ Las relaciones de aquellas clases del diseño implicadas con otras clases, a menudo tienen un significado directo cuando la clase es implementada. Por ejemplo, la generalización o algún estereotipo de generalización tiene una semántica que se corresponde con el significado de generalización (o herencia) en el lenguaje de programación. Esto es, las asociaciones y agregaciones a menudo se corresponden con variables (atributos) de clases en la implementación para proporcionar referencias entre objetos.
- ▶ Los métodos (o lo que es lo mismo, las realizaciones de operaciones) de una clase del diseño tienen correspondencia directa con el correspondiente método en la implementación de las clases (esto es, en el código). Si los métodos se especifican en el diseño, se suelen especificar en lenguaje natural, o en pseudo código, y por eso pueden ser utilizados como comentarios en las implementaciones del método.
- ▶ Una clase de diseño puede posponer el manejo de algunos requisitos para las subsiguientes actividades de implementación, indicándolos como requisitos de implementación de la clase. Esto hace posible posponer decisiones que son inapropiadas de manejar en el modelo de diseño, como las que tienen que ver con el código de la clase.

### ***Realización de un caso de Uso del diseño***



Una realización caso de uso -diseño es una colaboración en el modelo de diseño que describe cómo se realiza un caso de uso específico, y cómo se ejecuta, en términos de clases de diseño y sus objetos. Una *realización de caso de uso-diseño* proporciona una traza directa a una realización de caso de uso-análisis en el modelo de análisis.

Una realización de caso de uso-diseño tiene una descripción de flujo de eventos textual:

- ▶ diagramas de clases que muestra sus clases de diseño participantes,
- ▶ diagramas de interacción que muestran la realización de un flujo o escenario concreto de un caso de uso en términos de interacción entre objetos del diseño

Una realización de caso de uso-diseño proporciona una realización física de la realización de caso de uso-análisis para la que es trazado, y también gestiona muchos requisitos no funcionales (es decir, requisitos especiales) capturados de la realización de caso de uso-análisis.

Por consiguiente, una realización de caso de uso-diseño puede, como pueden las clases diseñadas, posponer el manejo de algunos requisitos hasta las subsiguientes actividades de implementación anotándolas como requisitos de implementación en la realización.

## ***Diagramas de clases***

Una clase de diseño y sus objetos, y de ese modo también los subsistemas que contienen las clases de diseño, a menudo participan en varias realizaciones de casos de uso.

También puede darse el caso de algunas operaciones, atributos y asociaciones sobre una clase específica que son relevantes para sólo una realización de caso de uso. Esto es importante para coordinar todos los requisitos que diferentes realizaciones de casos de uso imponen a una clase, a sus objetos y a los subsistemas que contiene.

Para manejar todo esto, utilizamos diagramas de clases conectados a una realización de caso de uso, mostrando sus clases participantes, subsistemas y sus relaciones.

Para ejemplificar cada uno de los diagramas (de clases, de interacción y de estados) que corresponden a esta nueva etapa del Proceso Unificado de Software, vamos a continuar con el ejemplo que comenzamos a trabajar en SISTEMAS III, ¿Se acuerdan?

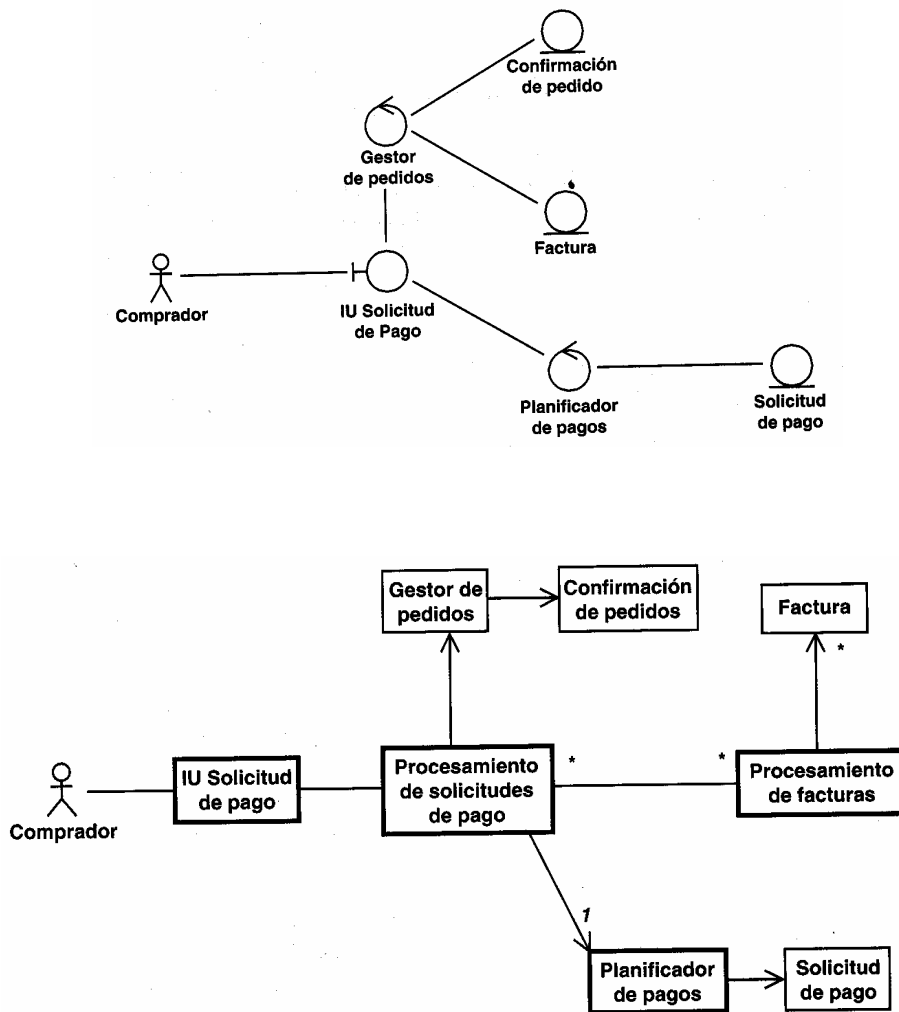
### ***“El consorcio Interbank estudia un Sistema Informático”***

El consorcio Interbank, una institución financiera ficticia, se enfrenta a cambios importantes debido a la desregulación, la nueva competencia, y las posibilidades abiertas por la WWW. El consorcio quiere desarrollar aplicaciones nuevas que soporten los rápidamente mercados financieros. Ha encargado a su subsidiaria de desarrollo de software, que comience el desarrollo de esas aplicaciones.

Interbank Software decide diseñar el “Sistema de Pagos y Facturación” en colaboración con algunos de sus principales clientes. El sistema utilizara Internet para el envío de pedidos, facturas y pagos entre compradores y vendedores. La motivación del banco para el desarrollo del sistema es atraer nuevos clientes ofreciendo comisiones bajas por el proceso de los pagos. El banco también podrá reducir sus costos laborales procesando las solicitudes de cambio automáticamente por Internet, en lugar de manualmente mediante cajeros.

Las motivaciones para compradores y vendedores son reducir los costos, el papeleo y el tiempo de proceso. Por ejemplo ya no tendrán que enviar pedidos o facturas por correo. El pago de facturas se llevara a cabo entre las computadoras del comprador y del vendedor. Ambos tendrán también una visión general mejor del estado de sus facturas y pagos.

Para construir el diagrama de clases de Diseño, debemos basarnos, en los diagramas de colaboración que se crearon en el flujo de trabajo análisis (Módulo Sistemas III). Recordemos el Diagrama de colaboración para el Caso de uso “Pagar Factura”



Las clases que participan en la realización del caso de uso Pagar Factura y sus asociaciones. En el diagrama, mostramos las clases activas con bordes más gruesos

### ► Qué considerar al momento de construir el diagrama de clases

Este modelo estará compuesto de bloques, los cuales serán objetos de diseño. Estos bloques muestran cómo el sistema está diseñado, y serán codificados en la etapa de implementación. Los bloques pueden ser implementados por una clase o por varias clases diferentes.

Inicialmente cada objeto de análisis se transforma en un bloque. Esto ayuda al rastreo entre los modelos. Es decir que cada objeto de análisis es rastreable a un bloque, si sufre un cambio en el modelo de análisis, estos serán locales y se podrán rastrear a los bloques del modelo del diseño. Este rastreo es bidireccional y se realiza asignando rectángulos a cada uno de los objetos del modelo de Análisis.

## Subsistema de Diseño



Los subsistemas de diseño son una forma de organizar los artefactos del modelo de diseño en piezas más manejables. Un subsistema puede constar de clases del diseño, realizaciones de caso de uso, interfaces y otros subsistemas (recursivamente). Por otro lado, un subsistema puede proporcionar interfaces que representan la funcionalidad que exportan en términos de operaciones.

Un subsistema debería ser *cohesivo*; es decir, sus contenidos deberían encontrarse fuertemente asociados. Además, los subsistemas deberían ser débilmente *acoplados*; esto es, sus dependencias entre unos y otros, o entre sus interfaces, deberían ser mínimas.

Los subsistemas de diseño también deberían poseer las siguientes características, pudiendo representar:

- ▶ una separación de aspectos del diseño (por ejemplo, en un sistema grande, algunos subsistemas pueden desarrollarse por separado, y quizá de manera simultánea, por equipos de desarrollo diferentes con aptitudes de diseño distintas);
- ▶ componentes de trazo grueso en la implementación del sistema es decir, componentes que proporcionan varios interfaces compuestos a partir de otros varios componentes de trazo más fino, como los que especifican clases de implementación individuales;
- ▶ productos software reutilizados que han sido encapsulados en ellos, (por tanto, pueden utilizarse los subsistemas en el modelo de diseño para representar la integración de productos software reutilizados);
- ▶ sistemas heredados (o parte de ellos) encapsulándolos (por tanto, podemos utilizar subsistemas para incluir sistemas heredados en el modelo de diseño)

## Interfaz

Las interfaces se utilizan para especificar las operaciones que proporcionan las clases y los subsistemas del diseño.

Una clase del diseño que proporcione una interfaz, debe proporcionar también métodos que realicen las operaciones de la interfaz. Un subsistema que proporcione una interfaz debe contener también clases del diseño u otros subsistemas (recursivamente) que proporciona interfaz.



Las interfaces constituyen una forma de separar: la especificación de la funcionalidad en términos de operaciones, de sus implementaciones en términos de métodos.

La mayoría de las interfaces entre subsistemas se consideran relevantes para la arquitectura debido a que definen las interacciones permitidas entre los subsistemas.



En algunos casos también es útil diseñar interfaces estables, antes de implementar mediante subsistemas la funcionalidad que representan. De esta forma, los equipos de desarrollo encargados del diseño de los subsistemas pueden considerar estas interfaces como requisitos, y pueden utilizarse también como instrumentos de sincronización entre diferentes equipos que pueden estar trabajando de manera simultánea con diferentes subsistemas.



Ingresa a la plataforma y revisa los diferentes diseños de interfaz que la cátedra te propone, desarrollos intranet y diseños para Web. Realiza una crítica constructiva a cada modelo.

## Modelo de despliegue

### ► Diagrama de despliegue



El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo.

El modelo de despliegue se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño.

Podemos observar lo siguiente sobre el modelo de despliegue:

- Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo hardware similar.
- Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como *Internet*, *intranet*, *bus*, y similares.
- El modelo de despliegue puede describir diferentes configuraciones de red, incluidas las configuraciones para pruebas y para simulación.
- La funcionalidad (los procesos) de un nodo se definen por los componentes que se distribuyen sobre ese nodo
- El modelo de despliegue en sí mismo representa una correspondencia entre la arquitectura software y la arquitectura del sistema (el hardware)

Continuando con el ejemplo de Interbank, el diagrama de despliegue podría visualizarse de la siguiente manera:

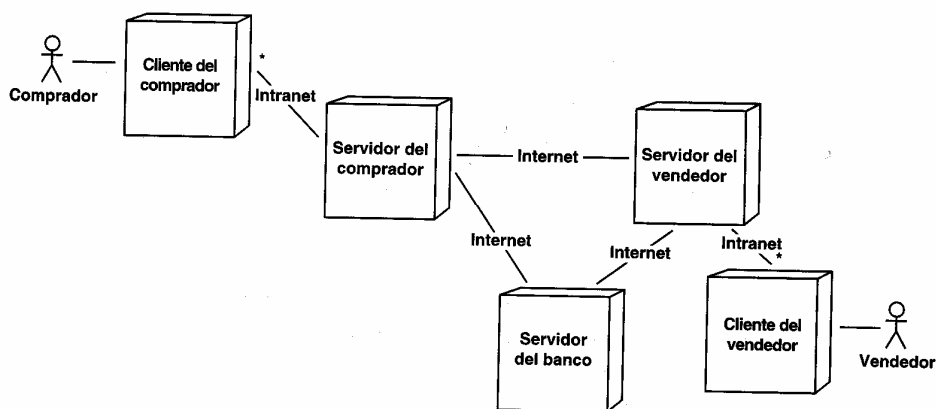


Diagrama de despliegue para el sistema Interbank

A esta presentación del diagrama, se la puede acompañar con un Lay out del equipamiento, (gráfico del plano de la organización) con los equipos distribuidos en él.

## ► Ambiente Implementación

### Requisitos de implementación



Los requisitos de la implementación son una descripción textual que recoge requisitos, tales como los requisitos no funcionales, sobre una realización de caso de uso. Nos referimos a requisitos que se capturan sólo en la fase de diseño, pero que es mejor tratar en la implementación.

Algunos de estos requisitos pueden haber sido identificados en flujos de trabajo anteriores y por tanto sólo se cambian a una realización de caso de uso.

Sin embargo, puede que algunos de ellos sean requisitos nuevos o derivados, que se identifican a medida que avanza el trabajo del diseño.

Un ejemplo de requisitos de implementación sobre la realización de caso de uso- diseño Pagar Factura.

*Un objeto de la clase activa Procesamiento de Solicitudes de pago debería ser capaz de soportar 10 clientes de comprador deferentes, sin un retraso perceptible para cada determinado comprador.*

### Otros aspectos que se deben incluir en el modelo

- El lugar donde el sistema será ejecutado.
- El lenguaje de programación: Se debe considerar las propiedades básicas del mismo, por ejemplo, si maneja herencia, polimorfismo, etc., los componentes para programación que vienen en librerías.

- ▶ Otros productos que deben ser utilizados
- ▶ Bases de Datos
- ▶ Administradores de Interfaces
- ▶ En grandes sistemas deberá tenerse en cuenta como debe configurarse, administrarse, y distribuirse el sistema.
- ▶ Como se distribuirán las bases de datos
- ▶ El sistema operativo a utilizar
- ▶ Medios de comunicación, tipos, y sus características.

Estos puntos se expresan en forma textual y detallada, completando así, los aspectos del modelo de despliegue necesarios para obtener una visión del proyecto a implementar.

## **Diseño en el ciclo de vida (vista dinámica)**

### ***Diseño de la arquitectura***

El objetivo del diseño de la arquitectura es esbozar los modelos de diseño y despliegue y su arquitectura mediante la identificación de los siguientes elementos

- a- Clases del diseño significativas para la arquitectura, como las clases activas.
- b- Subsistemas y sus interfaces
- c- Nodos y sus configuraciones de red.

A lo largo de esta actividad los arquitectos consideran distintas posibilidades de reutilización, como la reutilización de partes de sistemas parecidos, o de productos software generales.

Los subsistemas, interfaces u otros elementos del diseño resultantes se añadirán posteriormente al modelo de diseño.

### ***a- Identificación de clases del diseño***

#### **▶ Diseño de una clase**

El propósito de diseñar una clase es crear una clase del diseño que cumpla su papel en las realizaciones de los casos de uso y los requisitos no funcionales que se aplican a estos. Esto incluye el mantenimiento del diseño de clases en sí mismo y los siguientes aspectos:

- » Esbozar las clases
- » Sus operaciones
- » Sus atributos
- » Las relaciones en las que participa
- » Sus métodos (que realizan sus operaciones)

## ► **Esbozar la clase del diseño**

---

Como un primer paso, necesitamos esbozar una o varias clases del diseño, dada la entrada en términos de clases de análisis y/o interfaces.

Cuando tomamos una interfaz como entrada, suele ser simple y directo el asignar a una clase de diseño para que proporcione esa interfaz.

Cuando se dan como entrada una o varias clases del análisis, los métodos utilizados dependen del estereotipo de la clase de análisis:

### ***Puede ser: Diseñar clases de interfaz***

Es dependiente de la tecnología de interfaz específica que se utilice. Por ejemplo, las clases de interfaz diseñadas en Visual Basic implicarían clases del diseño estereotipadas como «forms» junto con otras clases del diseño que representan «controls», posiblemente «controls» de ActiveX, en el interfaz de usuario.

### ***Diseñar clases de entidad***

Que representen información persistente (o clases que tienen otros requisitos persistentes) a menudo implica el uso de tecnologías de bases de datos específicas. Por ejemplo, se puede incluir la creación de clases de diseño para hacer la correspondencia con tablas en un modelo de datos relacional. Este paso puede automatizarse parcialmente utilizando las herramientas de modelado disponibles actualmente, aunque puede ser bastante delicado y requerir la adopción de una estrategia persistente. Puede haber muchos aspectos del diseño involucrados en la adopción de una estrategia, especialmente a la hora de hacer corresponder el modelo de diseño orientado a objetos con el modelo de datos relacional.

### ***Diseñar clases de control***

Es una tarea delicada. Debido a que encapsulan secuencias, coordinación de otros objetos o algunas veces pura lógica del negocio, es necesario considerar los siguientes aspectos:

- Aspectos de distribución: si la secuencia necesita ser distribuida y manejada por diferentes nodos de una red, se puede requerir separar las clases del diseño en diferentes nodos para realizar la clase de control.
- Aspectos de rendimiento: puede que no sea justificable tener clases del diseño separadas para realizar la clase de control. En cambio, la clase de control podrá realizarse por las mismas clases del diseño que están realizando algunas clases de interfaz o clases de entidad relacionadas.
- Aspectos de transacción: las clases de control a menudo encapsulan transacciones. Sus correspondientes diseños deben incorporar cualquier tecnología de manejo de transacción existente que se esté utilizando.

Las clases de diseño identificadas en este paso deberán ser asignadas trazando dependencias a las correspondientes clases de análisis que son diseñadas. Es importante conservar en mente los "orígenes" de la clase de diseño cuando sea refinada en los pasos posteriores.

## ► Identificar operaciones

En esta etapa identificamos las operaciones que las clases de diseño van a necesitar y describimos esas operaciones utilizando la sintaxis de los lenguajes de programación. Esto incluye especificar la visibilidad de cada operación (por ejemplo, *public*, *protected*, *private* en C++).

## ► Identificar atributos

En este paso, identificamos los atributos requeridos por la clase de diseño y los describimos utilizando la sintaxis del lenguaje de programación. Un atributo especifica una propiedad de una clase de diseño y está a menudo implicado y es requerido por las operaciones de la clase (como se consideró en el paso anterior). Las siguientes normas generales han de tenerse en cuenta cuando se identifiquen atributos:

Considerar los atributos sobre cualquier clase de análisis que tenga una traza con la clase de diseño. Algunas veces estos atributos implican la necesidad de uno o más atributos de la clase de diseño.

Los tipos de atributos disponibles están restringidos por el lenguaje de programación. Cuando se elige un tipo de atributo, intentar reutilizar uno que ya exista.

Una simple instancia de atributo no puede ser compartida por varios objetos de diseño. Si esto fuera necesario, el atributo necesita ser definido en una clase separada.

Si una clase de diseño llega a ser muy complicada de comprender por culpa de sus atributos, algunos de estos atributos pueden ser separados en clases independientes.

Si hay muchos atributos o son complejos los atributos de una clase, se puede ilustrar ésta en un diagrama de clases separado que muestre solamente el apartado de atributos.

## ► Identificar asociaciones y agregaciones

Los objetos de diseño interactúan unos con otros en los diagramas de secuencia. Estas interacciones a menudo requieren asociaciones entre las clases correspondientes. Las instancias de las asociaciones deben ser utilizadas para abarcar las referencias con otros objetos, y para agrupar objetos en agregaciones con el propósito de enviarles mensajes.

Por ello deben tenerse en cuenta que algunas veces las relaciones (en el modelo de análisis) implican la necesidad de una o varias relaciones correspondientes (en el modelo de diseño) que involucre la clase de diseño.

Refinar la multiplicidad de las asociaciones, nombres de rol, clases de asociación, roles de ordenación, roles de cualificación de acuerdo con el soporte del lenguaje de programación utilizado. Por ejemplo, los nombres de los roles pueden llegar a ser atributos de la clase de diseño cuando se genere el código, por lo tanto se restringe la forma de los nombres de roles. O, una asociación de clases puede llegar a ser una clase nueva entre las dos clases originales, por lo que se requiere nuevas asociaciones con la multiplicidad apropiada entre la clase "asociación" y las otras dos clases.

Refinar la navegabilidad de las asociaciones; Considerar los diagramas de interacción en los que se emplean asociaciones. La dirección de las transmisiones de los mensajes entre los objetos de diseño implica que se corresponda con la navegabilidad de las asociaciones entre sus clases.

### ► **Identificar las generalizaciones**

---

Las generalizaciones deben ser utilizadas con la misma semántica definida en el lenguaje de programación. Si el lenguaje de programación no admite generalización (o herencia), las asociaciones y/o agregaciones deben utilizarse en lugar de ésta para proporcionar delegación desde los objetos de clases más específicas a objetos de clases más generales (por ejemplo, enviar un mensaje como petición para realizar trabajo, obtener el trabajo realizado y un mensaje de confirmación).

### ► **Generalizaciones en el modelo diseño**

---

Tanto Facturas como Pedidos cambian de estado de forma similar y soportan operaciones similares. Ambas son especializaciones de una clase Objeto de Comercio más general.

La clase Objeto de Comercio generaliza Facturas y Pedidos. Adviértase que estas generalizaciones también existen en el modelo de análisis entre las correspondientes clases de análisis.

### ► **Describir los métodos**

---

Los métodos pueden ser utilizados durante el diseño para especificar cómo se deben realizar las operaciones. Por ejemplo, un método puede especificar un algoritmo que sea utilizado para realizar una operación. El método puede ser especificado utilizando lenguaje natural o pseudocódigo si se considera más apropiado.

No obstante, la mayoría de las veces los métodos no son especificados durante el diseño. En cambio, pueden ser creados durante la implementación utilizando un lenguaje de programación directamente.

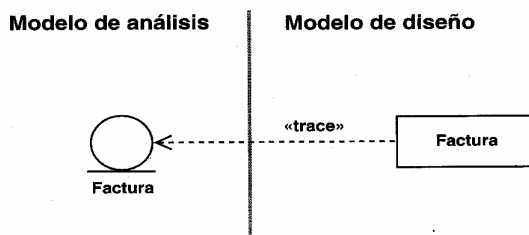
### ► **Identificación de clases del diseño a partir de clases del análisis**

---

Podemos esbozar inicialmente algunas clases del diseño a partir de las clases del análisis significativas para la arquitectura que encontramos en el análisis. Además, podemos utilizar las relaciones entre esas clases del análisis para identificar un conjunto tentativo de relaciones entre las correspondientes clases del diseño.

Por ejemplo:

La clase del diseño Factura se esboza inicialmente a partir de la clase de entidad Factura del modelo de análisis



La clase diseño Factura se esboza inicialmente a partir de la clase de entidad Factura

## ► Descripción de las interacciones entre objetos del diseño

Cuando tenemos un esquema de las clases del diseño necesarias para realizar el caso de uso, debemos describir cómo interactúan sus correspondientes objetos del diseño. Esto se hace mediante diagramas de secuencia que contienen las instancias de los actores, los objetos del diseño, y las transmisiones de mensajes entre éstos, que participan en el caso de uso. Si los casos de uso tienen varios flujos o subflujos distintos, suele ser útil en crear un diagrama de secuencia para cada uno de ellos. Esto puede hacer más clara la realización del caso de uso, y también permite extraer diagramas de secuencia que representan interacciones generales y reutilizables.

Para comenzar este paso, debemos estudiar la correspondiente realización de caso de uso- análisis. Podemos utilizarla para obtener un esbozo de la secuencia necesaria de mensajes entre los objetos del diseño, aunque puede que se hayan añadido muchos objetos del diseño nuevos. En algunos casos, incluso puede merecer la pena transformar un diagrama de colaboración de la realización de caso de uso-análisis en un esbozo inicial del correspondiente diagrama de secuencia.

Para crear un diagrama de secuencia, debemos comenzar por el principio del flujo del caso de uso, y después seguir ese flujo paso a paso, decidiendo qué objetos del diseño y qué interacciones de instancias de actores son necesarias para realizar cada paso. En la mayoría de los casos, los objetos se ajustan de manera natural a la secuencia de interacciones de la realización de caso de uso. Debemos observar lo siguiente sobre estos diagramas de secuencia:

- El causante de la invocación del caso de uso es un mensaje de una instancia de un actor hacia un objeto del diseño.
- Cada clase del diseño identificada en el paso anterior debería tener al menos un objeto del diseño participante en el diagrama de secuencia.
- Los mensajes que realizan el caso de uso se envían entre líneas de vida de los objetos. Un mensaje puede tener un nombre temporal que después pasará a ser el nombre de una operación.
- La secuencia en el diagrama debería ser la principal preocupación, ya que la realización de caso de uso-diseño es la entrada principal para la implementación del caso de uso. Es importante comprender el orden cronológico de las transferencias de mensajes entre objetos.
- Utilizaremos etiquetas y el flujo de sucesos-diseño para complementar los diagramas de secuencia.

- ▶ El diagrama de secuencia debería tratar todas las relaciones del caso de uso que realiza. Por ejemplo, si el caso de uso A es una especialización de un caso de uso B mediante una relación de generalización, el diagrama de secuencia que realice el caso de uso A puede tener que hacer referencia a la realización (es decir, al diagrama de secuencia) del caso de uso B. Obsérvese que también es posible que ese tipo de referencias esté presente en las correspondientes realizaciones de caso de uso-análisis.

---

## ***b- Identificación de nodos y configuraciones de red***

Las configuraciones físicas de red suelen tener una gran influencia sobre la arquitectura del software incluyendo las clases activas que se necesitan y la distribución de la funcionalidad entre los nodos de red. Las configuraciones de red habituales utilizan un patrón de tres capas en el cual los clientes, se dejan en una capa, la funcionalidad de base de datos en otra, y la lógica del negocio en otra.

El patrón cliente / servidor simple es un caso especial de este patrón de tres capas en el cual la lógica del negocio o de la aplicación se ubica en una de las otras capas (en la capa del cliente o en la de base de datos).

Entre los aspectos de configuraciones de red podemos resaltar:

- ▶ ¿Qué nodos se necesitan, y cuál debe ser su capacidad en términos de potencia de proceso y tamaño de memoria?
- ▶ ¿Qué tipo de conexiones debe haber entre los nodos, y qué protocolos de comunicaciones se deben utilizar sobre ellas?
- ▶ ¿Qué características deben tener las conexiones y los protocolos de comunicaciones, en aspectos tales como ancho de banda, disponibilidad y calidad?
- ▶ ¿Es necesario tener alguna capacidad de proceso redundante, modos de fallo, migración de procesos, mantenimiento de copias de seguridad de los datos, o aspectos similares?

---

## ***c- Identificación de subsistemas y sus interfaces***



Subsistemas constituyen un medio para organizar el modelo de diseño en piezas manejables. Pueden bien identificarse inicialmente como forma de dividir el trabajo de diseño, o bien pueden irse encontrando a medida que el modelo de diseño evoluciona y va "creciendo" hasta convertirse en una gran estructura que debe ser descompuesta

---

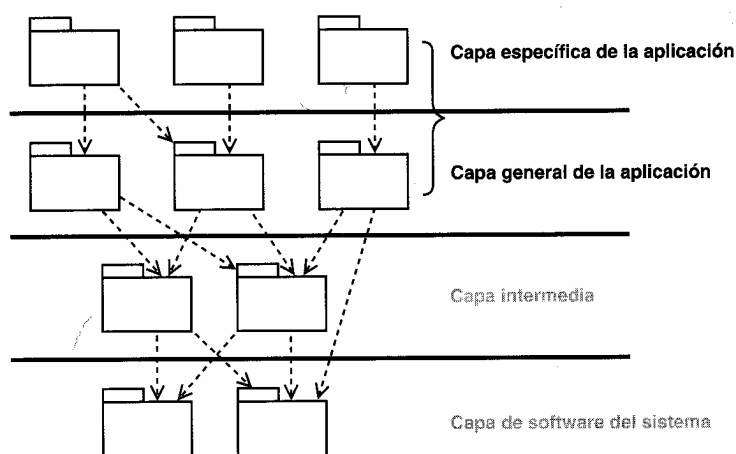


Obsérvese también que no todos los subsistemas se desarrollan internamente en el proyecto en curso. En realidad, algunos subsistemas representan productos reutilizados y otros son recursos existentes en la empresa. La inclusión de subsistemas de esos tipos en el modelo de diseño permite analizar y evaluar las alternativas de reutilización.

### ► **Identificación de subsistemas de aplicación**

En este paso identificamos los subsistemas de las capas específicas de la aplicación y general de la aplicación (es decir, los subsistemas en las dos capas superiores).

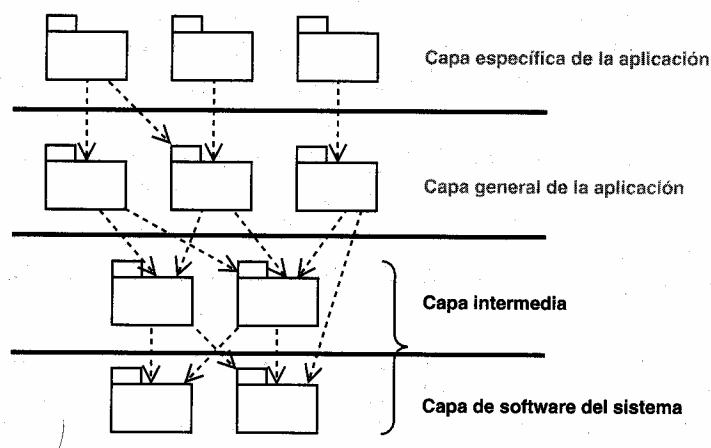
Si se hizo durante el análisis una descomposición adecuada en paquetes del análisis, podemos utilizar esos paquetes tanto como sea posible e identificar los correspondientes subsistemas dentro del modelo de diseño. Esto es especialmente importante en el caso de los paquetes de servicio (Módulo Sistemas III), de forma que podamos identificar subsistemas de servicio correspondientes que no rompan con la estructuración del sistema de acuerdo a los servicios que ofrece.



Subsistemas en las capas específica de la aplicación y general de la aplicación

### ► **Identificación de subsistemas intermedios y de software del sistema**

El software del sistema y la capa intermedia constituyen los cimientos de un sistema, ya que toda la funcionalidad descansa sobre software como sistemas operativos, sistemas de gestión de base de datos, software de comunicaciones, tecnologías de distribución de objetos, kits de diseño de IGU, y tecnologías de gestión de transacciones.



Los subsistemas de las capas intermedia y de software del sistema encapsulan productos software comprado

Debemos intentar limitar bien las dependencias de productos comprados, limitando así el riesgo asociado con su uso, en caso de que cambien en el futuro, o bien ser capaces de cambiar de fabricante si es necesario.



En esta instancia, el trabajo colaborativo con la cátedra de Bases de Datos y Programación, son fundamentales. En nuestra plataforma podrás encontrar el trabajo práctico integrador. La elaboración de este, te permitirá ver la relación de cada uno de los contenidos trabajados desde cada disciplina.

## Comparación entre el modelo de Análisis y el modelo de Diseño

Modelo de análisis	Modelo de diseño
Modelo conceptual, porque es una abstracción del sistema y permite aspectos de la implementación.	Modelo físico, porque es un plano de la implementación.
Genérico respecto al diseño (aplicable a varios diseños).	No genérico, específico para una implementación.
Tres estereotipos conceptuales sobre las clases: Control, Entidad e Interfaz.	Cualquier número de estereotipos (físicos) sobre las clases, dependiendo del lenguaje de implementación.
Menos formal.	Más formal.
Menos caro de desarrollar	Más caro de desarrollar
Menos capas.	Más capas.
Dinámico (no muy centrado en la secuencia).	Dinámico (muy centrado en las secuencias).
Bosquejo del diseño del sistema, incluyendo su arquitectura.	Manifiesto del diseño del sistema, incluyendo su arquitectura (una de sus vistas).
Creado principalmente como "trabajo de a pie" en talleres o similares.	Creado principalmente como "programación visual" en ingeniería de ida y vuelta; el modelo de diseño es realizado según la ingeniería de ida y vuelta con el

## Autoevaluación

1. Realice una lectura del siguiente enunciado, y teniendo en cuenta los contenidos tratados, sobre subsistemas, diagrama de despliegue, ambiente implementación, intente descubrir y describir cada uno de ellos.

*Una empresa de colectivos emite bonos para viajes regulares a docentes y estudiantes. Para sacar el abono, los usuarios deben presentar un certificado de domicilio y comprobante de docente o alumno, además se llena una solicitud de abono donde constan los datos personales. Con esta solicitud se lo registra al usuario, también se registra el abono solicitado en otro almacenamiento aclarando el n° de viajes autorizados y se le entrega al cliente el n° de abono.*

*El área cobros receipta el pago y emite la factura que se le entrega al usuario. Una vez recibido el pago, se lo registra en el almacenamiento de abonos. Del chofer se recibe un listado de los números de abono que recibió en el colectivo y con esto se realiza la descarga correspondiente del registro. Diariamente se emite un listado de facturación al Contador.*

2. Establezca las diferencias entre un diagrama de colaboración del flujo de trabajo “Análisis” y el diagrama de clases de diseño, del flujo de trabajo “Diseño”.
3. En la unidad se hace una clara diferencia entre una visión estática del Diseño y una visión dinámica del Diseño. Explique con sus palabras, cuál es esa diferencia, y fundamente el o los beneficios de esas miradas.



# ► Unidad III

## Cómo construimos los modelos

### Objetivos

- Aplicar los conocimientos adquiridos (flujo de trabajo análisis y diseño) en las unidades anteriores.
- Enfatizar el proceso de diseño que determina la estructura de un proyecto
- Trabajar un caso de estudio, en forma completa, desde el análisis.

### Contenidos de la Unidad

Introducción

Caso de estudio: Enunciado

#### 1. Modelado del Sistema de Negocio

1.1. Modelo de Caso de uso del Sistema de Negocio (diagrama y descripciones de los casos de uso).

1.2 Descripción de objetos activos trabajadores.

#### 2. Modelado del Sistema de Información

2.1. Modelo de casos de uso (diagrama de caso de uso y descripción de cada caso de uso).

2.2. Modelo de Objetos del Dominio.

2.3. Descripción o Prototipo de Interfaz.

#### 3. Flujo de trabajo del Análisis

3.1. Diagrama de Colaboración.

3.2. Subsistemas.

#### 4. Flujo de trabajo de Diseño

4.1 Diagrama de Despliegue

4.2 Ambiente de implementación:

4.3 Diagrama de Clases de Diseño.

4.4 Mapeo de Diagrama de Clase a Modelo Relacional

4.5 Diagrama de Estados.

#### Documentación para desarrollo o construcción

Desarrollo

Capacitación de Programadores

Especificación de las necesidades de Formularios

Desarrollo de los Programas

Documentación para el desarrollo o construcción de un proyecto.

#### Diseño de la migración y carga inicial de datos

Especificación del Entorno de Migración

Diseño de Procedimientos de Migración y Carga Inicial

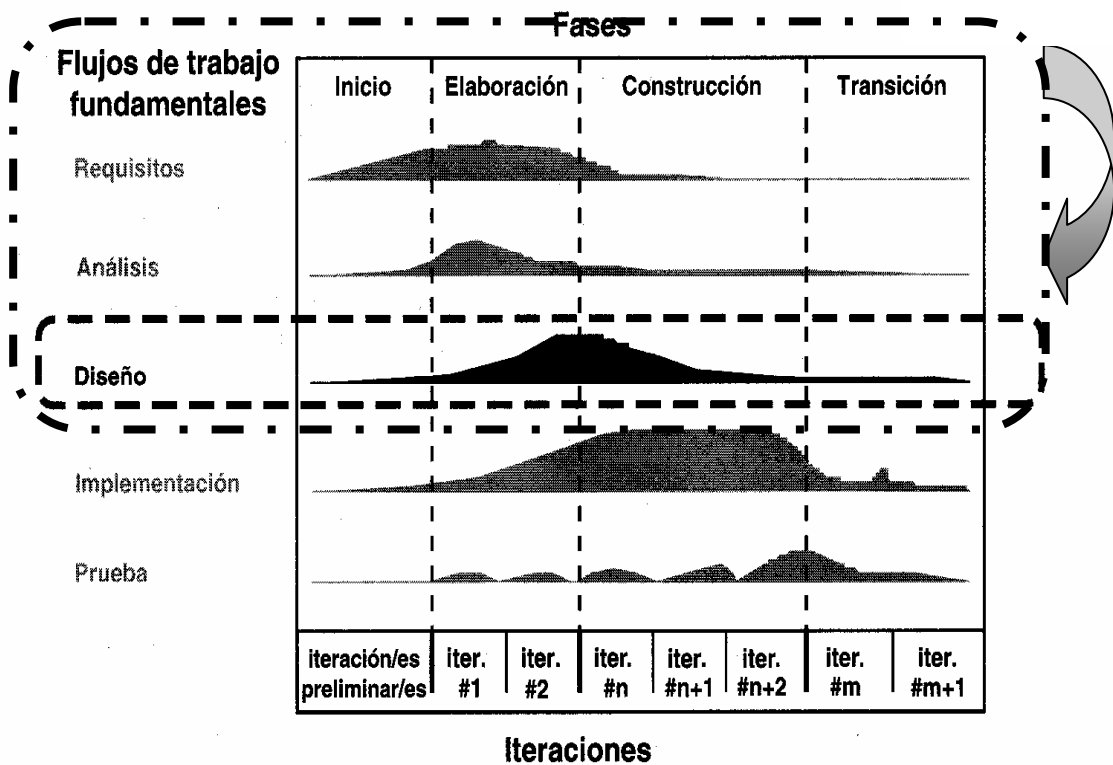
Diseño Detallado de Componentes de Migración y Carga Inicial

#### Autoevaluación

## Mapa Conceptual

# Proceso Unificado de desarrollo de Software

## Ciclo de Vida



## Introducción

El modelado es una parte central de todas las actividades que conducen a la producción de buen software. Construimos modelos para comunicar la estructura deseada y el comportamiento de nuestro sistema. Construimos modelos para visualizar y controlar la arquitectura del sistema. Construimos modelos para comprender mejor el sistema que estamos construyendo, muchas veces descubriendo oportunidades para la simplificación y reutilización. Construimos modelos para controlar el riesgo.

En esta unidad, construimos modelos y diagramas para una propuesta de desarrollo de software particular: El Complejo Cineclub.

## Caso de Estudio

### *Premisas del caso*

El caso de estudio presentado es una simplificación de un caso real, del que se tomaron algunos procesos de negocio, que se consideraron de interés, para destacar y aplicar conceptos teóricos vistos en el módulo de Sistemas III y Sistemas IV.

Desde esa perspectiva hay funciones que no se modelan para nada, aun siendo perfectamente posible hacerlo. Otra razón para tal simplificación es la extensión que de otra manera tendría la resolución del ejercicio.

Se seleccionaron aspectos a modelar buscando que cada una de los conceptos a destacar, esté presente al menos una vez.

Cada uno de los modelos que integran la solución, es en este sentido plausible de ser ampliado o modelado de otras formas.

Se trabaja el caso de estudio desde el primer flujo de trabajo (negocio) para poder lograr la trazabilidad de las herramientas utilizadas. De esta manera se podrá derivar luego en los modelos y diagramas correspondientes al Análisis y Diseño, que es el objeto de nuestro estudio.

Se Incorporarán aclaraciones a los modelos que se consideran de interés para la comprensión de los mismos.

### *Enunciado y Resolución de un caso*

La empresa Todo Viaje se dedica a brindar servicio de transporte diferencial puerta a puerta, ofreciendo comodidad, seguridad y puntualidad, para lograr satisfacer las necesidades de sus clientes. Además de realizar el envío de encomiendas, como un servicio adicional a sus clientes.

Es una empresa unipersonal, que nació hace ya 5 años en la ciudad de Río Segundo, en donde contaba con dos unidades, y el recorrido original era Río Segundo – Córdoba; dos años después se amplió su recorrido hacia la ciudad de Pilar. Es así como la empresa incrementó su número de unidades llegando a tener una flota de 10 coches, en consecuencia también se amplió el número de empleados a 36, distribuidos en los diferentes departamentos incluyendo a los choferes, y estableciendo las distintas agencias que están funcionando actualmente. Finalmente, se radicó en Córdoba, estableciendo la administración, y dispuso cuatro agencias de ventas ubicadas en distintos lugares: Agencia Córdoba Capital. Agencia Toledo. Agencia Río Segundo. Agencia Pilar.

La estructura que posee es la siguiente:

El mando superior es ejercido por un *Gerente General* encargado de la toma de decisiones estratégicas, define las políticas en la empresa, realiza la supervisión de las agencias y de todas las áreas a su cargo, haciendo conocer al personal las normas establecidas y responsabilidades pertinentes.

De él dependen los siguientes departamentos:

*Administración de Personal:* se ocupa de la liquidación de sueldos y jornales, de motivar al personal para que se sienta parte de la empresa, conocer sus problemas e intentar darles solución para que se sientan cómodos en su lugar de trabajo.

*Departamento Servicios:* conformado por los choferes encargados de realizar el transporte de pasajeros y rendir la venta de boletos efectuada a los pasajeros que suben durante el recorrido confeccionando la planilla de boletos vendidos.

*Tesorería:* es quien recibe las rendiciones de los choferes y de las agencias sobre la venta de boletos y envío de encomiendas. Paga al personal y maneja los fondos de la empresa. Además paga las compras efectuadas a los proveedores de los distintos insumos, y al taller mecánico por las reparaciones de las unidades.

*Compras:* realiza el trato con proveedores manteniendo actualizados sus datos, genera las órdenes de pedido según insumos necesarios y controla que sean entregados en tiempo y forma.

*Departamento Ventas:* controla el accionar de las agencias y realiza convenios con instituciones que solicitan el servicio de transporte para grupos o contingentes puntualmente, asignando unidades disponibles. Además asigna los distintos horarios a cubrir por los distintos choferes.

De ella dependen las agencias: Córdoba Capital, Toledo, Río Segundo, Pilar

Las funciones de todas las agencias son las siguientes: realizar las reservas y ventas de boletos, recibir las encomiendas a enviar y realizar su respectivo cobro, controlar el cumplimiento de los horarios de los choferes, realizar las solicitudes de insumos a compras, además de rendir a Tesorería los ingresos por los distintos conceptos de venta de boletos y envío de encomiendas.



Además cuenta con un Staff Contable que realiza el asesoramiento correspondiente y lleva los libros reglamentarios.

En lo que respecta al mantenimiento de las unidades, tienen un convenio firmado con el taller mecánico que realiza esta función. Cuando se ha producido un desperfecto de los coches se encarga de su reparación y entrega las unidades reparadas, remitiendo su correspondiente factura.

A continuación se describe el proceso a considerar para su estudio:

Trimestralmente se establecen los horarios a cubrir según lo autorizado por el ente regulador de transporte. Y se asigna mensualmente los choferes que cubrirán cada uno de los horarios y con qué unidades se harán.

En cada agencia el cliente efectúa la reserva personalmente o telefónicamente, en ningún caso se deja seña, y la tolerancia para presentarse y efectuar el viaje es de diez minutos previos al horario de partida, para ello se pide nombre, apellido y teléfono al cliente por cualquier novedad. Cuando el cliente se presenta a realizar el viaje, ya sea con reserva o no, se le efectúa el cobro correspondiente, aceptando como única forma de pago al contado, asignándole un número de asiento si se encuentra en una agencia en donde se inicia el recorrido, y sino se comunican por radio con el chofer de la unidad y se consulta la disponibilidad de asientos, de ser así se le reserva uno, comunicándole al chofer el hecho.

Durante el recorrido el chofer puede subir pasajeros siempre y cuando tenga lugar, respetando las reservas ya efectuadas hasta ese momento. Durante dicho recorrido y hasta el final va asentando en una planilla las ventas realizadas en el mismo. Al final de cada recorrido le rinde a Tesorería el dinero y la planilla con la venta de boletos. Cada agencia envía el detalle de cobros de sus ventas por ventanillas y de encomiendas, con el chofer que cubre cada horario del día, para que lo entregue junto a las planillas propias del chofer.

Cuando un coche se rompe, el chofer informa inmediatamente por radio a las respectivas agencias, de manera de enviar el refuerzo previsto para estos casos, y se comunican con el auxilio del taller mecánico para su reparación, el cual se encargará del problema, una vez que la unidad está en condiciones se comunica esto a Departamento Ventas para que sea considerada su puesta en circulación, y se entrega la factura que se remite a Tesorería para su posterior pago.

El encargado de Tesorería verifica cada mes las facturas pendientes de pago con el taller mecánico y emite los cheques para su correspondiente pago.

Cuando un cliente se presenta para enviar una encomienda, se registra el destino, se cobra lo que corresponde y se deja dicho paquete en espera hasta que pase el coche del próximo horario para su envío correspondiente. Cuando el chofer llega al final del recorrido entrega el paquete en la agencia para su posterior retiro.

En forma semanal la Tesorería efectúa un informe con los ingresos y egresos para rendir a Gerencia los movimientos realizados durante una semana.

El Gerente General ha observado algunos problemas en la empresa referidos a que las planillas de venta de boletos de las agencias y/o choferes no están bien confeccionadas, ya sea con datos incompletos o erróneos, lo que no permite a Tesorería realizar un buen control de las rendiciones correspondientes. Además, la asignación de horarios no está bien realizada ya que ocurre a veces que un chofer tiene asignados horarios superpuestos, esto acarrea que se demoren en el cumplimiento de los horarios respectivos. A esto se agrega el hecho de que el Gerente no cuenta con información resumida y estadística de los viajes realizados, y de las rendiciones pertinentes, para una buena toma de decisiones y para conocer el accionar de la empresa.

Es por ello que desea contratar los servicios de un grupo de profesionales de sistemas de manera que le ofrezca una propuesta de solución a la situación planteada, además de estar interesado en automatizar los procesos de información previendo un presupuesto para la adquisición del equipamiento necesario.

Consideren que Uds. son ese grupo de profesionales, y teniendo en cuenta los requerimientos y problemas de información, plantearemos a continuación una propuesta de solución que incluya el desarrollo de los siguientes modelos:

1. Modelado del Sistema de Negocio
  - 1.1. Modelo de Caso de uso del Sistema de Negocio (diagrama descripciones de los casos de uso).
  - 1.2 Descripción de objetos activos trabajadores.
2. Modelado del Sistema de Información
  - 2.1. Modelo de casos de uso (diagrama de caso de uso y descripción de cada caso de uso).
  - 2.2. Modelo de Objetos del Dominio.
  - 2.3. Descripción o Prototipo de Interfaz.
3. Flujo de trabajo del Análisis
  - 3.1. Diagrama de Colaboración.
  - 3.2. Subsistemas.
4. Flujo de trabajo de Diseño
  - 4.1 Diagrama de Despliegue
  - 4.2 Ambiente de implementación:
  - 4.3 Diagrama de Clases de Diseño.
  - 4.4 Diagrama de Estados.

## Propuesta

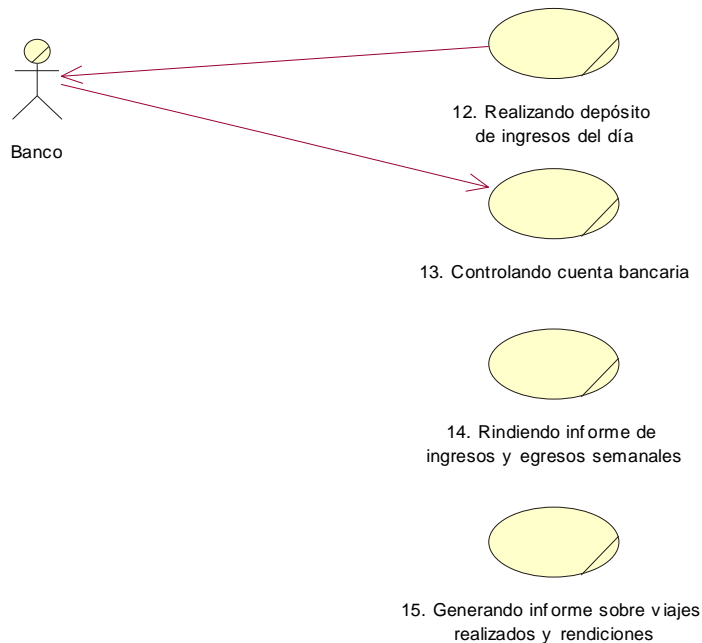
Nuestro sistema busca automatizar los procesos de venta, rendición de venta de pasajes y encomiendas, como así también la registración y generación de informes.

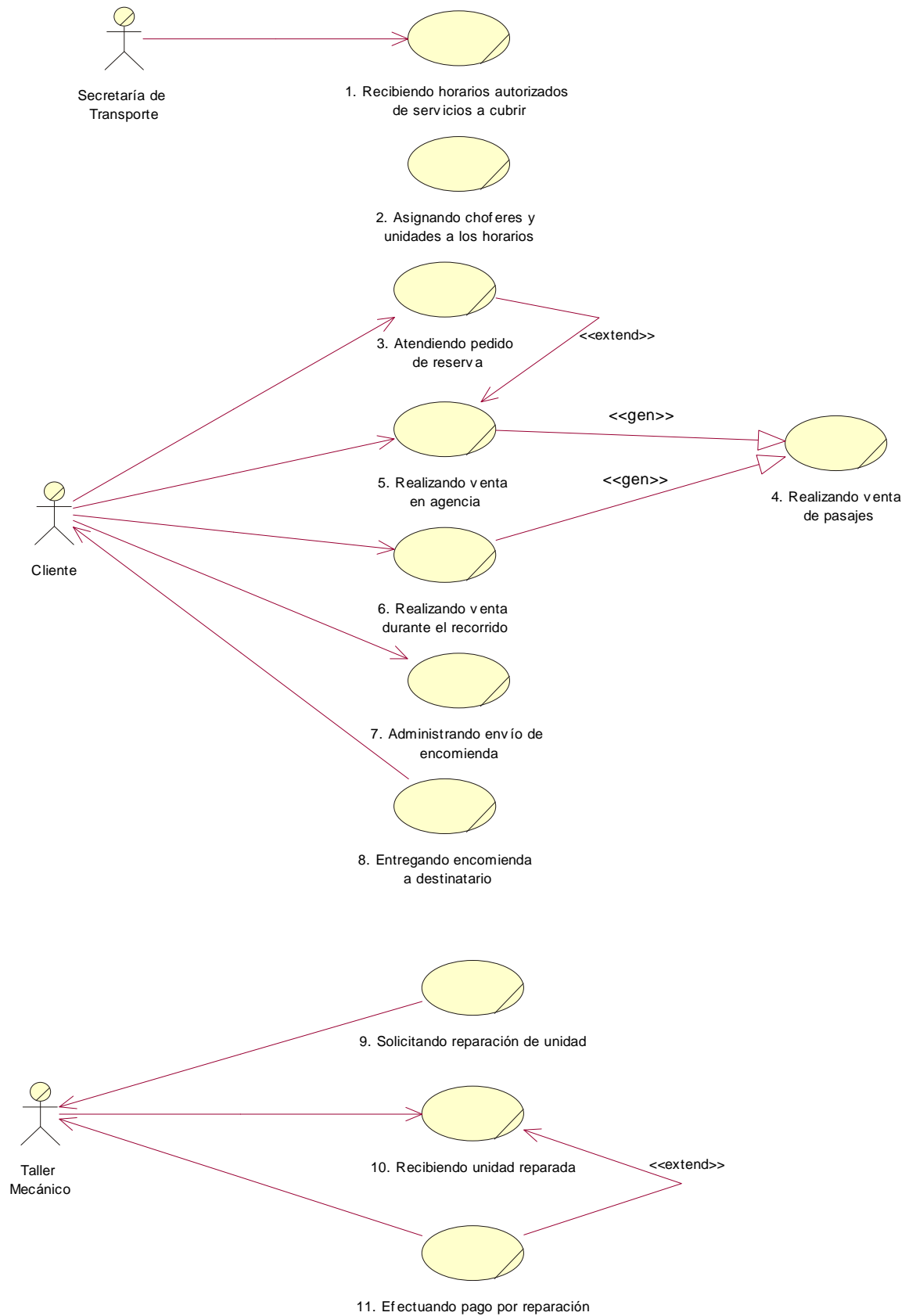
Es por ello que planteamos la instalación de una red entre las agencias, las cuales contarán con una terminal en línea y demás equipos necesarios para realizar la tarea requerida. Se contará además con un equipamiento mayor en la agencia de Córdoba Capital, donde serán los departamentos de Ventas, Tesorería y el Gerente General quienes tendrán acceso directo al sistema informático.

## I. Modelado del Sistema de Negocio

- 1.1. Diagrama de casos de uso del Sistema de Negocio
- 1.2. Descripciones de los casos de uso.
- 1.3. Descripción de objetos activos trabajadores.

### ***1.1. Diagrama de Casos de Uso del Sistema de Negocios***





## 1.2. Descripción de casos de uso del Sistema de Negocios

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso		<b>Recibiendo horarios autorizados de servicios a cubrir</b>	Nro. de Orden: 1
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal		<b>Secretaría de Transporte</b>	Actor Secundario:
Tipo de Caso de uso:		<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Recibir y registrar los horarios de servicios autorizados por el Ente Regulador para cubrir los distintos recorridos con que cuenta la empresa.</i>			
Precondiciones:			
Post-Condiciones: <i>Se han registrado los nuevos horarios de los servicios de la empresa.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el empleado del Departamento Ventas recibe, en forma trimestral, los horarios autorizados por la Secretaría de Transporte.  2. El empleado del Departamento Ventas consulta los horarios de los servicios del último trimestre, verificando la existencia de cambios. 3. Se consultan cuáles son las unidades disponibles en el próximo trimestre. Estas unidades han sido previamente registradas por el empleado del Departamento Ventas. 4. De acuerdo a esto, se determinan los horarios del próximo trimestre. 5. El empleado registra los nuevos horarios de servicios a brindar en el próximo trimestre. 6. Se envía al Gerente General la notificación acerca de los horarios del próximo trimestre.  <div style="text-align: center;">Termina U-C</div>			
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso		<b>Asignando choferes y unidades a los horarios</b>	Nro. de Orden: 2
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal		Actor Secundario:	
Tipo de Caso de uso:		<input type="checkbox"/> Concreto	<input checked="" type="checkbox"/> Abstracto
Objetivo: <i>Determinar, de forma mensual, cuáles serán los choferes y las unidades que serán asignadas a cada uno de los horarios.</i>			
Precondiciones:			
Post-Condiciones: <i>Se han asignado choferes y unidades a los servicios.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el empleado del Departamento Ventas, mensualmente, consulta los horarios de los servicios que brinda la empresa. 2. Se consultan los horarios disponibles de cada uno de los choferes de la empresa. Estos datos han sido registrados por el mismo empleado en su debido momento. 3. El empleado del Departamento Ventas consulta las unidades disponibles en el próximo mes. Los datos correspondientes a estas unidades han sido registrados con anterioridad por este mismo empleado. 4. Con los datos obtenidos, el empleado realiza la asignación de los choferes y las unidades a los distintos horarios. 5. Se registra, para cada horario, el chofer y la unidad asignada. 6. Se informa a los choferes los horarios de trabajo del próximo mes.			
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso	<b>Atendiendo pedido de reserva</b>	Nro. de Orden: 3
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Cliente</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Atender el pedido de reserva, y de ser necesario, llevar a cabo la misma.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha receptado y eventualmente realizado la reserva de el/los pasaje/s correspondientes de acuerdo a las preferencias del cliente.</i>		
Curso Normal	Alternativas	
<p>1. Comienza cuando un cliente realiza una solicitud de reserva, ya sea en forma personal o telefónicamente al empleado de la agencia.</p> <p>2. El empleado de agencia consulta al cliente el horario, fecha y recorrido para el cual desea realizar la reserva.</p> <p>3. Si es una agencia de inicio, el empleado de agencia consulta de inmediato la disponibilidad de pasajes para las condiciones establecidas.</p> <p>4. Si hay asientos disponibles, se solicita al cliente nombre, apellido y teléfono, además del lugar donde subirá al coche.</p> <p>5. Se registran los datos del cliente que realizó la reserva y los asientos del coche como reservados.</p> <p style="text-align: right;">Termina U-C</p>	<p>3.1. Si es una agencia intermedia, el empleado de agencia verifica si el coche de ese horario ya inició el recorrido.</p> <p>3.1.1. Si no se inició el recorrido, se verifica en la agencia la disponibilidad de asientos.</p> <p>3.1.2. Si ya se inició el recorrido, el empleado de agencia se comunica con el chofer y consulta si hay asientos disponibles.</p> <p>4.1. Si no hay asientos disponibles, ya sea en agencia o en recorrido, se informa al cliente de la situación. Termina U-C</p> <p>5.1. Si es una agencia intermedia y el recorrido ya se inició, se informa al chofer de la reserva, el destino y nombre del cliente, como así también el lugar donde subirá.</p> <p>5.2. El chofer registra la reserva en la planilla de ventas que posee.</p> <p style="text-align: right;">Termina U-C</p>	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende: Realizando venta en agencia		
Caso de uso de Generalización:		

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Realizando venta de pasajes</b>		Nro. de Orden: 4	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal		Actor Secundario:	
Tipo de Caso de uso:		<input type="checkbox"/> Concreto	<input checked="" type="checkbox"/> Abstracto
Objetivo: <i>Efectuar la venta de pasajes, y registrar la misma.</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha producido y registrado la venta, a la vez que se ha entregado al cliente el comprobante de pago y el boleto correspondiente.</i>			
Curso Normal		Alternativas	
A. Comienza cuando el cliente se presenta para realizar el viaje. B. El empleado consulta al cliente si ha realizado alguna reserva. C. Si el cliente no cuenta con reserva, el empleado averigua acerca del destino en el cual desea realizar el viaje. D. Si hay asientos disponibles, el empleado consulta el precio del pasaje. E. El empleado realiza el cobro. F. Se entrega el comprobante de pago y el boleto al cliente. G. Se registra la venta.		C.1. Se verifican los datos personales del cliente. C.2. Se consulta cuál fue la reserva efectuada. D.1. De no haber asientos disponibles, se informa al cliente de la situación. Fin U-C	
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			



Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso	<b>Realizando venta en agencia</b>	
Nro. de Orden:	5	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Cliente</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Realizar la venta de pasajes por ventanilla.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha vendido el/los pasaje/s y se ha efectuado el cobro por ventanilla.</i>		
Curso Normal	Alternativas	
1. = A 2. = B 3. = C  4. Se averigua el horario en el cual el cliente quiere viajar. 5. Si es una agencia de inicio, el empleado verifica la disponibilidad de asientos, consultando las ventas y reservas efectuadas.  6. = D  7. = E 8. = F 9. = G 10. Si es una agencia de inicio, el empleado asigna un número de asiento.  11. Si es una agencia de inicio, el empleado emite y entrega al chofer la planilla de venta de boletos y las reservas de asientos, a tener en cuenta antes de iniciar el recorrido. Termina U-C	3.1. Si el cliente cuenta con reserva, el empleado de agencia controla que el cliente haya concurrido con diez minutos de anticipación. 3.1.1. Si no concurrió en el horario establecido, se informa al cliente que ha perdido la reserva. 3.1.2. Se registra el asiento como disponible. 3.2. = C.1 3.3. = C.2 3.4. = E 3.5. = F Termina U-C  5.1. Si es una agencia intermedia, el empleado de agencia consulta si el coche de ese horario ya salió de la agencia anterior. 5.2. Si ya se inició el recorrido, el empleado de agencia se comunica con el chofer. 5.2.1. Si aun no se inició el recorrido, el empleado de agencia consulta si hay asientos disponibles. 5.3. Se consulta al chofer la disponibilidad de asientos. 5.4. El chofer consulta la planilla de ventas e informa al empleado si hay o no asientos desocupados. 6.1. = D.1 6.2. Si el cliente lo desea, se procede a realizar una reserva en un próximo horario. Se llama al U-C "Recibiendo pedido de reserva". Termina U-C  10.1. Si es una agencia intermedia, y el coche ya inició el recorrido, el empleado de agencia informa al chofer de la nueva reserva. 10.2. El chofer registra en la planilla el asiento como ocupado. Termina U-C	
Asociaciones de Extensión: Recibiendo pedido de reserva		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización: Realizando venta de pasajes		

Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre Caso de U. <b>Realizando venta durante el recorrido</b>	Nro. de Orden: 6	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal <b>Cliente</b>	Actor Secundario:	
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Realizar la venta de pasajes a los clientes durante el recorrido.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha realizado la venta de pasajes durante el recorrido.</i>		
Curso Normal	Alternativas	
1. = A 2. = B 3. = C  4. El chofer del vehículo consulta si hay asientos disponibles, viendo en la planilla de venta y reservas los lugares ocupados. 5. = D  6. = E 7. El chofer registra en la planilla el cobro y el asiento que se ocupó 8. = F 9. El chofer comunica a las agencias intermedias próximas acerca de la venta del pasaje y qué asiento se ocupó. 10. = G 11. Al llegar a la agencia, el chofer entrega al empleado de la agencia el dinero y la planilla de ventas realizadas en el recorrido. 12. El empleado de la agencia verifica las ventas y controla el dinero entregado por el chofer. Termina U-C	3.1. Si el cliente tiene reserva, el chofer solicita los datos al cliente. 3.2. = C.1 3.3. = C.2 3.4. = E 3.5. = F  Termina U-C  5.1. = D.1 5.2. Si el cliente lo desea, el chofer consulta al empleado de la agencia correspondiente si habrá asientos disponibles en un próximo horario. 5.3. De ser así, el chofer solicita los datos al cliente e informa a la agencia de la nueva reserva. Termina U-C	
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización: <b>Realizando Venta de Pasajes</b>		

Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso	<b>Administrando envío de encomienda</b>	Nro. de Orden: 7
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Cliente</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Recibir, almacenar y realizar el traslado de la encomienda a destino.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha enviado la encomienda al destino correspondiente.</i>		
Curso Normal		Alternativas
1. Comienza cuando el cliente se presenta a la agencia para enviar una encomienda. 2. El empleado de la agencia consulta al cliente sobre cuál es el destino a donde desea enviar la encomienda.  3. El empleado de agencia recibe el paquete y pesa el mismo. 4. Se realiza el cálculo del precio del envío, de acuerdo al precio del kilómetro y el precio del kilogramo registrados anteriormente por el empleado del Departamento Ventas. 5. Se registra el destino de la encomienda, junto con los datos del destinatario y de quien lo envía. 6. Se realiza y registra el cobro. 7. Se entrega al cliente el comprobante de pago. 8. Se registra la agencia de destino en el paquete. 9. Se deja el paquete en espera hasta que pase el próximo coche. 10. Al llegar el coche a la agencia, el empleado entrega al chofer el listado de encomiendas (detallando destino, destinatario y remitente de la encomienda) y los paquetes correspondientes. 11. El chofer controla el listado y los paquetes que debe transportar. 12. El chofer traslada los paquetes de encomienda a destino, y llega a la agencia destino con el paquete de encomienda. 13. El chofer entrega al empleado de la agencia el paquete junto al listado de encomienda. 14. El empleado de la agencia verifica los datos del destinatario y encomienda, y guarda el paquete. 15. El empleado de la agencia registra que el paquete llegó a destino, entregando después el listado al chofer. Termina U-C		2.1. Si el destino no es cubierto por la empresa, se informa al cliente. Termina U-C
Asociaciones de Extensión:		
Asociaciones de Inclusión:		

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del CU <b>Entregando encomienda a destinatario</b>		Nro. de Orden: 8	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal	<b>Cliente</b>		Actor Secundario:
Tipo de Caso de uso:		<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Entregar la encomienda a destinatario.</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha entregado el paquete de encomienda al destinatario.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el cliente destinatario se presenta a la agencia para retirar la encomienda, el empleado de la agencia solicita los datos a la persona y el lugar de origen del envío. 2. Se verifica si el paquete se encuentra en la agencia.  3. Se entrega el paquete al cliente. 4. Se hace firmar el comprobante de entrega al cliente. 5. Se registra la entrega del paquete. Termina U-C		2.1. De no hallarse el paquete, se realiza una consulta a la agencia de origen. 2.2. Se informa al cliente acerca del estado del envío.  Termina U-C	
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso	<b>Solicitando reparación de unidad</b>	Nro. de Orden: 9
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Taller Mecánico</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Realizar, ante el desperfecto en la unidad, el aviso correspondiente a la agencia y solicitar la reparación al taller mecánico.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha asignado el refuerzo y se ha informado al auxilio del taller mecánico acerca de la rotura del coche.</i>		
Curso Normal	Alternativas	
1. Comienza cuando el chofer informa al empleado de la agencia más cercana acerca de la rotura del coche. 2. El empleado de la agencia consulta las unidades y choferes disponibles. 3. El empleado de la agencia asigna la unidad de refuerzo y el chofer a cargo. 4. El empleado informa al chofer del refuerzo sobre la situación. 5. Se registran los datos de la unidad que sufrió el desperfecto y el lugar donde se produjo. 6. Se comunica al auxilio mecánico del taller acerca de la unidad rota. 7. El empleado de la agencia consulta los horarios de la unidad rota para así reasignar estos horarios a otra unidad. 8. Cuando la unidad de refuerzo llega al lugar, se realiza el traspaso de los pasajeros y las encomiendas, junto con los listados de encomiendas y planilla de ventas y reservas. Termina U-C	6.1. Si es la primera vez que se solicita el servicio del taller, se registran los datos del mismo.	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Recibiendo unidad reparada</b>		Nro. de Orden: 10	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal	<b>Taller Mecánico</b>		Actor Secundario:
Tipo de Caso de uso:		<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Recibir la unidad reparada y la factura correspondiente al servicio realizado por el taller.</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha puesto a disposición de la empresa la unidad reparada.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el taller mecánico notifica al empleado del Departamento Ventas que la unidad se encuentra reparada. 2. El empleado del Departamento Ventas solicita al taller mecánico el envío de dicha unidad reparada. 3. El empleado de Ventas controla que la unidad ingrese a la empresa. 4. Se registra como disponible la unidad recibida y reparada. 5. El empleado de Ventas recibe la factura por la reparación realizada. 6. Se envía dicha factura al encargado de Tesorería. 7. El encargado de Tesorería registra los datos de la factura y su próxima fecha de pago. Termina U-C		7.1. Si la fecha corriente se corresponde con el final del mes, se llama al U-C “Efectuando pago por reparación” Termina U-C	
Asociaciones de Extensión: Efectuando pago por reparación			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso	<b>Efectuando pago por reparación</b>	Nro. de Orden: 11
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Taller Mecánico</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Realizar el pago correspondiente a las reparaciones efectuadas por el taller mecánico.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha efectuado el pago pendiente, emitiendo el cheque correspondiente al taller mecánico.</i>		
Curso Normal	Alternativas	
1. Comienza cuando el encargado de Tesorería, al finalizar el mes, consulta las facturas pendientes de pago al taller mecánico. 2. El encargado de Tesorería consulta los datos del taller mecánico. 3. Se calcula el monto total por el cual se debe emitir el cheque. 4. Se emite el cheque al taller mecánico por el total de la deuda. 5. El encargado de Tesorería registra los datos del cheque. 6. Se entrega el cheque al taller mecánico. 7. Se actualiza el estado de la deuda con el taller.  Termina U-C	1.1. Si no existen facturas pendientes de pago Termina U-C	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende: Recibiendo unidad reparada		
Caso de uso de Generalización:		

Nivel del Caso de uso	<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del CU	<b>Realizando depósito de ingresos del día</b>	Nro. de Orden: 12
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Banco</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Contabilizar el dinero que ingresó a cada agencia en el día y verificar si corresponde con lo registrado.</i>		
Precondiciones:		
Post-Condiciones: <i>Se han controlado los ingresos del día de cada agencia y se ha realizado el depósito correspondiente.</i>		
Curso Normal	Alternativas	
1. Comienza cuando el empleado de agencia, al final del día, verifica los ingresos de dinero de la jornada. 2. El empleado de la agencia consulta los ingresos de dinero por venta de pasajes en ventanilla y durante el recorrido. 3. Se consulta el monto de dinero que ingresó por envío de encomiendas. 4. Se calcula el monto total de dinero que ingresó por todo concepto. 5. El empleado cuenta el dinero que posee en caja en ese momento. 6. Se controla si el dinero en caja coincide con el monto establecido. 7. Se confecciona la boleta de depósito correspondiente al monto determinado. 8. Se realiza el depósito de dinero que ingresó en el día. 9. Se registra monto, fecha y datos de la boleta de depósito. 10. El empleado de la agencia envía el comprobante de depósito al encargado de Tesorería. 11. El encargado de Tesorería registra la recepción de la boleta de depósito. Termina U-C	6.1. Si existe diferencia alguna entre los montos, se registra la misma.	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		



Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Controlando cuenta bancaria</b>		Nro. de Orden: 13	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal <b>Banco</b>		Actor Secundario:	
Tipo de Caso de uso:		<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Recibir y registrar el estado de cuenta, además de controlar los depósitos realizados por cada uno de las agencias.</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha recibido el informe de estado de cuenta y efectuado el control de depósitos.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el encargado de Tesorería recibe, semanalmente, el estado de cuenta del banco. 2. El encargado de Tesorería consulta los montos diarios depositados por cada agencia. 3. Se verifica el estado de cuenta actual con los montos depositado diariamente y los pagos efectuados. 4. Se registra el estado de cuenta recibido. Termina U-C		2.1. Si es el primer estado de cuenta del mes, se consultan los pagos efectuados al taller en el mes anterior.  4.1. Si existe alguna diferencia, el encargado de Tesorería envía una notificación al Gerente General acerca de la situación. Termina U-C	
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Rindiendo informe con ingresos y egresos semanales</b>		Nro. de Orden: 14	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal		Actor Secundario:	
Tipo de Caso de uso:		<input type="checkbox"/> Concreto	<input checked="" type="checkbox"/> Abstracto
Objetivo: <i>Generar un informe acerca de los ingresos y egresos semanales de la empresa.</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha emitido el informe de rendición de ingresos y egresos.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el Gerente General solicita, a Tesorería, el informe semanal de ingresos y egresos. 2. El encargado de Tesorería recibe la solicitud del informe. 3. El encargado de Tesorería consulta los ingresos y egresos que se han producido en la semana. 4. Se consulta el estado de la cuenta del banco. 5. Se confecciona el informe solicitado, adjuntando el informe de estado de cuenta actual. 6. Se envía el informe al Gerente General. Termina U-C			
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso		<input checked="" type="checkbox"/> Negocio	<input type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Generando informe sobre viajes realizados y rendiciones</b>		Nro. de Orden: 15	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal		Actor Secundario:	
Tipo de Caso de uso:		<input type="checkbox"/> Concreto	<input checked="" type="checkbox"/> Abstracto
Objetivo: <i>Generar un informe mensual que contenga los datos acerca de las rendiciones efectuadas y los viajes realizados.</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha emitido el informe mensual de viajes realizados y rendiciones al Gerente General.</i>			
Curso Normal		Alternativas	
1. Comienza cuando el Gerente General, de forma mensual, solicita el informe de rendiciones por viajes y encomiendas. 2. El encargado de Tesorería recibe la notificación. 3. El encargado de Tesorería consulta los datos de los viajes realizados en el último mes. 4. Se consultan cuáles fueron las ventas realizadas y las encomiendas recibidas. 5. Se consultan los ingresos recibidos por cada concepto en el último mes. 6. Se realizan los cálculos correspondientes y se confecciona el informe de viajes y rendiciones. 7. Se envía el informe de viajes y rendiciones al Gerente General. Termina U-C			
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

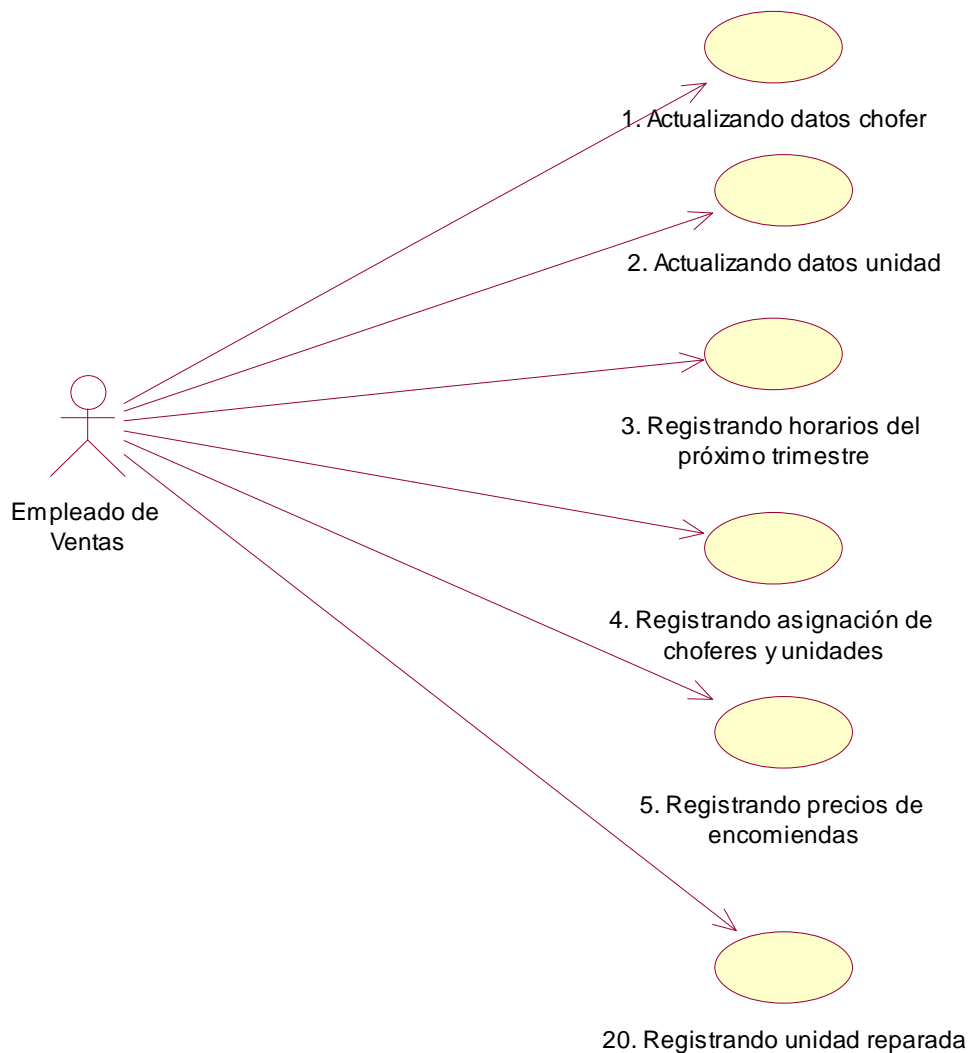
### ***1.3 Descripción de objetos activos trabajadores.***

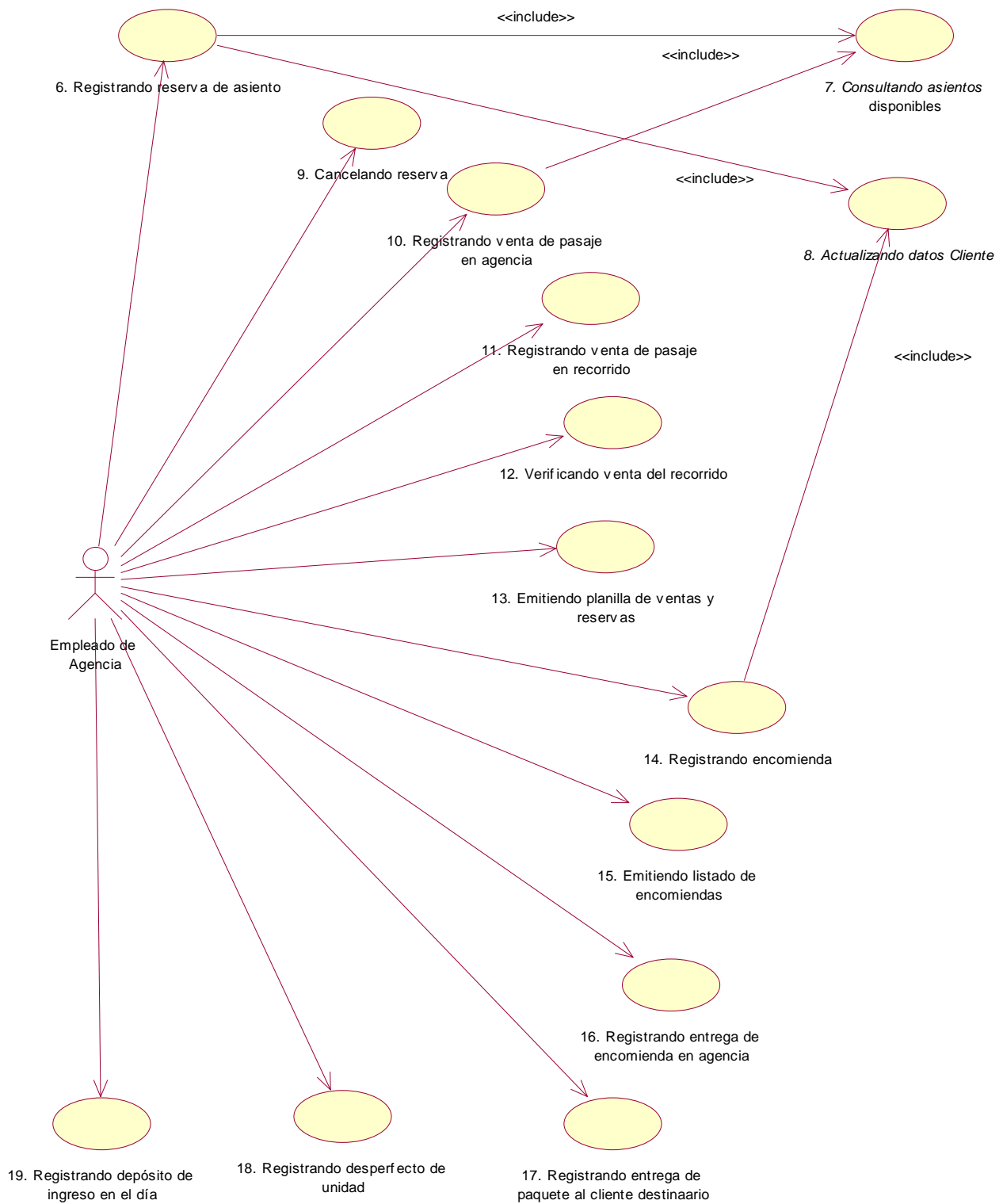
Los trabajadores identificados en el modelo de negocio,, coinciden con los actores del sistema de información , en sistemas intranet. En nuestro caso se identifica como trabajadores. El empleado de ventas, empleado de agencia, y el encargado de tesorería, en esta porción de sistema analizada.

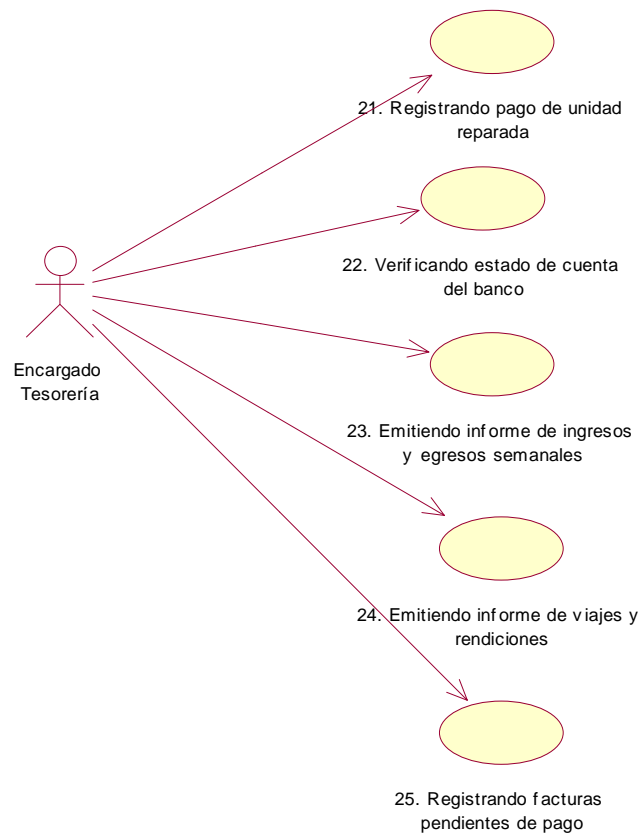
## **2. Modelado del Sistema de Información**

- 2.1. Diagrama de caso de uso
- 2.2. Descripción de cada caso de uso. Prototipo de Interfaz.
- 2.3. Modelo de Objetos del Dominio.

### ***2.1 Diagrama de Casos de Uso***







## 2.2 Descripción de cada Caso de Uso y Prototipos de Interfaz

A partir de este punto se han seleccionado un grupo de Casos de Uso y las descripciones y prototipos de interfaz se desarrollarán sobre dicho grupo, a modo de ejemplo, y en el resto de los flujos de trabajo también se desarrollarán los distintos modelos teniendo como referencia este grupo de casos de uso.

Los Casos de Uso considerados son los siguientes:

- U-C 1: Actualizando datos del chofer
- U-C 6: Registrando reserva de asiento
- U-C 7: Consultando asientos disponibles
- U-C 8: Actualizando datos de cliente
- U-C 9: Cancelando reserva
- U-C 10: Registrando venta de pasaje en agencia
- U-C 11: Registrando venta de pasaje en recorrido
- U-C 12: Verificando venta en recorrido
- U-C 13: Emitiendo planilla de ventas y reservas

Nivel del Caso de uso		<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Actualizando datos chofer</b>		Nro. de Orden: 1	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal	<b>Empleado de Ventas</b>		Actor Secundario:
Tipo de Caso de uso:		<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Registrar o actualizar los datos de cada chofer de la empresa.</i>			
Precondiciones:			
Post-Condiciones: <i>Se han actualizado los datos del chofer en cuestión y se ha asignado un número de legajo.</i>			
Curso Normal		Alternativas	
<p>1. Comienza cuando el empleado de Ventas necesita actualizar los datos de un chofer ingresando a la pantalla “Actualización de datos de Chofer”.</p> <p>2. El empleado de Ventas desea ingresar los datos de un nuevo chofer al sistema, selecciona la opción Nuevo Chofer</p>		<p>2.1. El empleado de Ventas necesita modificar los datos de un chofer, selecciona la opción Modificar.</p> <p>2.1.1. Se ingresa el número de legajo del chofer</p> <p>2.1.2. Se acepta el mismo, mostrando todos los datos personales del chofer.</p> <p>2.1.3. Se modifican los datos necesarios.</p> <p>2.1.4. Se selecciona la opción Aceptar.</p> <p>2.1.4.1. Si no se desea registrar los datos modificados se selecciona la opción Cancelar.</p> <p style="text-align: right;">Termina U-C</p> <p>2.1.5. Se muestra la confirmación de la modificación.</p> <p style="text-align: right;">Termina U-C</p> <p>2.2. El empleado de Ventas necesita dar de baja a un chofer del sistema, selecciona la opción Eliminar</p> <p>2.2.1. Se ingresa el número de legajo del chofer.</p> <p>2.2.2. Se acepta el mismo, mostrando los datos personales del chofer.</p> <p>2.2.3. Se ingresa la fecha de egreso y el motivo por el cual se produce la baja.</p> <p>2.2.4. Se selecciona la opción Aceptar.</p> <p>2.2.4.1. Si no se desea registrar los datos ingresados se selecciona la opción Cancelar.</p> <p style="text-align: right;">Termina U-C</p> <p>2.2.5. Se muestra la confirmación de la baja, el chofer se considera inactivo.</p> <p style="text-align: right;">Termina U-C</p>	
<p>3. El empleado de Venta ingresa el apellido y nombre, dirección, teléfono, fecha de nacimiento y fecha de ingreso.</p> <p>4. Al momento de ingresar el tipo de documento se muestra una lista desplegable de opciones, de la cual se deberá seleccionar una de ellas.</p>			

5. Se ingresa el número de documento, y en una grilla el día de la semana, el horario de entrada y el horario de salida que poseerá el chofer para cada día

6. Se selecciona la opción Aceptar.

7. Se actualiza el campo legajo luego de Aceptar.

8. Se muestra un mensaje con el número de legajo asignado y la confirmación de la registración de los datos del chofer.

Termina U-C

6.1. Si no se desean registrar los datos del chofer, se selecciona la opción Cancelar.

Termina U-C

**Actualizacion de Datos de Chofer**

Datos Chofer

Nro Legajo  Fecha Nacimiento

Apellido  Tipo Documento - Numero

Nombre  Fecha Ingreso

Direccion  Fecha Egreso

Telefono  Motivo Egreso

Horario de Trabajo

	Dia	Desde	Hasta

Nuevo Chofer    Modificar    Eliminar    Aceptar    Cancelar

**Reserva de Asiento**

Datos Viaje

Origen  Lugar Ascenso

Destino  Nro Reserva

Fecha

Hora

Datos Pasajero

Apellido

Nombre

Telefono

Asientos

	Numero	Precio

Total

Consultar Asientos    Registrar Cliente    Aceptar    Cancelar



Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Registrando reserva de asiento</b>		Nro. de Orden: 6
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal	<b>Empleado de Agencia</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Registrar la reserva de pasaje realizada por el cliente</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha llevado a cabo la registración de la reserva del asiento correspondiente y se ha asignado un número de reserva</i>		
Curso Normal	Alternativas	
1. Comienza cuando el empleado de Agencia necesita registrar una reserva ingresando a la pantalla "Reserva de asiento". 2. Se selecciona la opción Consultar Asientos, se llama al caso de uso "Consultando asientos disponibles". 3. Se actualizan los campos: origen , destino, fecha, hora, número(s) de asiento(s), precio del pasaje y precio total. 4. Se selecciona la opción Registrar cliente, se llama al caso de uso "Registrando Cliente". 5. Se actualizan los campos nombre , apellido y teléfono del cliente que está realizando la reserva. 6. Se registra el lugar donde sube el cliente. 7. Se selecciona la opción Aceptar. En este momento se asigna un número de reserva, el cual aparece en el campo "Nº de Reserva". Termina U-C	7.1. Si no se desea registrar la reserva correspondiente, se selecciona la opción Cancelar. Termina U-C	
Asociaciones de Extensión:		
Asociaciones de Inclusión:	7. Consultando datos disponibles 8. Registrando cliente	
Caso de uso donde se incluye:		
Caso de uso donde se extiende: 10. Registrando venta de pasaje		
Caso de uso de Generalización:		

Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Consultando asientos disponibles</b>		Nro. de Orden: 7
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal		Actor Secundario:
Tipo de Caso de uso:		<input type="checkbox"/> Concreto <input checked="" type="checkbox"/> Abstracto
Objetivo: <i>Consultar la disponibilidad de asientos para las condiciones establecidas</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha consultado la disponibilidad de asientos del colectivo</i>		
Curso Normal	Alternativas	
1. Comienza cuando se necesita consultar la disponibilidad de asientos ingresando a la pantalla “Asientos Disponibles”. 2. Se selecciona de una lista el origen/destino. 3. Se selecciona de otra lista adjunta el día y horario para el cual se desea realizar la reserva. 4. Se selecciona la opción Consultar, llamando a la pantalla “Asientos”. 5. Se muestra la disposición de los asientos en el colectivo correspondiente a ese día y horario. 6. Se selecciona el o los asientos requeridos.          7. Se retorna a la pantalla “Asientos Disponibles” y se selecciona la opción Aceptar, inhabilitando así la disponibilidad del o los asientos, a la vez que se muestra un mensaje “El o los asientos asignados son...” 8. Seleccionamos la opción Aceptar de la pantalla principal Termina U-C	6.1. De no haber asientos disponibles, se selecciona la opción Aceptar, con lo cual se muestra un mensaje de “Colectivo Lleno” Termina U-C 6.1.1. De no estar de acuerdo con la selección del o los asientos, se elige la opción Cancelar. Termina U-C          8.1. Si se desea no asignar los asientos consultados, se selecciona la opción Cancelar. Termina U-C	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye: 6. Registrando reserva de asientos 10. Registrando venta de pasaje en agencia		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

**Asientos Disponibles**

Localidad

	Origen	Destino
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		

Horario

	Día	Hora Partida	Hora Llegada
<input type="checkbox"/>			
<input type="checkbox"/>			
<input type="checkbox"/>			
<input type="checkbox"/>			
<input type="checkbox"/>			
<input type="checkbox"/>			
<input type="checkbox"/>			
<input type="checkbox"/>			

**Asientos**

Datos Horario

Unidad

Día

Horario

Datos Asientos

<input type="checkbox"/> 1	<input type="checkbox"/> 5	<input type="checkbox"/> 9	<input type="checkbox"/> 13	<input type="checkbox"/> 17	<input type="checkbox"/> 21
<input type="checkbox"/> 2	<input type="checkbox"/> 6	<input type="checkbox"/> 10	<input type="checkbox"/> 14	<input type="checkbox"/> 18	<input type="checkbox"/> 22
<input type="checkbox"/> 3	<input type="checkbox"/> 7	<input type="checkbox"/> 11	<input type="checkbox"/> 15	<input type="checkbox"/> 19	<input type="checkbox"/> 23
<input type="checkbox"/> 4	<input type="checkbox"/> 8	<input type="checkbox"/> 12	<input type="checkbox"/> 16	<input type="checkbox"/> 20	<input type="checkbox"/> 24

Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Actualizando datos Cliente</b>		Nro. de Orden: 8
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal		Actor Secundario:
Tipo de Caso de uso:		<input type="checkbox"/> Concreto <input checked="" type="checkbox"/> Abstracto
Objetivo: <i>Registrar y actualizar los datos de los clientes</i>		
Precondiciones:		
Post-Condiciones: <i>Se han registrado y/o actualizado los datos del cliente</i>		
Curso Normal		Alternativas
1. Comienza cuando es llamado por otro caso de uso y se ingresa a la pantalla “ Actualización de Datos del Cliente”. 2. Se ingresa tipo y número de documento del cliente. 3. De no existir el cliente, se muestra un mensaje “El cliente no existe”, ante lo cual se permite el ingreso de los datos personales.  4. Si se desea, se ingresa: apellido, nombre, teléfono dirección, tipo de documento seleccionando de una lista desplegable y número de documento. 5. Se selecciona Aceptar, ante lo cual aparece un mensaje “Se ha registrado un nuevo cliente” Termina U-C		3. Se muestran los restantes datos del cliente. 3.1. Si se desea actualizar algún dato se lo modifica y luego se selecciona la opción Actualizar con lo cual los datos modificados quedan registrados. Termina U-C 4.1. Si no se desea registrar un nuevo cliente, se selecciona la opción Cancelar. Termina U-C 5.1. Si no se desea registrar un nuevo cliente, se selecciona la opción Cancelar. Termina U-C
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye: 6. Registrando reserva de asiento 14. Registrando encomienda		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

**Actualizacion de Datos de Cliente**

Datos Cliente

Tipo Documento  Numero

Apellido

Nombre

Direccion

Telefono

Nuevo Cliente Modificar Eliminar Aceptar Cancelar

**Cancelacion de Reserva**

Datos Viaje

Nro Reserva

Fecha

Origen

Destino

Lugar Ascenso

Asientos

	Numero	Precio

Total

Datos Cliente

Apellido

Nombre

Telefono

Aceptar Cancelar

Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Cancelando reserva</b>		Nro. de Orden: 9
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal	<b>Empleado de Agencia</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Registrar la cancelación de la reserva efectuada por el cliente</i>		
Precondiciones:.		
Post-Condiciones: <i>Se ha eliminado la reserva realizada por el cliente</i>		
Curso Normal	Alternativas	
<p>1. Comienza cuando el empleado de Agencia necesita registrar que se ha cancelado la reserva ingresando a la pantalla “Cancelación de reserva”.</p> <p>2. Se ingresa el número de reserva actualizándose los datos de la reserva: origen, destino, lugar de ascenso, número de asiento y precio como así también los datos del cliente que realizó la reserva: apellido, nombre y teléfono.</p> <p>3. Se selecciona la opción Cancelar exhibiéndose un mensaje “Desea eliminar la presente reserva?”</p> <p>4. Se selecciona la opción “Si” dándole de baja a la reserva y liberándose el o los asiento/s que poseía la reserva.</p> <p style="text-align: center;">Termina U-C</p>	<p>4.1. Si se selecciona la opción “No”, no se elimina la reserva. Termina U-C</p> <p>4.2. Si no se desea continuar con la cancelación de reserva, se selecciona la opción Cancelar. Termina U-C</p>	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del CU <b>Registrando venta de pasaje en agencia</b>	Nro. de Orden: 10	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal <b>Empleado de Agencia</b>	Actor Secundario:	
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Registrar la venta de pasajes efectuada en la agencia correspondiente</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha registrado la venta de pasajes en la agencia y se ha emitido el boleto</i>		
Curso Normal	Alternativas	
<p>1. Comienza cuando el empleado de Agencia necesita registrar la venta de una o más pasajes ingresando a la pantalla "Venta de pasaje en agencia".</p> <p>2. Si el cliente no tiene reserva se selecciona la opción Consultar asientos, se llama al caso de uso "Consultando asientos disponibles".</p> <p>3. Si hay asientos disponibles, se actualiza el origen, destino, fecha y hora de viaje.</p> <p>4. Se asigna el precio y el número de boleto, número de viaje y unidad (en una lista).</p> <p>5. Se muestra el importe total.</p> <p>6. Se selecciona la opción Aceptar.</p> <p>7. Se emite el pasaje correspondiente, registrándose el pago.</p> <p style="text-align: center;">Termina U-C</p>	<p>2.1. Si el cliente tiene reserva se ingresa el tipo y número de documento del cliente, fecha y hora del viaje.</p> <p>2.2. Se selecciona la opción Reserva, la cual actualiza los campos nombre y apellido del cliente, origen y destino del viaje, número(s) de asiento(s), número(s) de pasaje(s), precio del pasaje, número de viaje y unidad (El número de asiento y número de pasaje se muestra en una lista).</p> <p>2.3. Se actualiza el campo importe total</p> <p>2.4. Se selecciona la opción Aceptar.</p> <p>2.5. Se emite el pasaje correspondiente</p> <p style="text-align: center;">Termina U-C</p> <p>3.1. Si no hay asientos disponibles, el empleado puede efectuar alguna reserva. Se llama al caso de uso "Registrando reserva de asiento".</p> <p style="text-align: center;">Termina U-C</p> <p>6.1. Si no se desea concretar la venta, se selecciona la opción Cancelar</p> <p style="text-align: center;">Termina U-C</p>	
Asociaciones de Extensión:	6. Registrando reserva de asiento	
Asociaciones de Inclusión:	7. Consultando asientos disponibles	
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

**Venta de Pasaje en Agencia**

Datos Cliente

Apellido  Tipo Documento

Nombre  Numero

Datos Viaje

Fecha  Origen

Hora  Destino

Unidad

Datos Pasaje

	Nro Asiento	Nro Pasaje	Precio

Total

Reserva Consultar Asientos Imprimir Aceptar Cancelar

**Venta de Pasajes en Recorrido**

Datos Chofer

Apellido

Nombre

Datos Viaje

Fecha  Origen

Hora  Destino

Unidad

Datos Pasaje

	Nro Asiento	Precio

Total

Registrar Aceptar Cancelar



Nivel del Caso de uso		<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Registrando venta de pasaje en recorrido</b>		Nro. de Orden: 11	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal	<b>Empleado de Agencia</b>		Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto		<input type="checkbox"/> Abstracto
Objetivo: <i>Registrar venta de pasaje en recorrido</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha registrado la venta del pasaje en recorrido</i>			
Curso Normal		Alternativas	
1. Comienza cuando el empleado de Agencia necesita registrar la venta de un pasaje efectuado durante el recorrido por el chofer ingresando a la pantalla "Venta de pasajes en recorrido". 2. Se ingresa el origen y destino y la fecha. 3. Se actualizan los datos del chofer: nombre, monto y número de unidad. 4. Se ingresa el o los números de asientos asignados, la hora y el importe cobrado. 5. Se selecciona la opción Aceptar. Termina U-C		5.1. Si no se desea registrar la venta del pasaje se selecciona la opción Cancelar. Termina U-C	
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso	<b>Verificando venta del recorrido</b>	Nro. de Orden: 12
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media
	<input type="checkbox"/> Baja	
Actor Principal	<b>Empleado de Agencia</b>	Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Verificar el monto de las ventas realizadas durante el recorrido del colectivo</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha verificado el monto de las ventas de pasajes en el recorrido</i>		
Curso Normal	Alternativas	
1. Comienza cuando el empleado de Agencia verifica la rendición de cuenta del chofer con la rendición en el sistema, ingresando a la pantalla "Control Venta en Recorrido". 2. Se ingresa el origen, destino, fecha y hora. 3. Se actualizan los campos: chofer, número de unidad, número de viaje y una grilla con los pasajes vendidos, el importe correspondiente a cada pasaje y el importe total. 4. Se ingresa el importe real entregado por el chofer. 5. Se selecciona de una lista desplegable el nombre del empleado de agencia que realiza el control. 6. Se selecciona la opción Aceptar. Termina U-C	5.1. Si se desea realizar algún comentario u observación, se completa el campo "Observaciones". 6.1. Si no se desea cargar la verificación al sistema, se selecciona la opción Cancelar. Termina U-C	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

**Verificando Venta en Recorrido**

Datos Viaje

Fecha  Origen

Nro Unidad  Destino

Chofer

Pasaje

	Pasajes	Precio

Empleado

Total

Importe Real

Aceptar Cancelar

**Emision Planilla de Ventas Reservas**

Datos Viaje

Fecha  Origen

Hora  Destino

Datos Chofer

Nro Legajo  Nro Unidad

Apellido  Nro Viaje

Nombre

Ventas - Reservas

	Nro Asiento	Estado	Destino	Lugar Donde Sube	Precio	Cliente

Imprimir Cancelar

Nivel del Caso de uso		<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del CU <b>Emitiendo planilla de ventas y reservas</b>		Nro. de Orden: 13	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Actor Principal	<b>Empleado de Agencia</b>		Actor Secundario:
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto		<input type="checkbox"/> Abstracto
Objetivo: <i>Generar una planilla con los asientos ocupados por ventas y reservas, anexando un listado de precios para cada viaje</i>			
Precondiciones:			
Post-Condiciones: <i>Se ha emitido la planilla de ventas y reservas con los datos del viaje y un listado de precios</i>			
Curso Normal		Alternativas	
1. Comienza cuando el empleado de Agencia necesita emitir la planilla de ventas y reserva a cada chofer, ingresando a la pantalla “Emisión planilla de ventas y reservas”. 2. Se ingresa: origen y destino del viaje, hora y la fecha. 3. Se actualizan los campos: número de legajo del chofer, nombre y apellido, número de unidad y número de viaje. 4. Se actualizan los datos de los pasajeros: número de asiento, estado y destino. 5. Se selecciona la opción Imprimir.  6. Se anexa en un extremo de la hoja, el listado de precios de los pasajes. Termina U-C		4.1. Para los asientos reservados se muestra lugar donde sube, número(s) de asiento(s), estado, precio unitario y total, y los datos del cliente. 5.1. Si no se desea emitir esta planilla, se selecciona la opción Cancelar. Termina U-C	
Asociaciones de Extensión:			
Asociaciones de Inclusión:			
Caso de uso donde se incluye:			
Caso de uso donde se extiende:			
Caso de uso de Generalización:			

Nivel del Caso de uso	<input type="checkbox"/> Negocio	<input checked="" type="checkbox"/> Sistema de Información
Nombre del Caso de uso <b>Emitiendo informe de viajes y rendiciones</b>	Nro. de Orden: 24	
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media <input type="checkbox"/> Baja
Actor Principal <b>Encargado de Tesorería</b>	Actor Secundario:	
Tipo de Caso de uso:	<input checked="" type="checkbox"/> Concreto	<input type="checkbox"/> Abstracto
Objetivo: <i>Emitir informes estadísticos de viajes y rendiciones.</i>		
Precondiciones:		
Post-Condiciones: <i>Se ha emitido el informe estadístico de viajes y rendiciones.</i>		
Curso Normal	Alternativas	
<p>1. Comienza cuando el encargado de Tesorería necesita emitir un informe con los viajes y rendiciones realizadas, ingresando en la pantalla “Informes de viajes y rendiciones”.</p> <p>2. Se ingresan las fechas del periodo del cual se desea emitir el informe.</p> <p>3. Se abre una nueva ventana, la cual contiene los siguientes datos: n° de viaje, n° de unidad, nombre del chofer, origen, destino, horario, cantidad de pasajeros correspondientes a cada día del periodo establecido y el monto en concepto de pasajes y encomiendas.</p> <p>4. Si no se desea adjuntar este informe se selecciona la opción Informar. Se retorna a la ventana “Informes de Viajes y Rendiciones”.</p> <p>5. Se actualizan los campos en la ventana principal, fecha, N° de viaje, origen y destino, correspondientes a la mayor cantidad de pasajeros en un viaje.</p> <p>6. Con respecto a la mayor cantidad de encomiendas se actualizan los campos fecha, origen y destino.</p> <p>7. Se selecciona la opción Imprimir emitiéndose el informe con los datos que posee la ventana principal.</p> <p>8. Se selecciona la opción Aceptar. Termina U-C</p>	<p>4.1. Si se desea adjuntar el informe se selecciona la opción Imprimir.</p> <p>8.1. Si se desea cancelar la emisión del informe se selecciona la opción Cancelar. Termina U-C</p>	
Asociaciones de Extensión:		
Asociaciones de Inclusión:		
Caso de uso donde se incluye:		
Caso de uso donde se extiende:		
Caso de uso de Generalización:		

Informe Viajes - Rendiciones

Datos Chofer

Fecha Ingreso

Fecha Egreso

Mayor Cantidad de Pasaje

Fecha

Nro Viaje

Origen

Destino

Mayor Cantidad de Encomienda

Fecha

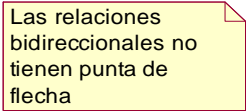
Origen

Destino

Imprimir Aceptar Cancelar

[illegible]

### 2.3. Modelo de Objetos del Dominio



<b>Persona</b> nombre apellido dirección teléfono	númeroDocumento tipoDocumento tipoCliente ↑Reserva ↑Encomienda	<b>Chofer</b> númeroDocumento tipoDocumento fechaNacimiento fechaIngreso fechaBaja motivoBaja ↑HorarioChofer ↑Viaje	<b>Taller</b> razónSocial númeroCUIT
<b>Unidad</b> númeroUnidad modelo patente marca cantidadAsientos fechaCompra Estado(reparación) ↑OcupaciónAsiento	<b>HorarioHabilitado</b> horarioPartida horarioLlegada origen destino díaSemana fechaDesde FechaHasta ↑Viaje	<b>Pasaje</b> númeroPasaje agencia ↑PrecioPasaje ↑OcupaciónAsiento ↑Viaje	<b>Viaje</b> númeroViaje fecha refuerzo (S/N) ↑PrecioPasaje ↑Reserva ↑Pasaje ↑Unidad ↑Horario-habilitado ↑RoturaCoche ↑Encomienda
<b>OcupaciónAsiento</b> númeroAsiento estadoAsiento ↑Viaje	<b>Reserva</b> númeroReserva fechaReserva lugarAscenso cantidadPersonas ↑Viaje ↑OcupaciónAsiento ↑Pasaje	<b>PrecioEncomienda</b> codPrecio kmDestino precioKilómetro precioKilogramo	<b>Rendición Diaria</b> fecha montoCaja montoEsperado concepto observaciones ↑Viaje ↑Depósito
<b>Depósito</b> ↑CuentaBanco nroBoletaDepósito fecha importeDepositado	<b>CuentaBanco</b> nombre númeroSucursal dirección teléfono númeroCuenta ↑EstadoCuenta	<b>Factura</b> númeroFactura descripción fecha monto estado ↑Pago ↑RoturaCoche	<b>Pago</b> montoTotal fechaPago númeroCheque fechaValidezCheque fechaCobroCheque nroReciboPago ↑CuentaBanco
<b>Encomienda</b> númeroEncomienda origen destino peso nombreDestinatario fechaEnvío monto estado fechaRecepción horaRecepción horaLlegada fechaLlegada fechaEntregaCliente ↑PrecioEncomienda ↑Viaje	<b>RoturaCoche</b> fechaDesperfecto horaDesperfecto lugarDesperfecto fechaReparación ↑Taller ↑Unidad	<b>Estado Cuenta</b> fecha saldoEsperado saldoActual observaciones	<b>HorarioChofer</b> díaSemana horaDesde horaHasta fechaDesde fechaHasta
		<b>PrecioPasaje</b> localidadOrigen localidadDestino precioPasaje porcentajeDescuento	



### 3. Flujo de trabajo del Análisis

- 3.1. Diagrama de Colaboración.
- 3.2. Subsistemas.

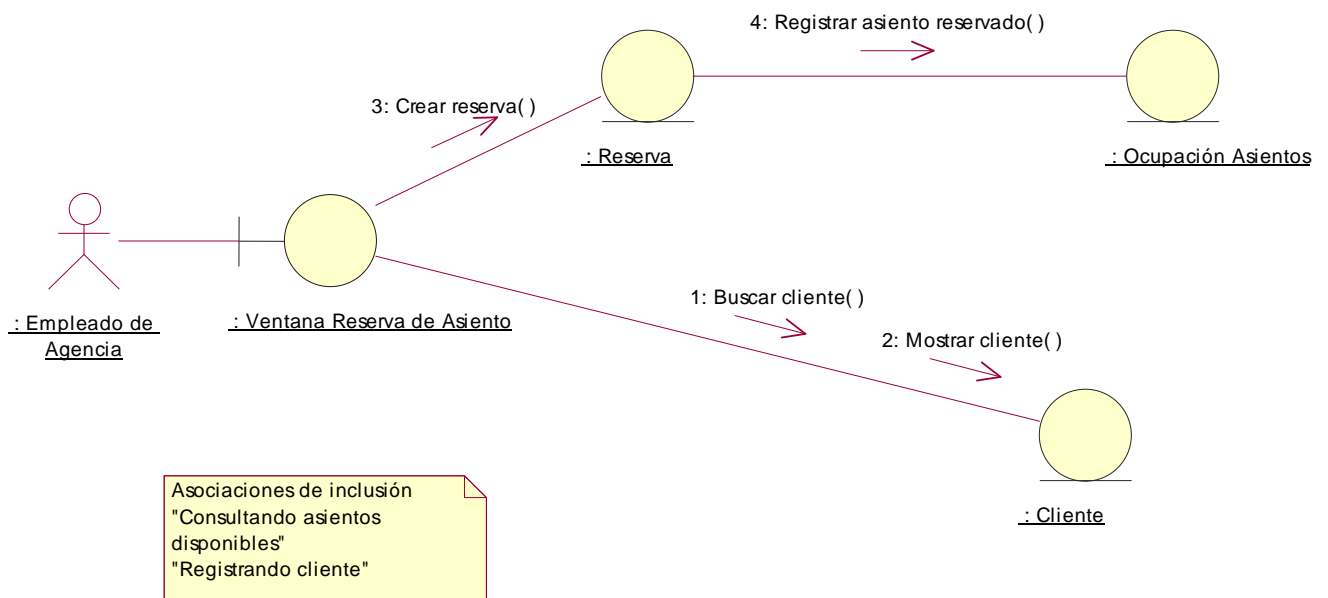
A partir del modelo de análisis se han considerado los siguientes casos de uso para desarrollar los diagramas de colaboración y las responsabilidades y colaboraciones de los objetos que aparecen en dichos diagramas

Los Casos de Uso considerados son los siguientes:

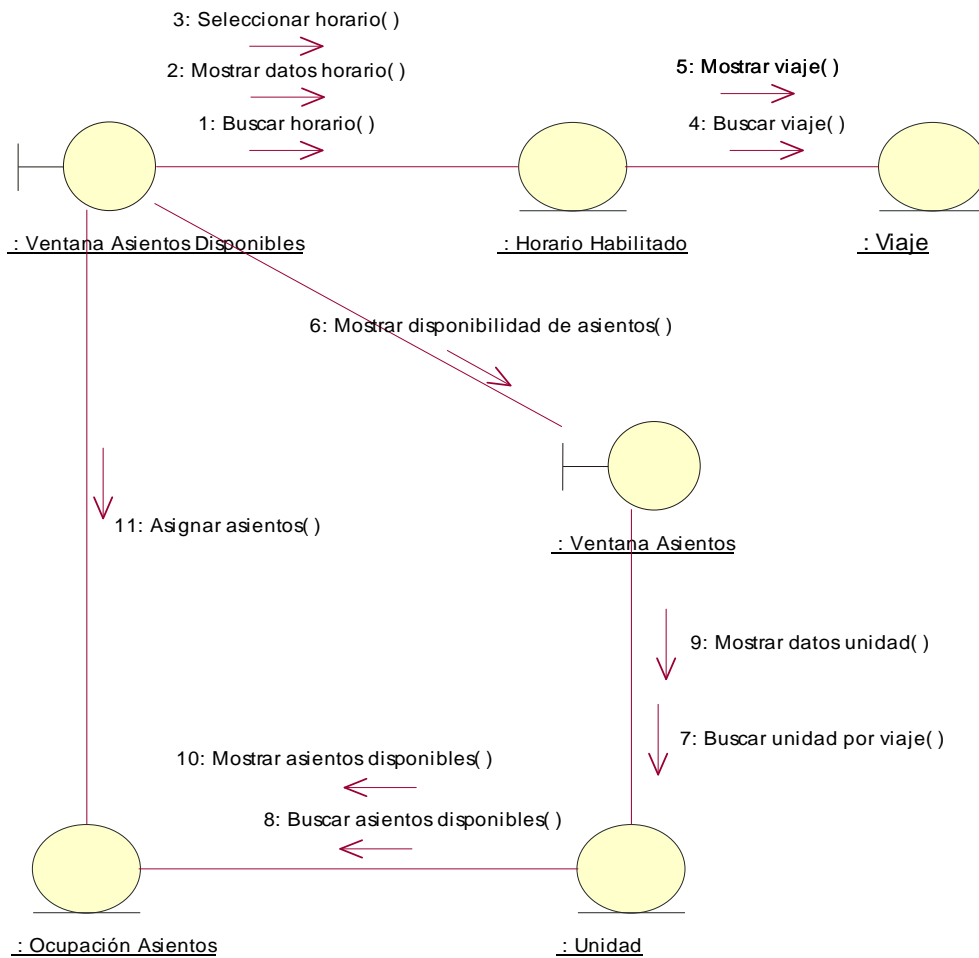
- U-C 6: Registrando reserva de asiento
- U-C 7: Consultando asientos disponibles
- U-C 8: Actualizando datos de cliente
- U-C 10: Registrando venta de pasaje en agencia
- U-C 13: Emitiendo planilla de ventas y reservas

#### 3.1 Diagramas de Colaboración

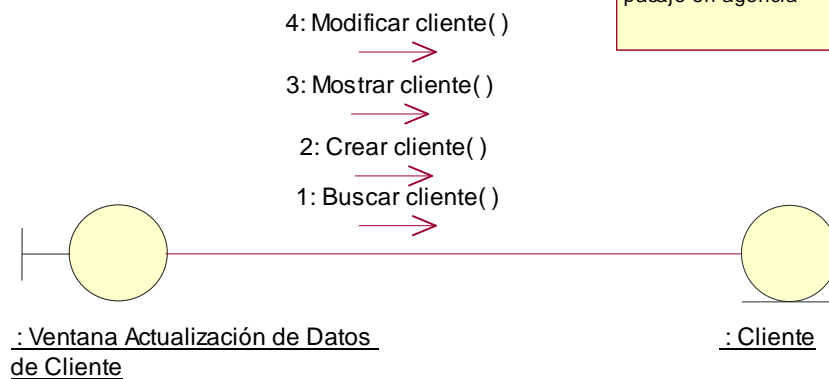
##### U-C 6 “Registrando reserva de asiento”



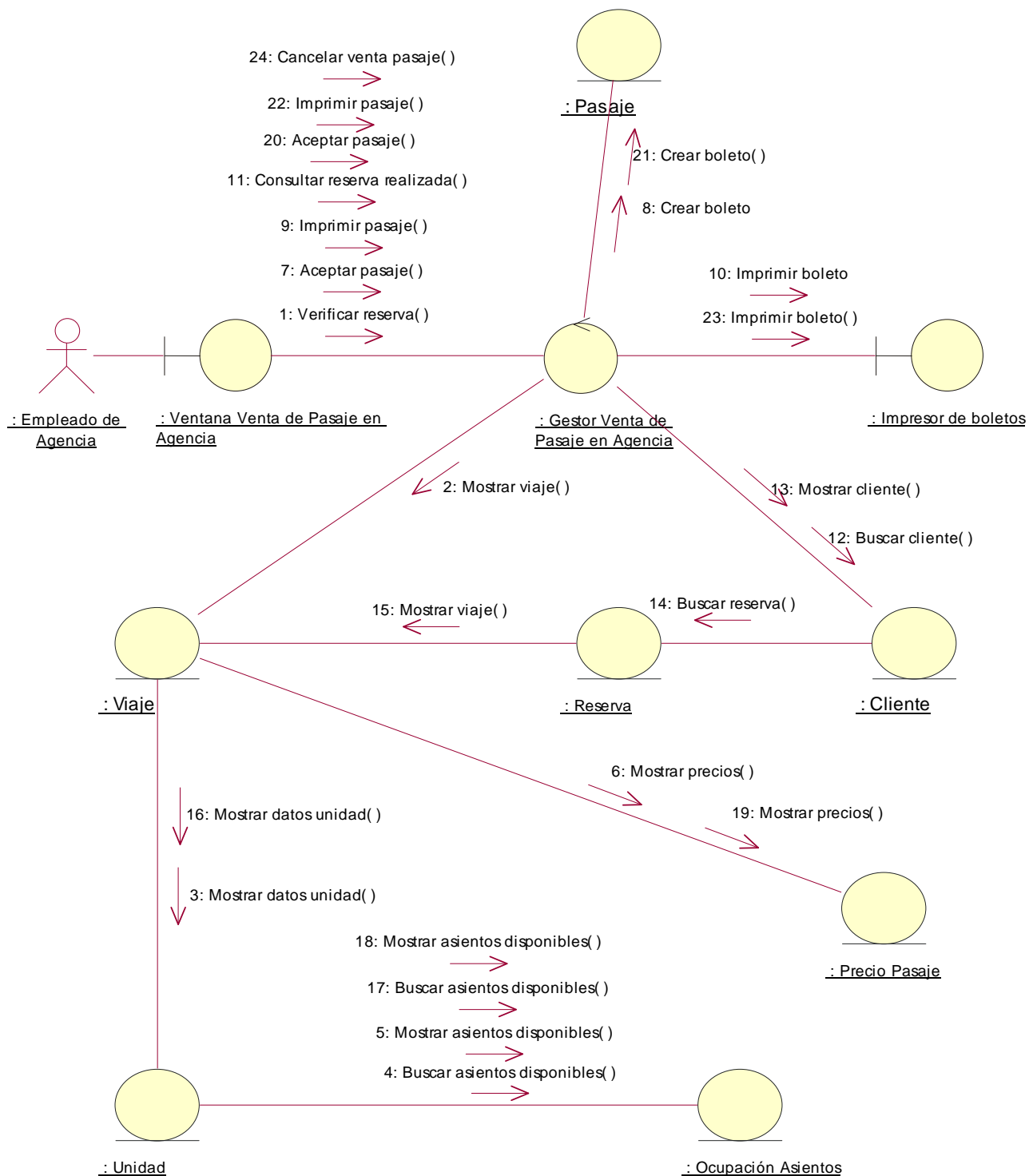
### U-C 7 “Consultando asientos disponibles”



### U-C 8 “Actualizando datos cliente”



U-C donde se incluye  
"Registrando reserva  
de asiento"  
"Registrando venta de  
pasaje en agencia"

**U-C 10 “Registrando venta de pasaje en agencia”**

Asociaciones de inclusión  
 " Consultando asientos  
 disponibles"  
 "Actualizando datos cliente"

## **3.2. Subsistemas**

---

### **▶ I. Subsistema Archivos**

---

- ▶ U-C.1. Actualizando datos de chofer
- ▶ U-C.2. Actualizando datos de unidad
- ▶ U-C.8. Actualizando datos de cliente

### **▶ 2. Subsistema Horarios de Viaje**

---

- » U-C.3. Registrando horarios del próximo trimestre
- » U-C.4. Registrando asignación de chofer y unidad

### **▶ 3. Subsistema Reserva y Ventas de pasaje**

---

- ▶ U-C.6. Registrando reserva de asiento
- ▶ U-C.7. Consultando asientos disponibles
- ▶ U-C.9. Cancelando reserva
- ▶ U-C.10. Registrando venta de pasaje en agencia
- ▶ U-C.11. Registrando venta de pasaje en recorrido
- ▶ U-C.12. Verificando venta en recorrido
- ▶ U-C.13. Emitiendo planilla de venta y reserva

### **▶ 4. Subsistema Encomiendas**

---

- » U-C.14. Registrando encomienda
- » U-C.15. Emitiendo listado de encomienda
- » U-C.5. Registrando precio de encomienda
- » U-C.16. Registrando entrega de encomienda en agencia
- » U-C.17. Registrando entrega de paquete al cliente destinatario

### **▶ 5. Subsistema Reparación de Unidades**

---

- ▶ U-C.20. Registrando unidad reparada
- ▶ U-C.18. Registrando desperfecto de unidad

## ► 6. Subsistema Control de Ingresos y Egresos

- » U-C.21. Registrando pago de unidad reparada
- » U-C.25. Registrando factura pendiente de pago
- » U-C.19. Registrando depósito de ingreso en el día
- » U-C.22. Verificando estado de cuenta del banco
- » U-C.23. Emitiendo informe de ingresos y egresos semanales
- » U-C.24. Emitiendo informe de viajes y rendiciones

## 4. Flujo de trabajo de Diseño

- 4.1 Ambiente de implementación:
- 4.2 Diagrama de Despliegue
- 4.3 Diagrama de Clases de Diseño.
- 4.4 Diagrama de Estados.

### 4.1. Ambiente de implementación

#### ► Lenguaje de Programación

El sistema de información de la empresa Todo Viaje se implementará utilizando un lenguaje de programación visual: Visual Basic 6.0.

#### ► DBMS

El sistema de administración de bases de datos (DBMS) es el conjunto de programas que manejan todo acceso a la base de datos. En este caso se utilizará SQL Server 2000

#### ► Aspectos de Seguridad

Se establecerán niveles de usuarios, administrando los permisos y accesos a las distintas opciones del sistema.

#### ► Definición del Hardware

Se debe determinar la configuración del equipamiento que soportará el sistema de información.

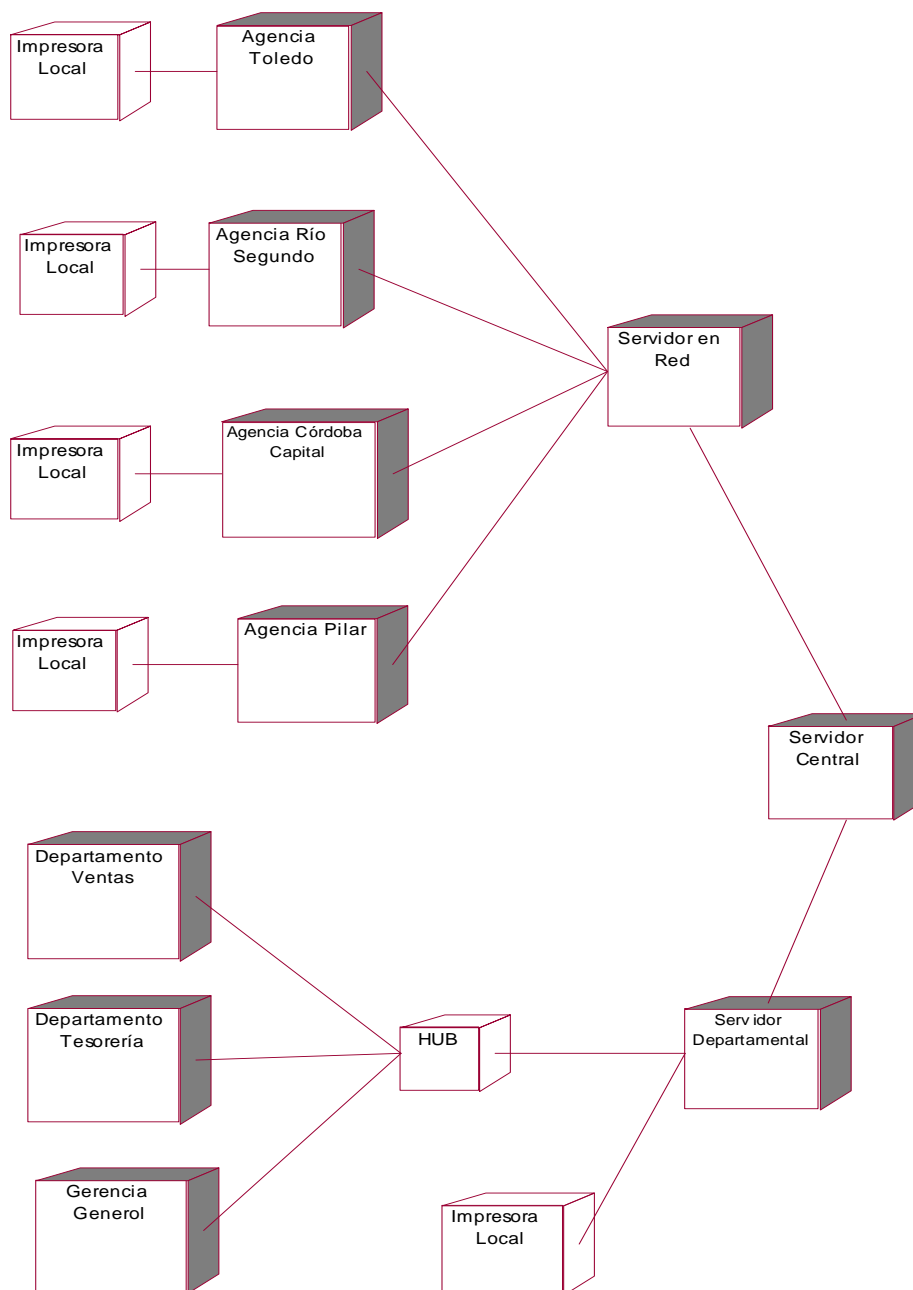


#### Nota

En cuanto a este punto, ambiente de implementación, aquí se desarrolla una descripción del software y hardware relacionado con el sistema de información a desarrollar, efectuando una justificación del porque de dicha elección en cada aspecto en particular.

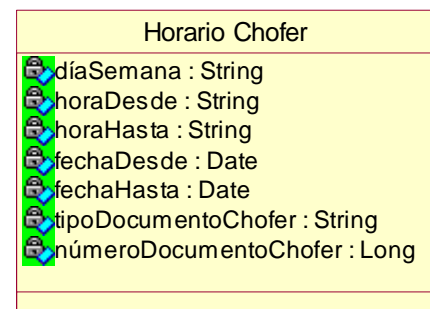
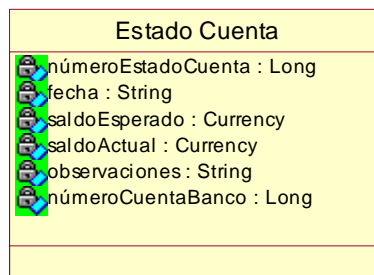
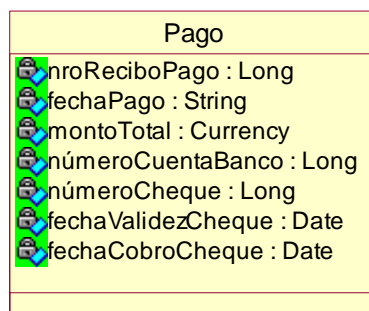
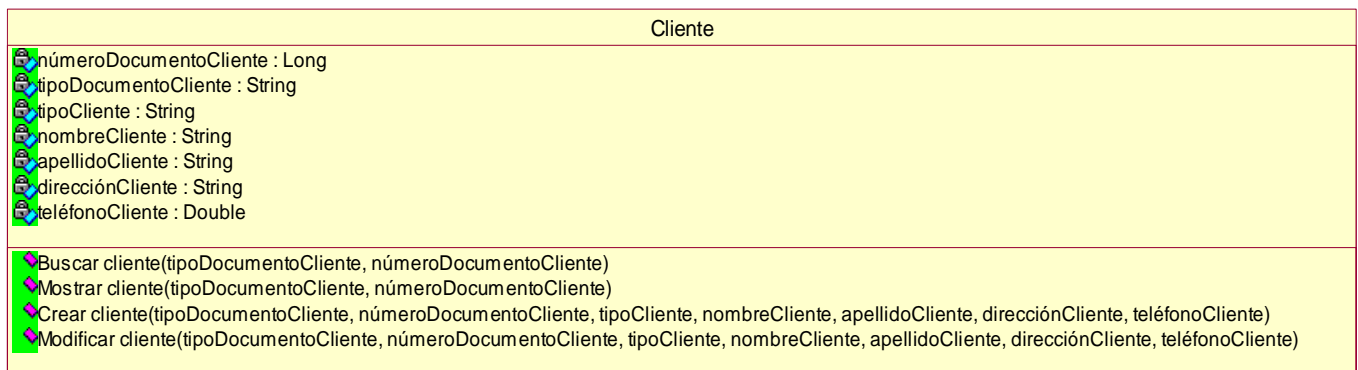
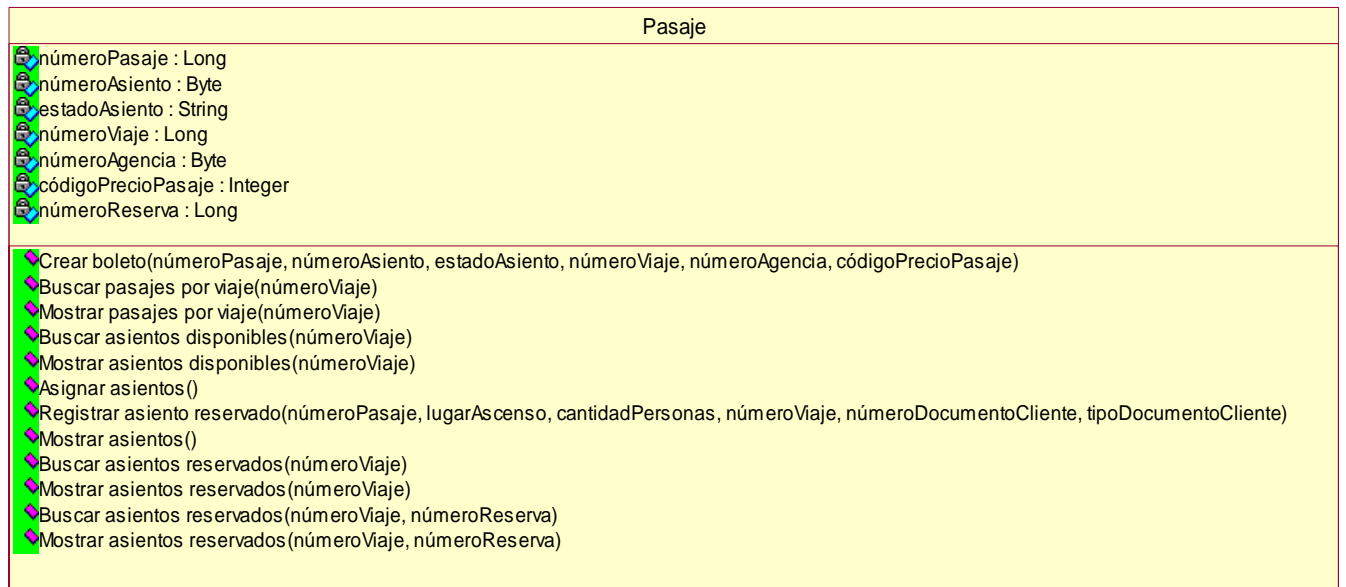
Aquí solo se ha hecho una referencia muy general al respecto.

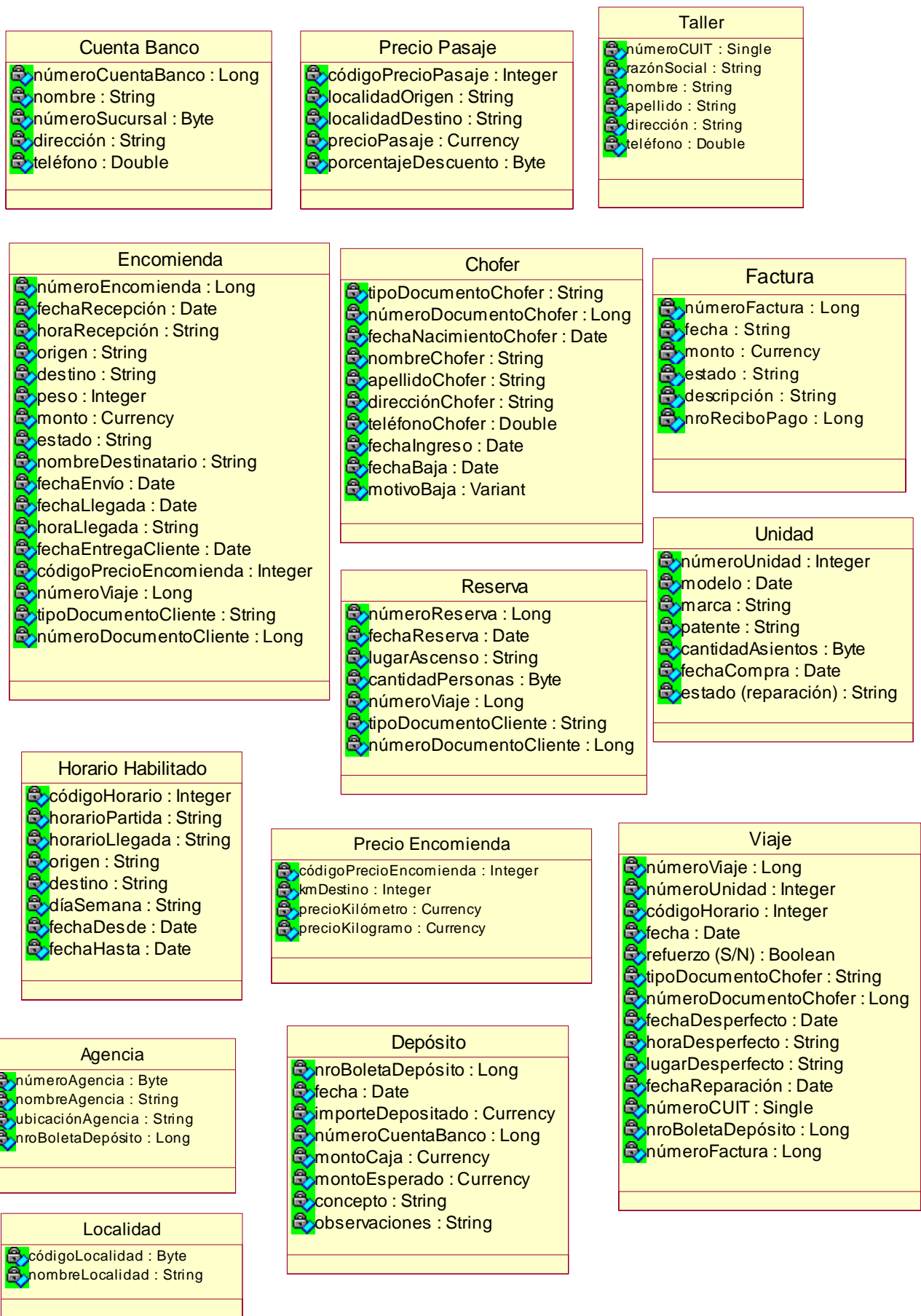
## 4.2 Diagrama de Despliegue



## 4.2 Diagrama de Clases

Las clases que se muestran arriba son una referencia de cómo completar las clases de diseño: atributos con tipos y operaciones con parámetros. Esto se realiza para todas las clases del diagrama de clases de diseño

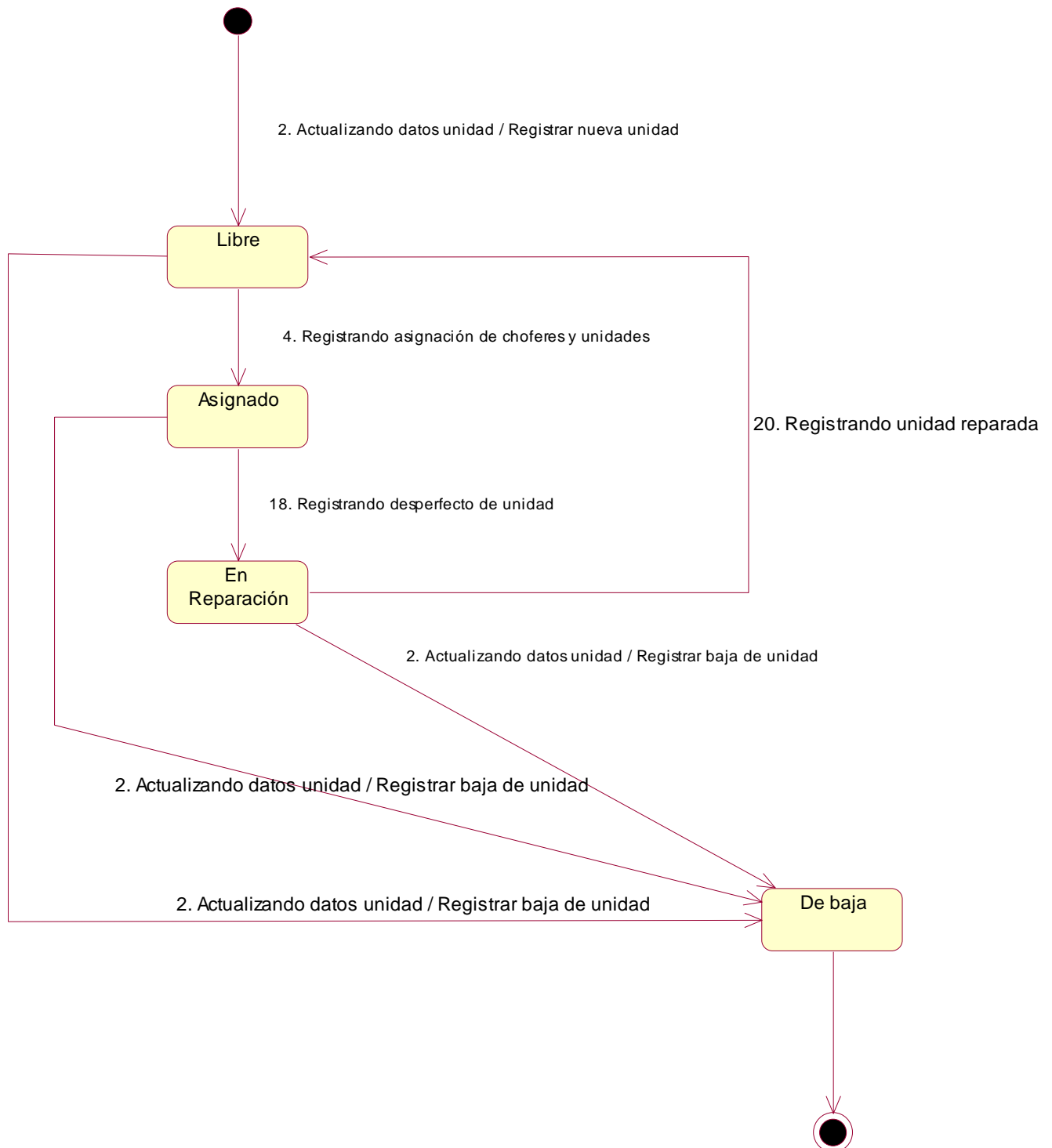




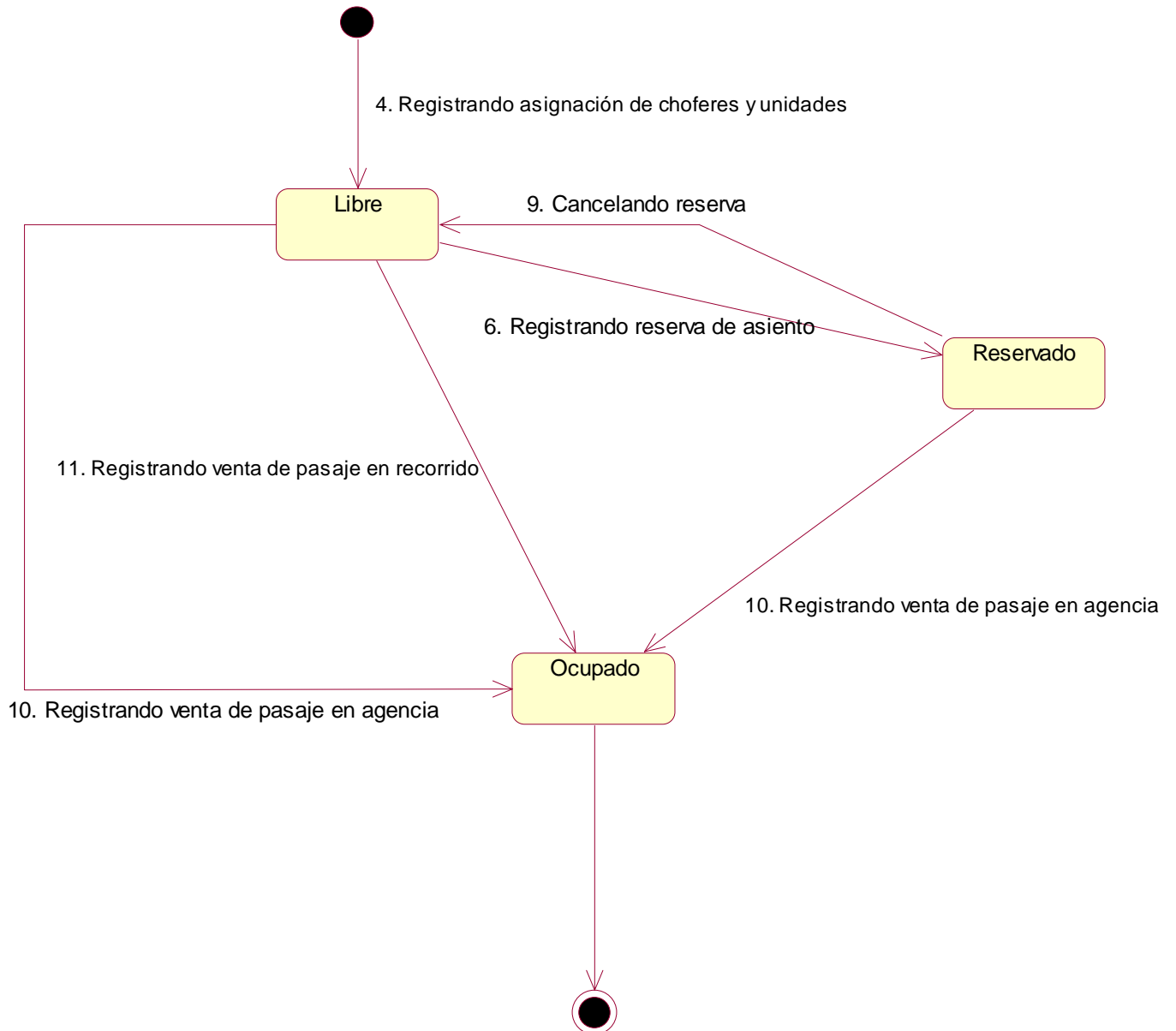


## 4.4. Diagramas de Estado

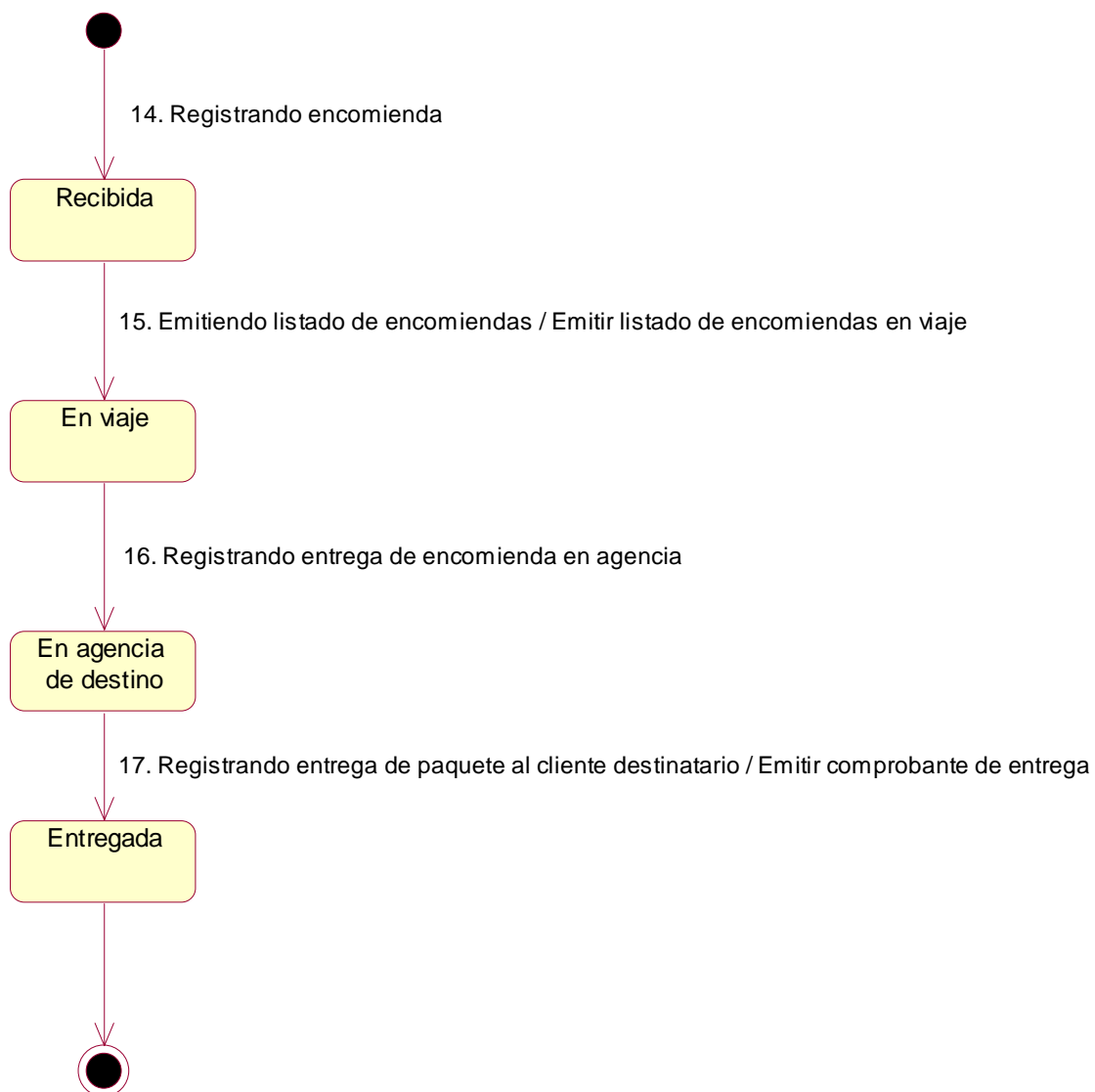
### Objeto Unidad



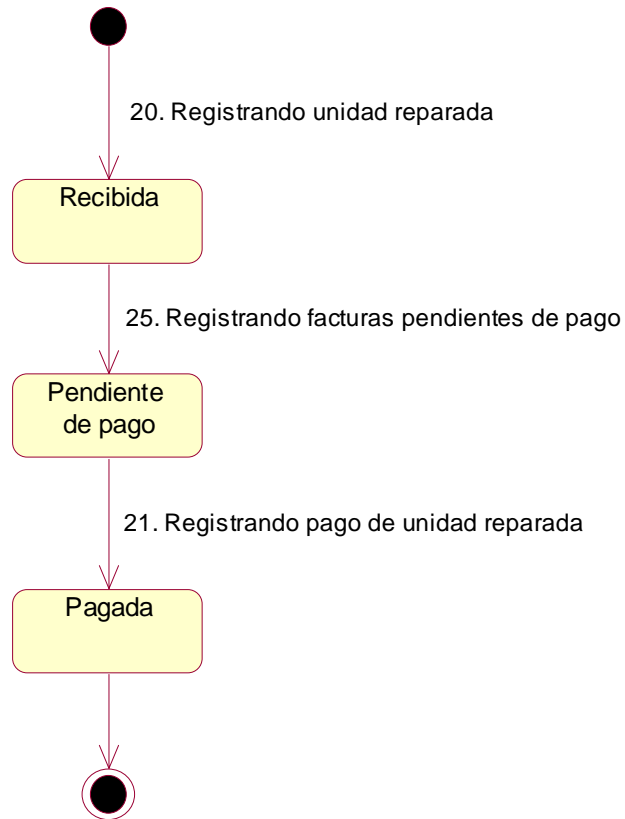
## Objeto Pasaje



## Objeto Encomienda



## Objeto Factura



Recuerda que la cátedra te ofrece un nuevo práctico, completamente desarrollado para que consultes las herramientas aplicadas. También se pueden consultar en la Web, los trabajos finales.

# Documentación para desarrollo o construcción

## Desarrollo

Desarrollar y documentar cada uno de los subsistemas que forman parte del Sistema Integrado, y que deberán ser construidos por los desarrolladores.

## Capacitación de Programadores

La finalidad es capacitarlos para que puedan desarrollar los programas especificados y definidos con la nueva configuración.

Esta capacitación deberá proporcionar conocimientos sobre:

1. Descripción, posibilidades y limitaciones del nuevo equipo
2. Descripción del software de base, utilitarios y lenguaje de programación.
3. Organización y Diseño de archivos
4. Transmisión de datos
5. Ejercicios prácticos
6. Manuales

## Especificación de las necesidades de Formularios

La finalidad es que la imprenta reciba las instrucciones para la elaboración de los formularios previstos en el diseño de salidas, en caso de ser formularios continuos preimpresos o especiales y en el diseño de entradas.

El Equipo de Proyecto debe supervisar toda la gestión de:

- » Elaboración de formularios
- » Corrección de los proyectos
- » Aprobación final
- » Recepción
- » Distribución por áreas

Las especificaciones para cada formulario contendrán:

- ▶ Papel: espesor, peso, color, tamaño
- ▶ Tinta: color
- ▶ Separación entre líneas y columnas
- ▶ Número de formulario
- ▶ Número de serie (si fuesen numerados)
- ▶ Plazo de entrega
- ▶ Cantidades requeridas
- ▶ Instrucciones para el llenado, que se detallará en los manuales del sistema

## **Desarrollo de los Programas**

---

En este paso se deberá desarrollar los programas del sistema. Se tendrán en cuenta las especificaciones de las necesidades de programación, y además que los programadores pueden ser internos de la empresa o externos.

Abarca las tareas de:

### **1. Diagramación**

Incluye la preparación del Diagrama de cada programa. Es la solución gráfica para el problema planteado en el programa.

### **2. Codificación**

Implica la transformación de las operaciones incluidas en los diagramas en instrucciones del lenguaje de programación seleccionado.

### **3. Compilación**

Una vez compilado el programa fuente se transforma en programa objeto. (Instrucciones de máquina)

Se debe imprimir cada uno de los programas fuente en salidas de impresora denominada “Listado Fuente”, de ser así requerido.

## Diseño de la migración y carga inicial de datos

Esta actividad sólo se llevará a cabo cuando sea necesaria una carga inicial de información o una migración de datos de otros sistemas, cuyo alcance y estrategia a seguir se habrá establecido previamente.

Para ello, se toma como referencia el plan de migración y carga inicial de datos, que recoge las estructuras físicas de datos del sistema o sistemas origen implicadas en la conversión, la prioridad en las cargas y secuencia a seguir, las necesidades previas de depuración de la información, así como los requisitos necesarios para garantizar la correcta implementación de los procedimientos de migración sin comprometer el funcionamiento de los sistemas actuales.

A partir de dicho plan, y de acuerdo a la estructura física de los datos del nuevo sistema obtenida en la actividad Diseño Físico de Datos y a las características de la arquitectura y del entorno tecnológico propuesto en la actividad Definición de la Arquitectura del Sistema, se procede a definir y diseñar en detalle los procedimientos y procesos necesarios para realizar la migración.

Se completa el plan de pruebas específico establecido en el plan de migración y carga inicial, detallando las pruebas a realizar, los criterios de aceptación o rechazo de la prueba y los responsables en la organización, realización y evaluación de los resultados de las mismas.

Asimismo, se determinan las necesidades adicionales de infraestructura tanto para la implementación de los procesos como para la realización de las pruebas.

Como resultado de esta actividad se actualiza el plan de migración y carga inicial de datos con la información siguiente:

- » Especificación del entorno de migración
- » Definición de procedimientos de migración
- » Diseño detallado de módulos
- » Especificación técnica de las pruebas
- » Planificación de la migración y carga inicial

Es importante considerar que una carga inicial de información no tiene el mismo alcance y complejidad que una migración de datos, de modo que las tareas de esta actividad se llevarán a cabo en mayor o menor medida en función de las características de los datos a cargar.

## Especificación del Entorno de Migración

---

El objetivo de esta tarea es especificar el entorno tecnológico propio de los procesos de migración y carga inicial, adecuando al entorno tecnológico del sistema las necesidades y requisitos reflejados en el plan de migración y carga inicial de datos. En la descripción del entorno tecnológico se tendrán en cuenta, asimismo, las herramientas o utilidades software específicas de estos procesos.

Se realiza una estimación de capacidades (*capacity planning*) para este entorno, que permita evaluar las necesidades de infraestructura, principalmente relacionadas con el espacio de almacenamiento y las comunicaciones.

## Diseño de Procedimientos de Migración y Carga Inicial

---

El objetivo de esta tarea es la definición de los procedimientos necesarios para llevar a cabo la migración y carga inicial de datos del sistema. Como punto de partida, junto con los requisitos y especificaciones de migración y carga inicial se tendrá en cuenta el modelo físico de datos optimizado y su localización en los nodos, así como la definición del entorno tecnológico del sistema de información.

Los procedimientos asociados a la migración y carga inicial de datos son, principalmente los relacionados con la preparación, la realización y la posterior verificación del proceso. Entre ellos se encuentran los siguientes:

### **Procedimientos de seguridad, relativos a**

- ▶ Control de acceso a la información
- ▶ Copias de seguridad de los procesos
- ▶ Recuperación de la información
- ▶ Tratamiento de las posibles contingencias durante la conversión

### **Procedimientos de carga de datos, relativos a:**

- ▶ Depuraciones previas de información
- ▶ Procesos de validación
- ▶ Procesos de importación
- ▶ Procesos de carga y prioridades
- ▶ Procedimientos de verificación de los procesos y comprobación de la integridad de la información resultante al finalizar la conversión, conforme a la estructura física de los datos destino



## Diseño Detallado de Componentes de Migración y Carga Inicial

El objetivo de esta tarea es el diseño detallado, en sucesivos niveles de detalle, de los módulos de migración y carga inicial, indicando la jerarquía y orden de ejecución.

El diseño de los módulos necesarios para la migración y carga inicial, no es conceptualmente distinto del diseño de cualquier otro módulo del sistema de información, por lo que se recomienda utilizar pautas similares. Se tendrá en cuenta el modelo físico de datos del sistema de información, así como las estructuras de datos del sistema o sistemas origen recogidas en el plan de migración y carga inicial de datos.

Finalmente, se complementa el plan de migración y carga inicial con la definición de los distintos tipos de prueba a realizar.

### Productos

- ▶ De entrada
  - ▶▶ Plan de migración y carga inicial de datos, detectado en el Plan del Sistema y/o relevamiento detallado
  - ▶▶ Diseño de la arquitectura del sistema
  - ▶▶ Entorno tecnológico del sistema
  - ▶▶ Modelo físico de datos optimizado
  - ▶▶ Esquemas físicos de datos
  - ▶▶ Asignación de esquemas físicos de datos a nodos
- ▶ De Salida
  - ▶▶ Plan de migración y carga inicial de datos
  - ▶▶ Especificación del entorno de migración
  - ▶▶ Definición de procedimientos de migración
  - ▶▶ Diseño detallado de módulos
  - ▶▶ Especificación técnica de las pruebas

### Participantes

- ▶ Diseñadores de Arquitectura
- ▶ Equipo de Soporte Técnico
- ▶ Diseñadores de Componentes

## Autoevaluación

---

Teniendo como referencia el caso, completamente desarrollado de la unidad II “El complejo Cineclub”, analice las siguientes situaciones y desarrolle cada uno de los modelos solicitados.

### **Caso práctico N°1**

*El hotel Resplandor, se encuentra ubicado en la ciudad de Santa Fe. El mismo aloja pasajeros de excursiones como clientes particulares que se hospedan por algunos días. Cada vez que un pasajero llega al hotel este debe registrarse en la recepción, en donde completa una ficha de ingreso al hotel. Esa ficha contiene los datos de los pasajeros (nombre, apellido, estado civil, documento, domicilio, etc.). Una vez que se ha completado el trámite el conserje verifica las habitaciones que tiene disponibles y le asigna una, entregándole la llave de la misma.*

*Diariamente el pasajero solicita alguno de los servicios que se brindan en el hotel (lavandería, cafetería, etc.), para eso, este debe dirigirse al conserje, el cual registrará la solicitud del servicio que desea, y se lo asignará a la habitación correspondiente, emitiéndole al pasajero un recibo como constancia del mismo. Al finalizar el día, se ordenan y registran en una ficha realizada por habitación todos los servicios brindados junto con el correspondiente comprobante y se lo emite a contaduría.*

*Cuando el pasajero cancela su estadía, se emite un listado y la factura correspondiente, detallando los importes del alojamiento como así también el de los servicios prestados. Mensualmente se realiza un listado que se emite a Gerencia informando de la cantidad de habitaciones ocupadas, los importes cobrados y las anomalías surgidas en ese período.*

Defina en el Modelo de Diseño

El diagrama de clases

Diagrama de transición de estados

Defina el modelo de despliegue

Ambiente implementación

Lay out de la organización

Diagrama de despliegue

### **Caso práctico N°2**

*En la Clínica XXX, los pacientes son atendidos de acuerdo al orden de llegada. Cuando el paciente ingresa, se le toman los datos con el objeto de buscar su historia clínica, en la cual el médico anotará: diagnóstico, terapia sugerida y análisis clínicos. En la clínica no se reciben mutuales, y el precio de cada consulta varía según la especialidad.*

*Cuando se cobra se emite un recibo para el paciente, y se registra el importe en un informe de ingreso a caja, que diariamente es enviado a Contaduría. Además del recibo, se le entrega al paciente una solicitud de consulta, que este le entregará al médico al ser atendido.*

*Todos los meses deben enviarse a Personal un reporte que contenga la cantidad de consultas atendidas para cada médico, para que le realicen la respectiva liquidación de honorarios.*

1- Según su criterio, seleccione dos artefactos necesarios y suficientes (aprendidas hasta el momento), para diseñar en forma exitosa los requerimientos esenciales que se plantean en este caso práctico.

**Nota**

Estos mismos casos fueron trabajados parcialmente en Sistemas III (los modelos y diagramas correspondientes a la captura de Requisitos y el Análisis), y ahora la idea es, completarlos con los nuevos diagramas aprendidos (pertenecientes al Diseño), y obtener así, dos casos MÁS, resueltos completamente, para que les sirvan de referencia, al momento de emprender el desarrollo de un sistema de Información.

Estos casos serán desarrollados en grupo o individualmente, corroborando las respuestas, con el docente en los encuentros presenciales.



# ► Unidad IV

## Implementación

### Objetivos

- Conocer de qué se tratan las pautas para una correcta implementación
- Distinguir los métodos de conversión: global, paralelo, modular, gradual
- Conocer qué es la migración y carga inicial de datos
- Diferenciar y relacionar evaluación de cumplimiento y calidad

### Contenidos de la Unidad

#### 1- Introducción a la implementación

Definición y Establecimiento del Plan de implementación

a- Definición del Plan de Implantación

b- Especificación del Equipo de Implantación

Formación o capacitación para la implementación

Objetivos de la Capacitación o Adiestramiento

Consideraciones al proceso de capacitación

Esquema de Capacitación

Formación para la implementación

Preparación de la formación del equipo de implementación

Preparación de la formación de usuarios finales

Incorporación del Sistema a entorno de operación

Preparación de la instalación:

Realización de la Instalación

Carga de datos al entorno de operación

Pruebas de implementación del sistema

Pruebas de aceptación del sistema

Evaluación y ajuste

Preparación del Mantenimiento del Sistema

#### 2- Presentación y aprobación del sistema

Informe Final

Paso a entorno de producción

#### 3-Elaboración de manuales de usuario

Manuales de Sistemas:

#### 4-Evaluación de la implementación

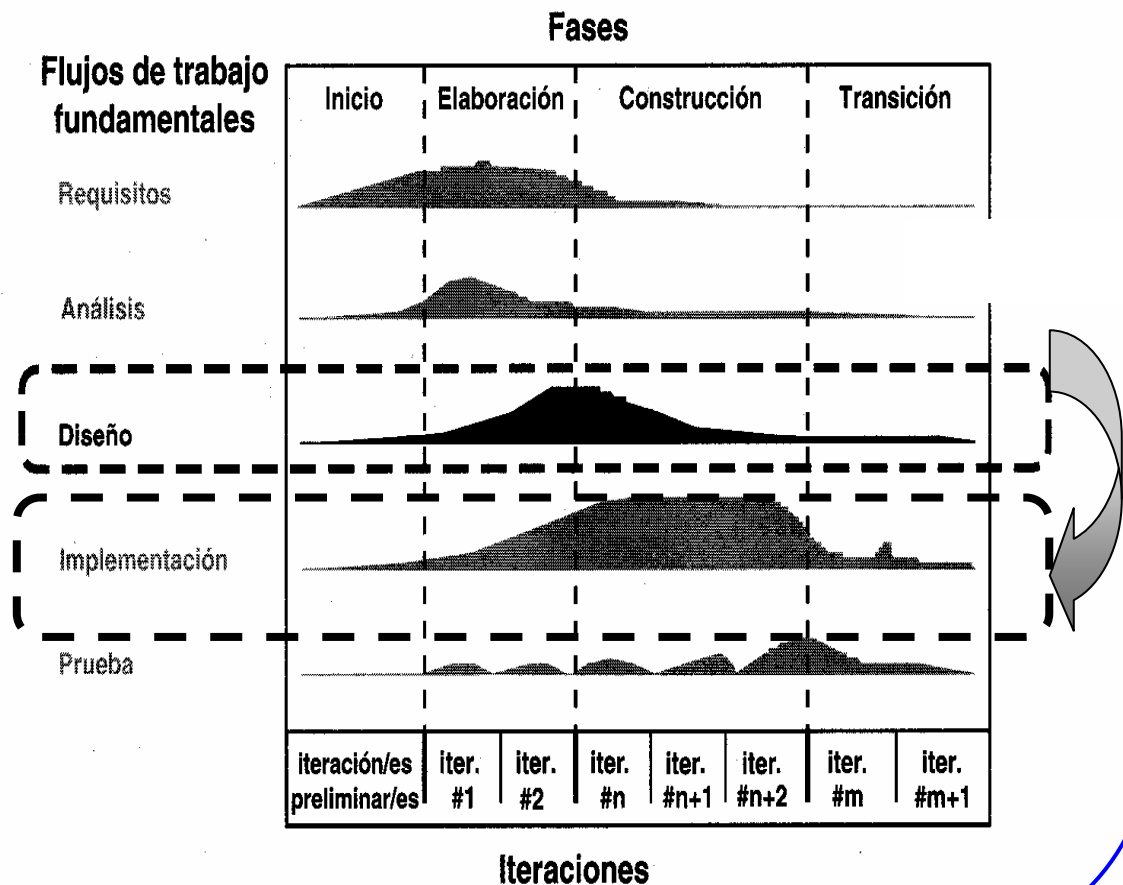
Mantenimiento

Autoevaluación

## Mapa conceptual

# Proceso Unificado de desarrollo de Software

## Ciclo de Vida



## Introducción a la Implementación

La implementación tiene como objetivo principal la entrega y aceptación del sistema en su totalidad y la realización de todas las actividades necesarias para el paso a producción del mismo.

Es decir, hace entrega al usuario para que comience su operación, pone en marcha el sistema desarrollado y lo someter a un proceso de evaluación.

En resumen, la implementación debe considerar los siguientes puntos:

- » En primer lugar, se revisa la estrategia de implantación que ya se determinó en el proceso Estudio de Factibilidad del Sistema, en el armado del Plan. Se estudia su alcance y en función de sus características, se define un plan de implantación y se especifica el equipo que lo va a llevar a cabo. Conviene señalar la participación del usuario de operación en las pruebas de implantación, del usuario final en las pruebas de aceptación y del responsable de mantenimiento.
- » Las actividades previas al inicio de la producción incluyen la preparación de la infraestructura necesaria para configurar el entorno, la instalación de los componentes, la activación de los procedimientos manuales y automáticos asociados y, cuando proceda, la migración o carga inicial de datos. Para ello se tomarán como punto de partida los productos software probados, obtenidos en el proceso Construcción del Sistema de Información, con su documentación asociada.
- » Se realizan las pruebas de implantación y de aceptación del sistema en su globalidad que responden a los siguientes propósitos:
  - ▶ Las pruebas de implantación cubren un rango muy amplio, que va desde la comprobación de cualquier detalle de diseño interno hasta aspectos tales como las comunicaciones.
  - ▶ Se debe comprobar que el sistema puede gestionar los volúmenes de información requeridos, se ajusta a los tiempos de respuesta deseados y que los procedimientos de respaldo, seguridad e interfaces con otros sistemas funcionan correctamente.
  - ▶ Se debe comprobar el comportamiento del sistema bajo las condiciones más extremas. Las pruebas de aceptación se realizan por y para los usuarios y tienen como objetivo validar formalmente que el sistema se ajusta a sus necesidades.
  - ▶ Asimismo, se llevarán a cabo las tareas necesarias para la preparación del mantenimiento siempre y cuando se haya decidido que los sistemas de información implicados en la implantación van a ser objeto de mantenimiento. En cualquier caso, es necesario que la persona que vaya a asumir el mantenimiento conozca el sistema, antes de su incorporación al entorno de producción.
- » Además se determinan los servicios que requiere el sistema que se va a implantar, especificando los niveles de servicio y el acuerdo que se adquiere una vez que se inicie la producción. Se distinguen los servicios de gestión de operaciones (servicios por lotes, seguridad, comunicaciones, etc.) y los servicios al cliente (servicio de

atención a usuario, mantenimiento, etc.) que se deberán negociar en cuanto a recursos, horarios, coste, etc. Se fija el nivel con el que se prestará el servicio como indicador de la calidad del mismo.

En síntesis



Conviene señalar que la implantación puede ser un proceso iterativo que se realiza de acuerdo al plan que se establezca para el comienzo de la producción del sistema en su entorno de operación.

Para establecer este plan se tendrá en cuenta:

- ▶ El cumplimiento de los requisitos de implantación definidos en el Plan
- ▶ Establecimiento de Requisitos y especificados en el Análisis
- ▶ Establecimiento de Requisitos de Implantación, en el Diseño
- ▶ La estrategia de transición del sistema antiguo al nuevo
- ▶ Finalmente se realizan las acciones que sean necesarias para el inicio de la producción

## Definición y Establecimiento del Plan de implementación

---

En esta actividad se revisa

- a) La **estrategia de implantación** para el sistema a implantar establecida inicialmente en el proceso Estudio de Factibilidad del Sistema. Se identifican los distintos sistemas de información que forman parte del sistema objeto de la implantación y se analizan, para cada uno de ellos, las posibles dependencias con otros proyectos que puedan condicionar el plan de implantación.  
Se estudia el alcance de la implantación y sus condicionantes, se decide si se puede llevar a cabo la implantación del sistema estableciendo, en su caso, la estrategia de forma definitiva que se concretará en el plan de implantación.
- b) Se constituye el **equipo de implantación** determinando los recursos humanos necesarios para la propia instalación del sistema, para las pruebas de implantación y aceptación, y para la preparación del mantenimiento. Se identifican para cada uno de ellos, sus perfiles y niveles de responsabilidad.



### ***(a) Definición del Plan de Implantación***

La estrategia de implantación del sistema se habrá determinado, como se especifica más adelante, en función de la envergadura del sistema, es decir, el número de sistemas de información implicados en la implantación y la cobertura geográfica, cuyo alcance dependerá de las características y complejidad de los sistemas de información que conforman el sistema objeto de la implantación.

Se revisan los requisitos de implantación (instalación, infraestructura, formación) establecidos en la tarea Ambiente Implementación y los procedimientos implicados en la implantación establecidos para cada uno de los sistemas de información en la tarea Especificación de Requisitos de Operación y Seguridad, con el fin de asegurar su adecuación a la estrategia global de implantación.

Una vez analizada la información anterior se define un plan de implantación que permita dimensionar adecuadamente el esfuerzo y los recursos necesarios para llevar a cabo con éxito la implantación.

### ***Plan, contiene...***

1. Formación necesaria para la implementación tanto a usuarios finales como al grupo que se encarga de realizar pruebas de implementación y aceptación del sistema
2. Preparación de recursos necesarios para incorporación a entorno de operación
3. Instalación de los componentes manuales y automáticos asociados a cada módulo
4. Ejecución de los procesos de carga inicial
5. Realización de las pruebas de implementación y de aceptación
6. Formalización del plan de mantenimiento

### ***Operación del nuevo sistema***



El objetivo de este paso es empezar a operar el nuevo sistema con transacciones reales, a fin de efectuar los ajustes que sean necesarios antes de dar por entregado el sistema al usuario

Los operadores deberían haber recibido y comprobado el equipo, además de estar capacitados.

Las áreas usuarias deben analizar las salidas del nuevo sistema y formular todas aquellas observaciones que pudieran surgir.

El traspaso o conversión del sistema actual, usualmente llamado estrategia de conversión, al nuevo sistema puede realizarse según alguno de estos métodos:

---

### ► **Proceso encadenado o gradual**

---

- » En este caso, el sistema actual sigue funcionando y produciendo la información establecida, el nuevo sistema toma las transacciones ya procesadas por el sistema actual y produce sus propios resultados en un período de tiempo más tarde.
- » La información producida por el sistema actual se la distribuye al usuario y la producida por el nuevo sistema queda en poder del Equipo de Proyecto para su evaluación.
- » Este método se usa cuando el nuevo sistema es muy complejo o vulnerable y se le exige un alto grado de confiabilidad.
- » Permite la comparación con los resultados del sistema actual y facilita el análisis de los resultados del nuevo sistema fuera de línea.

---

### ► **Proceso directo o global**

---

- En este caso se inactiva al sistema actual y las transacciones se procesan en el nuevo sistema.
- Este método se usa cuando el sistema de información no es complejo y si las dificultades iniciales esperadas se estiman inferiores a las consecuencias de no comenzar en tiempo el nuevo sistema.
- Además, cuando resulte difícil la simultaneidad de las tareas de los sistemas vigente y nuevo.

---

### ► **Proceso en paralelo**

---

- » Las transacciones se procesan en el sistema actual y en el nuevo sistema simultáneamente.
- » Por razones de costo, se puede inactivar el sistema actual tan pronto como los resultados del nuevo sistema de información sean confiables.
- » Este método se usa cuando la Gerencia (Máximo Nivel) exige una alta confiabilidad del nuevo sistema o cuando los sistemas son muy diferentes (No habrá duplicaciones en procesos y salidas).
- » Tiene la ventaja de minimizar el riesgo de comenzar con un sistema defectuoso.

---

### ► **Proceso por subsistemas o modular**

---

- El sistema actual continúa procesando una parte de las transacciones y el nuevo sistema el resto.
- La secuencia en la que los subsistemas comenzarán su operación debe ser cuidadosamente planificada.
- Reduce costos y riesgos.

#### **Productos**

- De entrada
  - » Descripción de la solución (Estrategia de implantación)
  - » Diseño de la arquitectura del sistema

- Entorno tecnológico del sistema
- Procedimientos de operación y administración del sistema
- Procedimientos de seguridad y control de acceso
- Plan de migración y carga inicial de datos
- Plan de pruebas
- Catálogo de requisitos (de implantación)
- De salida
  - Plan de implantación

### **Técnicas**

- Sesiones de trabajo

### **Participantes**

- Comité de Dirección
- Jefe de Proyecto
- Responsable de Implantación
- Responsable del Grupo de Usuarios



Después de haber hecho una lectura comprensiva del tema: “Plan de Implementación”, te invitamos a que participes del foro que el docente propone, para que elabores tus propias apreciaciones, experiencias y las compartas con el grupo de trabajo Autlico.

## ***b) Especificación del equipo de Implantación***

Se constituye el equipo de trabajo necesario para llevar a cabo la implantación y aceptación del sistema, según el plan de implantación establecido en la tarea anterior.

Para ello, se identifican en función del nivel de esfuerzo requerido, los distintos participantes implicados en la implantación del sistema (usuarios, equipo técnico y responsable de mantenimiento), determinando previamente sus perfiles, responsabilidades, nivel de implicación y fechas previstas de participación a lo largo de toda la implantación.

### **Productos**

- De entrada
  - Recursos humanos y técnicos disponibles (externo)
  - Catálogo de usuarios, obtenido del relevamiento
  - Plan de implantación
- De salida
  - Equipo de Implantación

**Participantes**

- ▶ Comité de Dirección
- ▶ Jefe de Proyecto
- ▶ Responsable de Implantación
- ▶ Responsable del Grupo de Usuarios

---

**Formación o capacitación para la implementación**

---



En esta actividad se prepara y se imparte la formación al equipo que participará en la implantación y aceptación del sistema y se realiza el seguimiento de la formación de usuarios finales

---

De esta forma se asegura que la implantación se llevará a cabo correctamente.

- ▶ Se determina la formación que va a ser necesaria para **el equipo de implantación**, en función de los distintos perfiles y niveles de responsabilidad identificados en la actividad anterior. Para ello, se establece un plan de formación que incluye los esquemas de formación correspondientes, los recursos humanos y de infraestructura requeridos para llevarlo a cabo, así como una planificación que quedará reflejada en el plan de formación.
- ▶ La formación para que **los usuarios finales** sean capaces de utilizar el sistema de forma satisfactoria. En esta actividad, se analizan los esquemas de formación definidos según los diferentes perfiles y se elabora un plan de formación que esté alineado **con** el plan de implantación.

---

**Objetivos de la Capacitación o Adiestramiento**

---



Es lograr que los usuarios tengan el dominio necesario de las cosas básicas acerca de las maquinarias y procesos que se emplean para su operación de manera eficiente y segura

---

---

**Consideraciones al proceso de capacitación**

---

Se refiere a la formación necesaria que debe ser provista, tanto al equipo de implementación como los usuarios finales. Es importante considerar quienes serán los docentes, según se trate de entrenar sobre el sistema o sobre conceptos básicos de informática.

Es enseñar a los usuarios que se relacionan u operan en un proceso de implantación, tanto a los que tengan un uso primario como un uso secundario del sistema.

La responsabilidad de esta capacitación de los Usuarios primarios y secundarios es del Analista, desde el personal de captura de datos hasta aquellos que toman las decisiones sin usar una Computadora. No se debe incluir a personas de diferentes niveles de habilidad e intereses de trabajo; debido a que si en una Empresa existen trabajadores inexpertos no se pueden incluir en la misma sección de los expertos ya que ambos grupos quedaran perdidos.

Aun y cuando la empresa puede contratar los Servicios de Instructores externos, el analista es la persona que puede ofrecer la mejor capacitación debido a que conoce el personal y al sistema mejor que cualquier otro. A la falta o imposibilidad del analista la organización puede contratar otros servicios de capacitación como son:

- » Vendedores: Son aquellos que proporcionan capacitación gratuita fuera de la Empresa de uno o dos días.
- » Instructor pagado externamente: Son aquellos que pueden enseñar todo acerca de las computadoras pero para algunos usuarios esta no es una capacitación necesaria.
- » Instructores internos: Están familiarizados con el personal y pueden adecuar los materiales a sus necesidades, pero le faltaría experiencia en Sistemas de Información que es realmente la necesidad del usuario.
- » Otros usuarios del sistema: Es en el caso de formar grupos de usuarios entrenados, entonces, primero se capacita a un grupo, y luego este colabora en la capacitación del grupo siguiente y así sucesivamente.

Esquema de Capacitación

Posibles instructores	A quién adiestrar	
	Usuarios primarios	Usuarios secundarios
Vendedores		X
Analistas de sistemas	X	X
Instructores externos contratados		X
Instructores internos	X	
Otros usuarios del sistema	X	

<b>Preparación de la formación del equipo de implementación:</b>	<b>Preparación de la formación de usuarios finales</b>
1. Definir formación necesaria para el equipo responsable de la implementación, estableciendo el esquema de formación para cada tipo de perfil dentro del equipo y la duración estimada y los objetivos a lograr, por cada grupo de perfiles de usuarios considerado.	1. Plan de formación a usuarios finales, con esquema de formación, materiales y planificación a seguir. 2. Nómina de usuarios a capacitar. Considerando las distintas tipificaciones de usuarios, desde gerenciales hasta operativos. Inclusive si poseen conocimientos básicos para la

<b>Preparación de la formación del equipo de implementación:</b>	<b>Preparación de la formación de usuarios finales</b>
<ol style="list-style-type: none"> <li>2. Asegurar recursos humanos, técnicos y materiales necesarios para realizar la formación</li> <li>3. Instalación donde se llevará a cabo la capacitación. Externo a la empresa, o dentro de la misma, en un lugar destinado a tal fin.</li> <li>4. Horarios en que se desarrollará el plan de capacitación en los distintos niveles, para no interferir en las tareas habituales, pero considerando el Plan del Proyecto de Implementación.</li> <li>5. Convocar a las personas que deban asistir a los cursos de formación, con antelación.</li> </ol>	utilización óptima del sistema.

### Esquema

Objetivos del adiestramiento	dependen de los requerimientos de las tareas del usuario
Métodos de adiestramiento	dependen de las tareas del usuario, personalidad, formación y experiencia; utiliza la combinación de conferencias, demostraciones, prácticas y estudio
Sitios de adiestramiento	dependen del objetivo del adiestramiento, costos, disponibilidad, sitios disponibles del vendedor con equipos operables; instalaciones internas, instalaciones rentadas
Materiales de adiestramiento	dependen de las necesidades del usuario; manuales de operación; casos prototipos del equipo y de la salida

## **Incorporación del Sistema a entorno de operación**

En esta actividad se realizan todas las tareas necesarias para la incorporación del sistema al entorno de operación en el que se van a llevar a cabo las pruebas de implantación y aceptación del sistema.

Mientras que las pruebas unitarias, de integración y del sistema se pueden ejecutar en un entorno distinto de aquél en el que finalmente se implantará, las pruebas de implantación y aceptación del sistema deben ejecutarse en el entorno real de operación.

El propósito es comprobar, que el sistema, satisface todos los requisitos funcionales y no funcionales especificados por el usuario en las mismas condiciones que cuando se inicie la producción.

Por tanto, como paso previo a la realización de dichas pruebas y de acuerdo al plan de implantación establecido, se verifica que están disponibles todos los recursos necesarios para que se pueda realizar, adecuadamente, la instalación de todos los componentes que integran el sistema, así como la creación y puesta a punto de las bases de datos en el entorno de operación. Asimismo, se establecen los procedimientos de explotación y uso de las bases de datos de acuerdo a la normativa existente en dicho entorno.

### ***Preparación de la instalación***

- ▶ Verificar que estén disponibles los recursos necesarios para introducir el sistema a entorno de operación.
- ▶ Verificar procedimientos de seguridad y control de acceso, operación y administración del sistema.
- ▶ Verificar si se necesita de migración de datos y las características del entorno.
- ▶ Instalar el software propiamente dicho.
- ▶ Verificar que el team de testing tenga la formación necesaria para llevar a cabo las pruebas.

### ***Realización de la Instalación***

Teniendo en cuenta los estándares y normativas por los que se rige la organización en los entornos de operación y de acuerdo al plan de implantación establecido, se realiza:

- » La instalación de todos los componentes del nuevo sistema de acuerdo a su ubicación física, establecida en el proceso Diseño del Sistema de Información, incluidos los procedimientos manuales y automáticos.
- » Se prepara el entorno de datos identificando los sistemas de información que forman parte del sistema objeto de la implantación y, para cada uno de ellos:
  - ▶ Se crean las bases de datos a partir del esquema físico elaborado en el proceso de construcción

- ▶ Se establecen los procedimientos de explotación y uso de las bases de datos, es decir, la normativa necesaria para la utilización de las bases de datos, actualización, consulta, etc.
  - ▶ Se revisan los procedimientos necesarios para realizar las copias de seguridad de los datos y de restauración de las copias indicando su frecuencia, así como los procedimientos de consolidación y sincronización de la información, éstos últimos cuando proceda
  - ▶ Se preparan las autorizaciones de acceso a los datos para los distintos perfiles de usuarios.
- » Una vez comprobada la correcta instalación del nuevo sistema, se activan los procedimientos de operación, de administración del sistema, de seguridad y de control de acceso, que incluirán el arranque y cierre del sistema según la frecuencia establecida, planificación de trabajos, recuperación y reanudación de trabajos, autorizaciones de acceso al sistema según los distintos perfiles de usuario, etc.
- » Además, si es necesaria una migración de datos se activarán también los procedimientos asociados.

### ***Carga de datos al entorno de operación***

Teniendo en cuenta que los sistemas de información que forman parte del sistema a implantar pueden mejorar, ampliar o sustituir a otros ya existentes en la organización, será necesaria una carga inicial y/o una migración de datos, cuyo alcance dependerá de las características y cobertura de cada sistema de información implicado.

Por tanto, la necesidad de una migración de datos puede venir determinada desde el proceso Estudio de Factibilidad del Sistema, evaluando las opciones del enfoque de desarrollo e instalación más apropiados para llevarlo a cabo.

## **Pruebas de implementación del sistema**

Se realizan las pruebas de implantación del sistema con el fin de comprobar el funcionamiento correcto del mismo en el entorno de operación y permitir al usuario, desde el punto de vista de operación, que determine la aceptación del sistema una vez instalado en su entorno real, según el cumplimiento de los requisitos no funcionales especificados.

Para ello:

1. Comprobar disponibilidad de recursos humanos y técnicos necesarios para realizar las pruebas.
2. Revisar verificaciones establecidas en el plan de pruebas.
3. Considerar condiciones límites previstas para las pruebas.
4. Comunicar el plan de pruebas al equipo responsable de llevarlas a cabo.
5. Referidas a recuperación, seguridad, rendimiento, comunicaciones... netamente técnicas.
6. Comparar los resultados obtenidos contra los esperados.



7. Identificar el origen de cada problema para remitirla a quien corresponda, determinar la envergadura de las modificaciones y qué acciones deben llevarse a cabo para resolverlo de forma satisfactoria.
8. Indicar si el plan de pruebas debe volver a realizarse total o parcialmente, y si será necesario contemplar nuevos casos de prueba no considerados anteriormente.
9. Registrar el resultado de las pruebas, indicando el Rechazo o Aceptación en el entorno instalado.

## Pruebas de aceptación del sistema



Las pruebas de aceptación se llevan a cabo con el fin de validar que el sistema cumple los requisitos básicos de funcionamiento esperado y permitir al usuario que determine la aceptación del sistema

Por este motivo, estas pruebas son realizadas por el usuario final del sistema y es durante este periodo de tiempo, cuando debe plantear todas las deficiencias o errores que encuentre antes de dar por aprobado el sistema definitivamente.

El responsable del grupo de usuarios revisa los criterios de aceptación que se especificaron previamente en el plan de pruebas del sistema y dirige las pruebas de aceptación final que llevan a cabo los usuarios a los que corresponda. A su vez, éstos últimos deben elaborar un informe que el responsable del grupo de usuarios analiza y evalúa con el fin de determinar la aceptación o rechazo del sistema.

Para ello:

1. Analizar criterios de aceptación establecidos por el usuario y revisar casos de prueba previstos.
2. Comunicar el Plan de pruebas de aceptación a los usuarios implicados.
3. Realizar las pruebas de aceptación final propiamente dichas, para asegurar que todos los componentes responden a los criterios de aceptación especificados.
4. Confeccionar un Informe en el que consten las desviaciones de los requisitos establecidos y los problemas que quedan aún sin resolver.
5. Comparar los resultados obtenidos con los esperados.
6. Identificar el origen de cada problema para remitirlo a quien corresponda y determinar acciones o medidas correctivas a llevar a cabo para resolverlo.
7. Indicar si hay pruebas que deben volver a realizarse o contemplar casos diferentes.
8. Documentar el resultado global de las pruebas de aceptación efectuados.

**Nota:** Ver Anexo '**Plan de Aceptación del Producto**'

## Evaluación y ajuste

---



El objetivo de este paso es evaluar los resultados de la operación del nuevo sistema, antes de la entrega final. Se analizarán los errores detectados en la puesta en marcha

---

Dichos errores pueden ser de Diseño y/o Desarrollo, como así también, salidas previstas que a pesar de no tener errores, no satisfacen a los usuarios.



### Importante

1. Los errores o insatisfacciones deben ser corregidos sin excepción.
  2. Si el error/los errores es/son crítico/s convendrá suspender la operación del nuevo sistema, reactivar el anterior y reajustar el nuevo según corresponda.
  3. Si el error/los errores no es/son crítico/s conviene seguir la operación del nuevo sistema y solucionar el problema en el momento que sea más conveniente.
- 

## Preparación del Mantenimiento del Sistema

---

El objetivo de esta actividad es que el equipo que va a asumir el mantenimiento el sistema esté familiarizado con él antes de que el sistema pase a producción. Para conseguir este objetivo se ha considerado al responsable de mantenimiento como parte integrante del equipo de implantación, y por lo tanto se habrá tenido en cuenta su perfil al elaborar el esquema de formación correspondiente.

Una vez que el responsable de mantenimiento ha recibido la formación necesaria y adquirido una visión global del sistema que se va a implantar, se le entregan los productos que serán objeto del mantenimiento. De esta manera obtiene de una forma gradual un conocimiento más profundo del funcionamiento y facilidades que incorpora el sistema, permitiéndole acometer los cambios solicitados por los usuarios con mayor facilidad y eficiencia y reduciendo, en consecuencia, el esfuerzo invertido en el mantenimiento.

Es importante resaltar que la existencia de una configuración del software aunque no garantiza un mantenimiento libre de problemas, sí permite reducir el esfuerzo requerido y mejora la calidad general del software a mantener. Una pobre configuración del software puede tener un impacto negativo sobre la capacidad de mantener fácilmente el software.

## Presentación y aprobación del sistema

Una vez que se han llevado a cabo las pruebas de implantación y de aceptación y se ha fijado el acuerdo de nivel de servicio, la Dirección debe formalizar la aprobación del sistema. Para esto, se lleva a cabo una presentación general del sistema y se espera la confirmación de su aprobación.

### Informe Final

El objetivo de este paso es informar a la Gerencia (Máximo Nivel) sobre el estado del nuevo sistema y su condición de entrega final.



Es un informe por escrito en el que se incluye lo actuado en esta etapa, las observaciones de los usuarios, errores, correcciones y aprobación. Se debe exponer a la Gerencia en presencia de los responsables de las áreas involucradas

En esta tarea se recopilan los productos del sistema de información y se presentan al Comité de Dirección para su aprobación.

#### Productos

- ▶ De entrada
  - ▶▶ Resultado de las pruebas unitarias
  - ▶▶ Evaluación del resultado de las pruebas de integración
  - ▶▶ Evaluación del resultado de las pruebas del sistema
  - ▶▶ Producto software
  - ▶▶ Especificación de la formación a usuarios finales
  - ▶▶ Código fuente de los componentes de migración y carga inicial de datos
  - ▶▶ Procedimientos de migración y carga inicial de datos
  - ▶▶ Evaluación del resultado de las pruebas de migración y carga inicial de datos
- ▶ De salida
  - ▶▶ Aprobación del sistema de información.

#### Técnicas

- ▶ Presentación

#### Participantes

- ▶ Comité de Dirección
- ▶ Jefe de Proyecto

## **Modelo de Informe Final**

**a) Introducción**

**b) Puesta en Marcha**

Se hace referencia a la operación del nuevo sistema, los errores detectados y las soluciones implementadas.

**c) Carpeta del Sistema**

Debe contener:

- ▶ Plan e informe del Reconocimiento
- ▶ Plan e informe del Relevamiento
- ▶ Estudio de Factibilidad
  - ▶▶ Técnica
  - ▶▶ Operativa
  - ▶▶ Económico/Financiera
- ▶ Informe sobre la situación del Diseño
- ▶ Informe sobre la situación del Desarrollo
- ▶ Documentación de:
  - ▶▶ Sistema
  - ▶▶ Subsistemas
  - ▶▶ Operación del Sistema

**d) Consideraciones finales**

- ▶ Conclusiones
- ▶ Cronograma de tareas
- ▶ Proposiciones para la entrega del nuevo sistema a los usuarios
- ▶ Decisiones necesarias

## **Aprobación**

Se requiere aprobación del Informe Final y de lo actuado para seguir adelante con el Mantenimiento o finalizar el Ciclo del Sistema.

## **Operación**

El objetivo de este paso es que el usuario se haga cargo de la operación del nuevo sistema.

Para ello debe tener toda la Documentación del Sistema de Información.

## **Entrega**

La entrega será tan prolongada y rigurosa como haya sido la prueba del sistema nuevo, la evaluación, el ajuste necesario y la participación del usuario.

El Equipo de Proyecto debe asegurarse que:

- ▶ El nuevo sistema cumple con los resultados esperados.
- ▶ Las Áreas tienen toda la Documentación a usar.
- ▶ Se ha concluido con la Capacitación.

## Recepción

Si el Proyecto del nuevo sistema fue encarado por un Equipo de Proyecto ajeno a la organización, el usuario receptor es todo el organismo, incluyendo el Área de Sistemas.

Si el Proyecto fue hecho por el Área de Sistemas, el usuario receptor es el personal de las Áreas involucradas.

## Paso a entorno de producción

Tarea netamente técnica, e implica la actualización física real del sistema de información existente por uno nuevo o modificado. Esta actividad tiene como objetivo establecer el punto de inicio en que el sistema pasa a producción, se traspasa la responsabilidad al equipo de mantenimiento y se empiezan a dar los servicios establecidos en el acuerdo de nivel de servicio, una vez que el Comité de Dirección ha aprobado el sistema.

Para ello, es necesario que después de haber realizado las pruebas de implantación y de aceptación del sistema, se disponga del entorno de producción perfectamente instalado en cuanto a hardware y software de base, componentes del nuevo sistema y procedimientos manuales y automáticos.

En función del entorno en el que se hayan llevado a cabo las pruebas de implantación y aceptación del sistema, se deberán instalar los componentes del sistema total o parcialmente y también se tendrá en cuenta la necesidad de migrar todos los datos o una parte de ellos.

Una vez que el sistema ya está en producción, se le notifica al responsable de mantenimiento, al responsable de operación y la Dirección.

## Documentación entregable

- » Guía de Instalación (Del Sistema o Subsistemas).
- » Debe estar el detalle de los pasos a seguir para instalar el nuevo sistema. Nivel Técnico.
- » Especificaciones de la Aplicación.
- » Detalle de cada una de las Aplicaciones (Subsistemas o Módulos). Nivel Técnico.
- » Notas de Liberación.
- » Detalle de errores detectados con posterioridad a la finalización de la Guía de Instalación (Es un complemento de la misma, tiene un semejante a la Fe de Erratas de un Libro), sus correcciones y métodos de implementación. Nivel Técnico.
- » Manual de Procedimientos de Usuario.

- » Detalle de procedimientos. Nivel Usuario operador.
- » Manual de Administración del Sistema.
- » Detalle completo de la administración del Sistema. Nivel Usuario supervisor.

## Elaboración de Manuales de Usuario

---

La familia de las normas ISO 9000 incluye requisitos para los sistemas de calidad que se puedan utilizar para lograr la interpretación común, el desarrollo, la implementación y la aplicación de la gestión y el aseguramiento de la calidad; además exigen el desarrollo y la implementación de un sistema documentado, que incluya la elaboración de manuales.



El vocabulario define un manual como un documento que enuncia la política del sistema y que describe el sistema de calidad de una organización. Este manual puede estar relacionado con las actividades totales de una organización o con una parte seleccionada de estas

El objeto es suministrar los lineamientos para la elaboración, la preparación y el control de manuales de sistemas ajustados a las necesidades específicas del usuario.

### Manuales de Sistemas

---



El manual de sistema debe referirse a procedimientos documentados del sistema destinados a planificar y gerenciar el conjunto de actividades que afectan la calidad dentro de una organización

Los manuales son elaborados y utilizados por una organización para:

- Comunicar la política del sistema, los procedimientos y los requisitos de la organización.
- Describir e implementar un sistema eficaz.
- Suministrar control adecuado de las prácticas y facilitar las actividades de aseguramiento.
- Suministrar las bases documentales para las auditorias.
- Adiestrar al personal en los requisitos del sistema.
- Presentar el sistema para propósitos externos: por ejemplo, demostrar la conformidad con las normas ISO 9001, 9002 ó 9003.
- Demostrar que el sistema cumple con los requisitos de la calidad exigidos en situaciones contractuales.

## ***Proceso de elaboración de un manual de sistema***

### **► Responsable en cuanto a la Elaboración**

El proceso en cuanto a la elaboración, con la asignación de la tarea de coordinación, es a un grupo delegado competente. Las actividades reales de redacción y transcripción deben ser ejecutadas y controladas por dicho grupo o por varias unidades funcionales individuales, según sea apropiado. El uso de referencias y documentos existentes puede acotar significativamente el tiempo de elaboración del manual, así como también ayudar a identificar aquellas áreas en las cuales existan deficiencias en el sistema que deba ser contemplados y corregidas.

### **► Uso de Referencias**

Siempre que sea apropiado se debe incorporar la referencia a normas o documentos que existen y estén disponibles para el usuario del manual.

### **► Exactitud y Adecuación**

El grupo competente delegado debe asegurar que el esquema del manual sea exacto y completo, y que la continuidad y el contenido del mismo sean adecuados.

## ***Proceso de aprobación, emisión y control del manual***

### **► Revisión y Aprobación Final**

Antes de que el manual sea emitido, el documento debe ser revisado por individuos responsables para asegurar la claridad, la exactitud, la adecuación y la estructura apropiada. La emisión de este manual debe ser aprobado por la gerencia responsable de su implementación y cada copia de este debe llevar una evidencia de su autorización.

### **► Distribución del Manual**

El método de distribución del manual debe proporcionar la seguridad de que todos los usuarios tengan acceso apropiado al documento. La distribución puede ser facilitada mediante la codificación de copias.

### **► Control de la Emisión y de los Cambios**

El control de la emisión y de los cambios del documento es esencial para asegurar que el contenido del manual está autorizado adecuadamente. Se pueden considerar diferentes métodos para facilitar el proceso físico de la realización de los cambios. En cuanto a la actualización de cada manual se debe utilizar un método para tener la seguridad de que cada poseedor del manual reciba los cambios y los incluya en su copia.

## ***Esquema del contenido de un manual***

- a) El título, el alcance y el campo de aplicación.
- b) La tabla de Contenido.
- c) Las páginas introductorias acerca de la organización y del manual
- d) Descripción de la estructura de la organización, las responsabilidades y autoridades.
- e) Descripción de los elementos del sistema.
- f) Definiciones, si es apropiado
- g) Guía para el manual, si es apropiado.
- h) Apéndice, si es apropiado.

Es importante mencionar que el orden del contenido del manual puede ser cambiado de acuerdo con las necesidades del usuario.

## ***¿Que incluir en un manual?***

### **► Título, Alcance y Campo de Aplicación**

El título y el alcance del manual deben definir la organización a la cual se aplica el manual. En esta sección también se deben definir la aplicación de los elementos del sistema. También es conveniente utilizar denegaciones por ejemplo, que aspectos no cumple un manual y en que situaciones no debería ser aplicado. Esta información puede ser localizado en la página del título.

### **► Tabla de Contenido**

Esta debe presentar los títulos de las secciones incluidas y como se pueden encontrar. La numeración de las secciones, subsecciones, páginas, figuras, ilustraciones, diagramas, tablas, etc., debe ser clara y lógica.

### **► Páginas Introductorias**

Las páginas introductorias de un manual deben suministrar información general acerca de la organización y del manual.

La información acerca de la organización debe ser su nombre, sitio, ubicación y los Medios de comunicación; también se puede adicionar información acerca de su línea de negocio y una breve descripción de sus antecedentes, su historia, su tamaño.

En cuanto a la información acerca del manual debe incluir la edición actual, la fecha de edición, una breve descripción de cómo se revisa y se mantiene actualizado el manual de sistema, una breve descripción de los procedimientos documentados utilizados para identificar el estado y para controlar la distribución del manual y también debe incluir evidencia de aprobación por aquellos responsables de autorizar el contenido del manual de sistema.



**► Descripción de la Organización, las Responsabilidades y las Autoridades**

*Esta sección suministra una descripción de la estructura de la organización de alto nivel. También puede incluir un organigrama de la organización que indique la responsabilidad, la autoridad y la estructura de interrelaciones.*

Igualmente subsecciones dentro de esta sección deben suministrar detalles de las responsabilidades, las autoridades y la jerarquía de todas las funciones que dirigen, desempeñan y verifican trabajos que afectan la calidad.

**► Definiciones**

*Esta sección debe ubicarse inmediatamente después del alcance y del campo de aplicación. Dicha sección debe contener las definiciones de los términos y conceptos que se utilicen únicamente dentro del manual. Las definiciones deben suministrar una comprensión completa, uniforme e inequívoca del contenido del manual. Es recomendable el uso de referencias, como por ejemplo, la norma ISO 8402.*

**► Guía para el Manual**

*Una guía puede suministrar una descripción de la organización del manual y un breve resumen de cada una de sus secciones.*

Con la ayuda de esta sección los lectores que están interesados solo en ciertas partes del manual deberían ser capaces de identificar, que parte del manual puede contener la información que está buscando.

**► Apéndice para la Información de Apoyo**

*Por último puede ser incluido un apéndice que contenga información de apoyo al manual.*

## Evaluación de la Implementación

La evaluación tiene por objetivo darle seguimiento a la implementación del sistema. Para ello se propone considerar las siguientes utilerías:

- » **De posesión:** para contestar a la pregunta ¿quién debe recibir la salida?. Esto es porque la información carece de valor en manos de alguien que carece de habilidad para utilizar esta información
- » **De forma:** para contestar a la pregunta ¿qué tipo de salida se distribuye entre quienes toman las decisiones?. Esto es así para considerar el formato y el lenguaje utilizado, recordando que la sobrecarga de información disminuye el valor del sistema informático.
- » **De lugar:** para contestar a la pregunta ¿dónde debe distribuirse la información?. La información debe llevarse al mismo lugar donde se toma la decisión, en otro lugar es inútil.
- » **De tiempo:** para contestar a la pregunta ¿cuándo debe proporcionarse la información?. La información debe llegar anticipadamente a la toma de decisiones, la información tardía no tiene ningún valor.

- » **De actualización:** implica que un sistema de información mantiene u valor si e mantiene una vez una vez que el equipo de sistemas a dejado la empresa, y si u uso proporciona resultados satisfactorios y duraderos.

Las respuestas a estas utilerías serán las que darán la pauta de cómo ha sido la implementación del sistema, y en general el desarrollo de todo el sistema.

## Mantenimiento

---



El objetivo de este paso es mantener el sistema puesto en marcha y evaluarlo periódicamente. Se dice que el mantenimiento debe perseguir por objetivo *mantener al sistema siempre vigente*

---

Un buen mantenimiento asegura una operación permanente. Pueden surgir nuevas correcciones /modificaciones motivadas por la evaluación del Área de Sistemas o nuevos requerimientos de usuarios. Algunos ejemplos son: nuevas salidas, extensión a otras áreas, corrección de errores, detección de desviaciones, verificación de normas impuestas. Si los cambios son muy profundos, tal vez justifiquen un nuevo Proyecto de Sistemas.



Recuerda que puedes ir completando el “Plan de Implementación” que esta en nuestro sitio, y que a través de un tutor, te va orientando en los contenidos a desarrollar.

## Autoevaluación

---

1. Mencione cuatro de las quejas más comunes que realizan los usuarios a los manuales que se le entregan con los sistemas.
2. Enumere las fuentes de adiestramiento para los usuarios del sistema.
3. ¿Por qué es importante tener bien definidos los objetivos del adiestramiento?
4. Presente ventajas y desventajas del entrenamiento in-situ.
5. ¿Cuáles son las estrategias de conversión o puesta en marcha de los sistemas nuevos y descartar los viejos?
6. Dé ejemplos de cuándo es conveniente usar cada una de las estrategias del punto anterior.
7. ¿Considera operativa la documentación para los desarrolladores?
8. Explique la diferencia entre Carga Inicial y Migración.

# ► Unidad V

## Prueba

### Objetivos

- Entender el método de evaluación.
- Comprender Técnicas de diseño de casos de pruebas.
- Comprender qué es una estrategia de pruebas y sus tipos.
- Establecer claras diferencias entre las distintas opciones del proceso de prueba.
- Realizar el análisis y aplicación de los conceptos vistos en la unidad en situaciones simuladas.

### Contenidos de la Unidad

Introducción a la Prueba  
La Prueba  
Propósito de la prueba  
Tipos de tests  
Niveles de Prueba

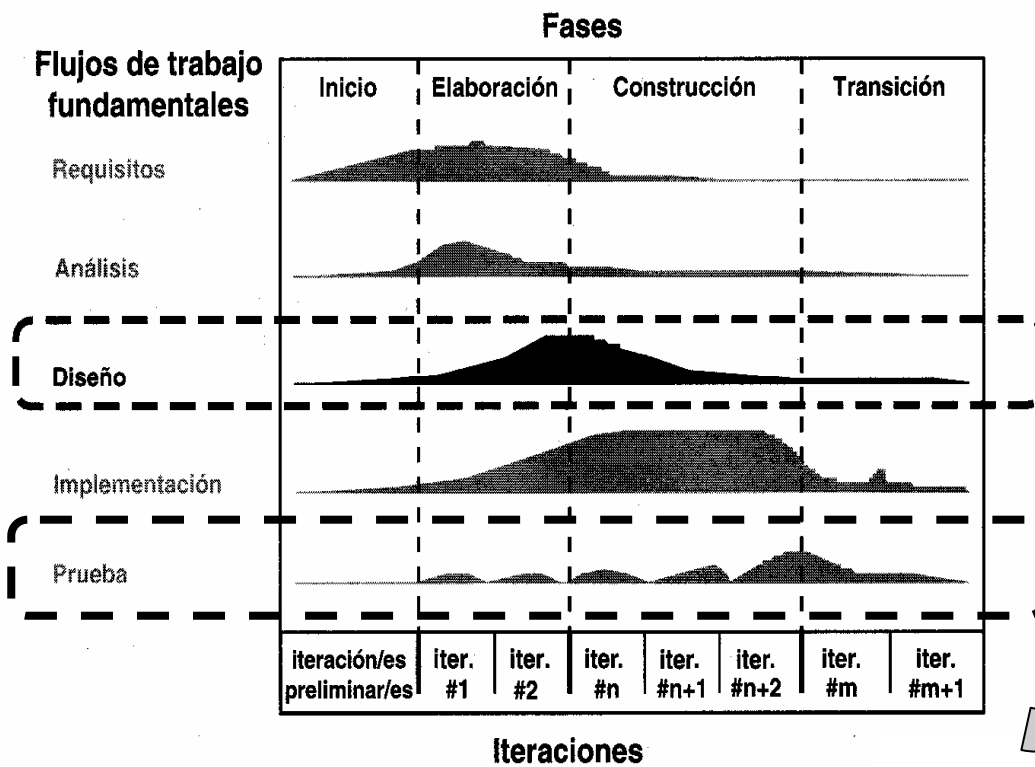
- Prueba de unidad
- Pruebas de Integración
- Prueba de Sistema

Estrategias de Prueba  
El Proceso de Prueba  
Planificación de la prueba  
Identificación de la Prueba  
Especificación de la prueba  
Ejecución de las Pruebas  
Análisis de Error  
Autoevaluación

## Mapa Conceptual

# Proceso Unificado de desarrollo de Software

## Ciclo de Vida



## Introducción a la Prueba

La evaluación de un producto es relativamente independiente del método utilizado para la construcción. En este sentido y en referencia al paradigma orientado a objetos, este proceso no es muy diferente al testeo de sistemas desarrollados con otras metodologías. Un método orientado a objetos proporciona nuevas posibilidades, pero también nuevos problemas. En cierta forma la tarea de evaluar se ve simplificada, ya que como el sistema esta compuesto por objetos que contienen datos y comportamientos, estos pueden formar una unidad que puede ser evaluada individualmente.

La fase de evaluación pretende verificar que lo diseñado esté de acuerdo con la especificación requerida. En sí, el objetivo de las actividades de evaluación es solamente asegurar una calidad certificada para el producto.

Así las actividades de testeo normalmente se dividen en verificación y validación. La verificación chequea que el resultado coincide con la especificación. Sin embargo esto no garantiza la satisfacción del cliente. La validación chequea si el resultado realmente es el que se desea.

La validación se obtiene principalmente por medio de un análisis de requerimientos completo, incluyendo, la participación activa de los clientes. Una herramienta firme para la validación son los Caso de uso.

## La Prueba

La prueba es bastante independiente del método de desarrollo utilizado. El enfoque Orientado a Objetos da nuevas posibilidades y también nuevos problemas. La actividad de prueba puede normalmente dividirse en:

Verificación: ¿Estamos construyendo el sistema CORRECTAMENTE?

Validación: ¿Estamos construyendo el sistema CORRECTO?

En este momento discutiremos la *verificación* porque la validación se resuelve con la determinación de requerimientos, por el uso de prototipos, etc.

## Propósito de la prueba

Se definirán en primer lugar algunos conceptos:

Falla: Cuando un programa funciona mal.

Falta: Existe en el código del programa. Puede provocar una falla.

Error: Acción humano que resulta en software que contiene una falla.

La primera lección a aprender es que no se puede probar que el sistema no tenga falta, sin que esté libre de fallas.



El propósito de la prueba es encontrar fallas.

La prueba es un proceso destructivo, tener que indagar sobre lo que hicimos para detectar lo que hicimos mal. Es conocido que la corrección de una falla provoca fallas adicionales, en consecuencia, si una falla aparece debemos probar todo el software.

## Tipos de tests

El siguiente es un resumen de varios tipos de tests. Ninguno es independiente de los otros, cuando se realiza la prueba de un sistema, se usan en combinación.

<b>Tests de Operación</b>	Es el más común. El sistema es probado en operación normal. Mide la confiabilidad del sistema y se pueden obtener mediciones estadísticas.
Ejecutamos el sistema al <i>máximo</i> , todos los parámetros enfocan a valores máximos, todos los equipos conectados, usados por muchos usuarios ejecutando caso de usos simultáneamente.	<b>Tests de Escala Completa</b>
<b>Tests de Performance o de Capacidad</b>	El objetivo de esta prueba es medir la capacidad de procesamiento del sistema. Los valores obtenidos (medidos) son comparados con los requeridos
Cumple la función de determinar cómo se comporta el sistema cuando es <i>sobrecargado</i> . No se puede esperar que supere esta prueba, pero sí que no se venga abajo, que no ocurra una catástrofe. Cuántas veces se cayó el sistema es una medida interesante.	<b>Tests de Sobrecarga</b>
<b>Tests Negativos</b>	El sistema es sistemática e intencionalmente usado en forma incorrecta. Este maltrato debe ser planeado para probar casos especiales.
Estos tests son los que pueden mapearse (rastrear), directamente desde la especificación de requerimientos.	<b>Tests basados en requerimientos</b>
<b>Tests Ergonómicos</b>	Son muy importantes si el sistema será usado por gente inexperta. Se prueban cosas como: <ul style="list-style-type: none"> <li>» Consistencia de la interfaz.</li> <li>» Consistencia entre las interfaces de los distintos casos.</li> <li>» Si los menús son lógicos y legibles.</li> <li>» Si se entienden los mensajes de falla.</li> </ul>
Con el estilo y características del anterior, se prueba la documentación del sistema.	<b>Tests de Documentación del Usuario</b>
<b>Test de Aceptación</b>	Este test es ejecutado por la organización que solicita el sistema. El sistema es probado en un entorno real usualmente llamado <i>A/fa</i> . Cuando no hay un usuario que solicita el producto se usan las pruebas <i>Beta</i> , que son encargadas a clientes selectos antes de liberar la versión

## Niveles de Prueba

Los niveles de prueba se mencionan ahora en forma general, para luego profundizarlos.

- ▶ Prueba de Unidad: se prueban las clases, bloques, paquetes.
- ▶ Prueba de Integración: el objetivo es probar que las unidades trabajan correctamente juntas.
- ▶ Prueba de Sistema: se prueba el sistema completo. Requiere la colaboración de un usuario final y de casos de prueba típicos.

### ▶ Prueba de unidad

La prueba de unidad involucra: clases, bloques, paquetes de servicio. En sistemas tradicionales: procedimientos, subrutinas.

Las pruebas de unidad de sistemas orientados a objeto son más complejas; conceptos como la herencia, el polimorfismo, etc., hacen más compleja la prueba.

Los requerimientos para depuradores son mayores para sistemas Orientados a Objeto. Normalmente entornos como Smalltalk, C++, Simula, contienen soportes para o inspeccionar la estructura de los objetos durante la ejecución.

La prueba de unidad consiste de:

#### **Prueba de Especificación o Caja Negra**

Verifican el comportamiento de la interfaz de la unidad. *Lo que hace* sin importar *cómo*. Es importante ver no solamente si se produce una salida, sino verificar que la misma sea correcta. También podemos probar la relación entrada/salida en diferente estados de la unidad, pero eso se hace con los casos de prueba basados en estados. Como las unidades sólo se comunican con interfaces definidas, las pruebas de especificación son bastante directas. Hemos definido cuáles operaciones soporta la unidad y qué comportamiento debería mostrar para cada operación.

#### **Prueba Estructural o de Caja Blanca**

Se verifica si la estructura interna es la correcta. Todos los caminos posible planteados en el código deben ser contemplados y ejecutados (los llamados caminos de decisión a decisión). Dado que los casos de prueba estructurales y los basados en estado pueden modificar la estructura del código, es preferible hacer la prueba estructural al último. Es casi imposible recorrer absolutamente todos los caminos posibles, considerando los parámetros y los valores de las variables. Los depuradores son de gran ayuda.

#### **Prueba Basada en estados**

Prueba la interacción entre las operaciones de una clase, monitoreando los cambios que tienen lugar en los atributos de los objetos, probar sólo una operación aislada no es suficiente para probar una unidad. Se deben probar también los atributos del

objeto, puesto que si persisten entre invocaciones de diferentes operaciones, y posiblemente a través de interacciones entre las operaciones. Es importante basarse en los diagramas de transición de estados, así, al menos cada estado es visitado por lo menos una vez y cada transición es atravesada al menos una vez.

La matriz de estados es una buena herramienta para este tipo de pruebas. La combinación de estados y estímulos puede probarse con esta matriz.

Estados →					
Estímulos ↓	S0	S1	S2	S3	S4
Estímulo 1	OK	OK	Respuesta Equivocada	OK	Falla
Estímulo 2		Falla	OK		
Estímulo 3		OK	Lento controlarlo		
Estímulo 4					

Una de las ventajas de este tipo de matrices es que focaliza la atención del diseñador en la combinación estímulo /estado que puede ser descuidada durante el diseño. Es posible incluir todas las combinaciones de atributos del objeto (todos los posibles valores de variables) y todas las variantes de estímulos (distintos parámetros). En general algunas combinaciones específicas de atributos pueden ser más interesantes que otras.

Algunas operaciones, como las de lectura, que no afectan el estado, no deben ser consideradas.

Debe verificarse que todos los posibles estados pueden alcanzarse con alguna combinación de operaciones, de otra manera puede haber una falla en el diseño de la clase.

## ► Pruebas de Integración

Una vez que las unidades han sido certificadas en las pruebas de unidad, estas unidades deberían integrarse en unidades más grandes y finalmente al sistema. El propósito de las pruebas de integración es determinar si las distintas unidades que han sido desarrolladas trabajan apropiadamente, juntas.

Aquí se incluyen pruebas de paquetes de servicio, de caso de usos, subsistemas y el sistema completo. Consecuentemente no hay una sola prueba de integración en un desarrollo, por el contrario, se realizan varias a distintos niveles.

Estas pruebas son necesarias porque:



Al combinar las unidades pueden aparecer nuevas fallas.  
La combinación aumenta exponencialmente el número de caminos posibles.



Por lo tanto hay fallas que no podrían detectarse de otra forma. Nuevamente aquí, los caso de usos se transforman en la herramienta que conduce la prueba de integración. Se puede comenzar la prueba de caso de usos tan pronto como la prueba de los bloques que los componen hayan sido certificados y aprobados.

Las bases para la especificación de estas pruebas vienen desde los diagramas de interacción, allí se ve claramente la interacción entre usuarios y el sistema y entre los objetos (bloques) del sistema.

Las pruebas de integración se hacen probando cada Caso de uso, uno a la vez desde dos puntos de vista:

- ▶ **Uno interno:** basado en los diagramas de interacción.
- ▶ **Uno externo:** basado en las descripciones del modelo de requerimientos.

Cada caso de uso, entonces, corresponde a un conjunto de especificaciones de prueba. Dividimos los tests en diferentes tipos, para un caso de uso debemos hacer las siguientes pruebas:

- » **Pruebas del curso básico.**
- » **Pruebas de cursos alternativos.**
- » **Pruebas de documentación de usuarios.**

Cada tipo de test es ahora dividido en subtests con diferentes condiciones.

Cuando se prueban los caso de usos puede ocurrir que algunos de ellos no puedan probarse solos, si no que necesiten de algún otro caso de uso para ser significativo.

Los casos de prueba no deben sobrecargarse incluyendo diferentes configuraciones de instanciaciones de clases, o diferentes configuraciones del sistema final.

Normalmente las pruebas de integración (desde el nivel de subsistema para arriba) se realizan con un equipo de prueba. Aquí la documentación es más formal que en las pruebas de unidad. Usualmente a las pruebas, el equipo las realiza en un entorno en el cual el sistema se ejecutará cuando este en operación.

## ▶ **Prueba de Sistema**

Una vez que se han probado todo los caso de usos por separado se probará el sistema completo. Algunos caso de usos son ejecutados en paralelo y el sistema es sometido a diferentes cargas.

Las pruebas de sistema pueden dividirse en los siguientes tests:

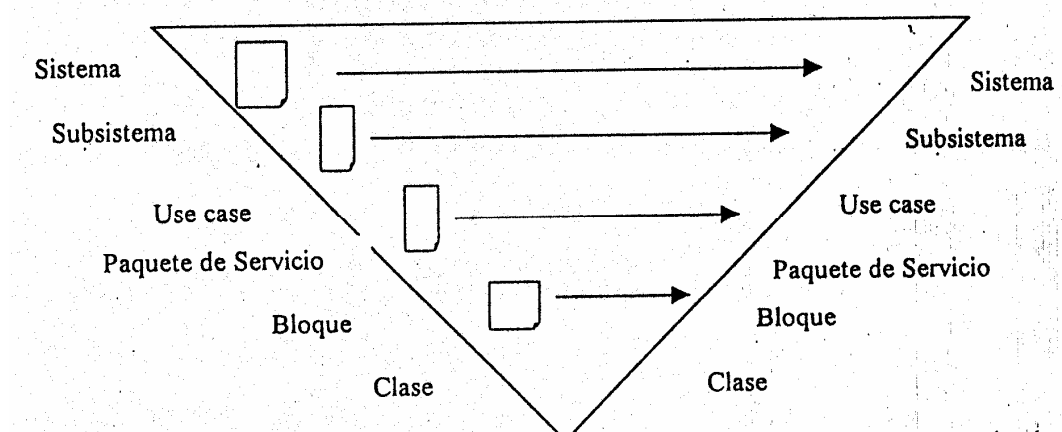
- ▶ Tests de operación
- ▶ Tests de escala completa
- ▶ Tests negativos
- ▶ Tests basados en especificación de requerimientos
- ▶ Tests de documentación del usuario

Cuando probamos el sistema, los caso de usos debería probarse en paralelo sincronizada y desincronizadamente. Se puede forzar el sistema corriendo varios caso de usos a la vez.

## Estrategias de Prueba

Las estrategias de prueba pueden realizarse de varias maneras, pero lo más común es hacerlas en orden inverso al que se realiza el diseño y la implementación.

La verificación se hace en varios niveles, veamos la siguiente figura:



Sin embargo podemos ir realizando pruebas conforme vamos terminando los diseños, es decir la implantación y la prueba se realizan en forma intercalada e incremental.

Puesto que podemos desarrollar de varias maneras: Top-down, Botton-Up, por caso de uso, podemos hacer estos incrementos usando las mismas estrategias.

Si se asume un enfoque Top-Down en el diseño, esto significa que primero desarrollamos las interfaces entre subsistemas, las que retornan valores que son controlados y posteriormente se reemplaza con el código real, esto permite probar el flujo completo en niveles superiores antes y luego ir a niveles inferiores.

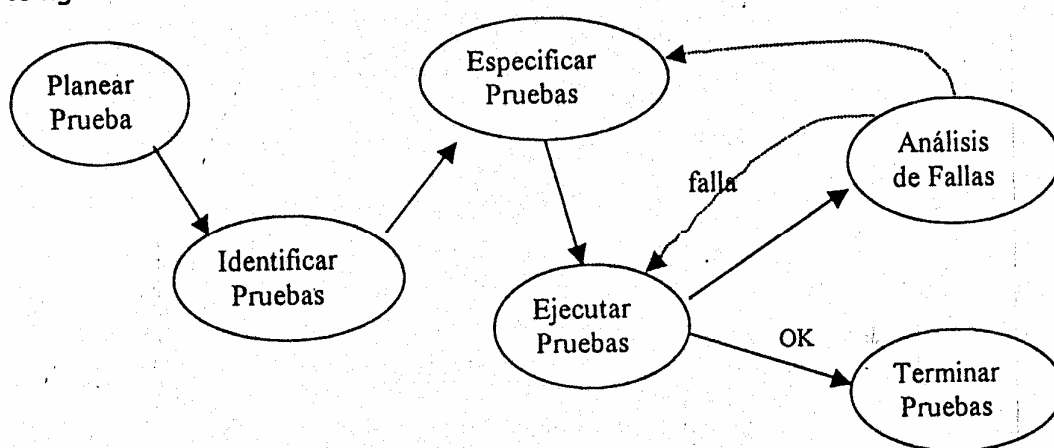
Las pruebas también pueden hacerse botton-up o por caso de usos básicos. El enfoque botton-up es preferible en los niveles más bajos, cuando la primera unidad está certificada y los clientes directos pueden ser certificados. Entonces el siguiente nivel de clientes puede ser certificado y así sucesivamente.

Esta técnica minimiza las necesidades de implementar clases piloto sólo para prueba, puesto que las unidades certificadas trabajan como servidores. Esto da una convergencia suave entre pruebas de unidad y de integración. Sin embargo, la detección de fallas en las unidades servidoras podría forzar a comenzar todo el proceso nuevamente.

## El Proceso de Prueba

Es importante planear la prueba, la prueba no es algo que uno haga en forma improvisada. El proceso de prueba es un proceso que en gran medida corre en paralelo con otros procesos.

Los procesos de prueba son un conjunto de actividades que se describen en la siguiente figura:



## Planificación de la prueba

La actividad de prueba comienza pronto en el proceso de desarrollo. La planificación puede comenzar cuando comenzamos el desarrollo, en general durante el análisis, pero no podemos preparar nada hasta no comenzar la construcción.

Los lineamientos de la prueba se establecen con anticipación, determinando el método y nivel de ambición, se crean las bases de la prueba.

Debería determinarse sí:

- » Las pruebas se hacen mecánica o automáticamente.
- » Hacer una estimación de recursos que se requerirán, y estas decisiones deberían reflejarse en el Plan de Proyecto.
- » Se estudia si existen programas de prueba y datos que puedan usarse, se deberían modificarse o crearse nuevos.

Usando estos lineamientos como base podemos determinar qué grado de cobertura tendrían los tests.

- ▶ Nunca comenzar las pruebas de integración antes que las de unidad.
- ▶ Se puede probar incrementalmente, agregando nueva funcionalidad a los bloques mientras se avanza.
- ▶ El plan no debe controlar los detalles de la prueba, sólo servir como base para las actividades de la prueba.

- ▶ Un registro de la prueba se debe mantener durante el proceso de prueba completo.
- ▶ El registro debería conectarse a la versión del sistema. El propósito del registro es mantener una breve historia de las actividades de prueba.
- ▶ El registro es archivado al finalizar las pruebas y sirve de base para el refinamiento del proceso de prueba y para la planificación de nuevos tests.

## Identificación de la Prueba

---

*Cuando identificamos lo que debería probarse, se pueden estimar, también los recursos requeridos. Es una estimación más detallada que la hecha anteriormente, y actúa como un principal lineamiento para la especificación y ejecución de la prueba.*

Esto requiere la configuración y determinación del equipamiento que será requerido para la prueba, tarea que se realiza junto con la empresa, para que estén en condiciones en el momento que se la requiera.

Cuando los recursos de la prueba son restringidos, cada caso de prueba debe maximizar la probabilidad estadística de detección de fallas. Se debería encontrar las fallas mayores primero.

## Especificación de la prueba

---

Cuando se identifica cuales subtests se harán, se especifican a *nivel funcional* donde describiremos la prueba y su propósito de manera general, y en un *nivel detallado*, donde describiremos exactamente cómo será ejecutado. La última parte incluye una descripción procedural completa de cada paso en la prueba.

Aquí nuevamente las descripciones de los caso de usos son una herramienta poderosa. El propósito de la especificación de la prueba es dar a las personas que no están familiarizadas con la prueba, o aún con el sistema, instrucciones detalladas para correr los casos de prueba.

Cada caso de prueba debe documentarse, para facilitar el reuso en los tests de regresión y tal vez en otras versiones del sistema. Deberían especificarse condiciones de prueba tales como: hardware, software, equipamiento de prueba. Debe indicarse también como se debe ejecutar la prueba, en qué orden, salidas esperadas y criterios para aprobar el test.

Cuando se escriben los tests de especificación, también se preparan los reportes requeridos para informar los resultados de la prueba. El esqueleto de los reportes se prepara con anticipación.



*Las pruebas ayudan a detectar faltas, si se encuentran faltas, pueden corregirse a nivel de diseño no solamente en el código.  
Es interesante enfocar las pruebas asumiendo que el sistema tiene fallas, y poder determinar cuántas horas hombre son necesarias para poder detectar nuevas fallas.*

## Ejecución de las Pruebas

Cuando ejecutamos las pruebas usamos la especificación de pruebas y los reportes de prueba preparados. La estrategia es probar lo que más se pueda en paralelo, aunque sea difícil.

Las pruebas se realizan en forma manual o automatizada, según se haya especificado.

Las especificaciones indican el resultado esperado. Si alguna de las pruebas falla se registra, se interrumpe la ejecución, y el defecto es analizado y corregido si se puede. Luego la subprueba se ejecuta nuevamente.

Al finalizar la prueba se analizan los resultados. Si está aprobado o no. Este análisis resulta en reportes de prueba. Los reportes contienen en forma resumida el resultado individual de cada subtest y uno final, los recursos gastados y si el test está aprobado o no.

Si se descubrieron cuellos de botella también deben registrarse y mostrarse.



Recuerda que puedes ir completando el “Plan de Prueba” que esta en nuestro sitio, y que a través de un tutor, te va orientando en los contenidos a desarrollar.

## Autoevaluación

1. Con sus propias palabras establezca la diferencia entre verificación y validación. ¿Utilizan las dos el diseño de casos de prueba y las estrategias de prueba?
2. Liste algunos problemas que pueden darse asociados con la creación de un grupo independiente de prueba.
3. Si solo pudiera seleccionar tres métodos de diseño de casos de prueba, para aplicarlos durante la prueba de unidad ¿Cuáles serían? ¿por qué?
4. ¿Es posible o incluso deseable la prueba de unidad en cualquier circunstancia? Proponga ejemplos que justifiquen su respuesta.



# ► Anexo A

## **Integramos los Flujos de trabajo del PDUS**

### **Objetivos**

- Integrar los contenidos teóricos prácticos del Proceso Unificado de Desarrollo de Software.
- Integrar tres nuevos conceptos: Planificar, Riesgos y Evaluación al ciclo de vida del PDUS.

### **Contenido de la Unidad**

Introducción

El flujo de trabajo de iteración genérico

La necesidad de equilibrio

Las fases son la primera división del trabajo

“El planificar” precede “al hacer”

Planear las cuatro fases

Plan de iteraciones

Pensar a largo plazo

Planear los criterios de evaluación

Evaluar las iteraciones y las fases

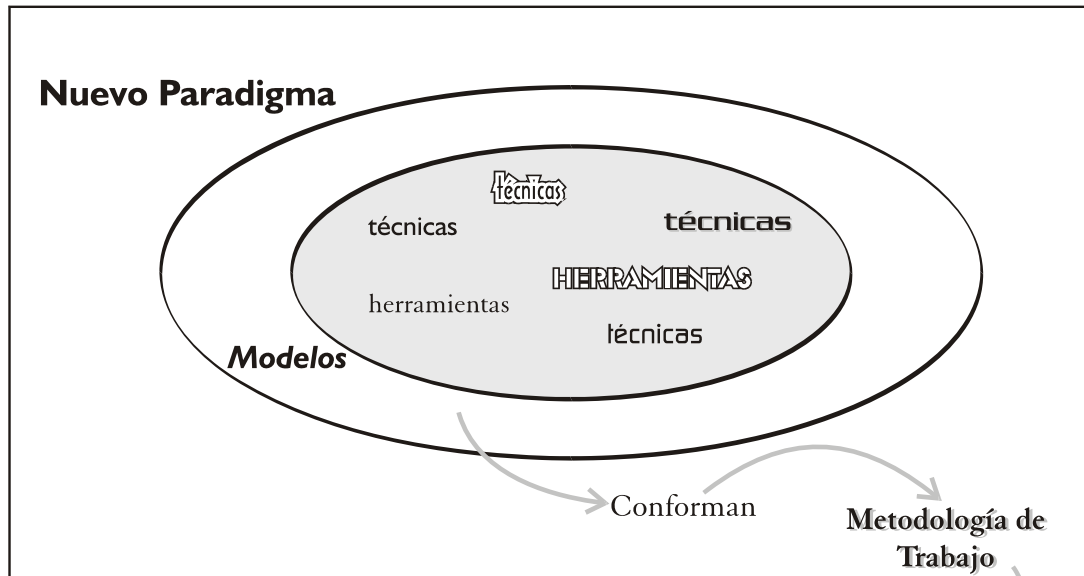
Criterios no alcanzados

La siguiente iteración

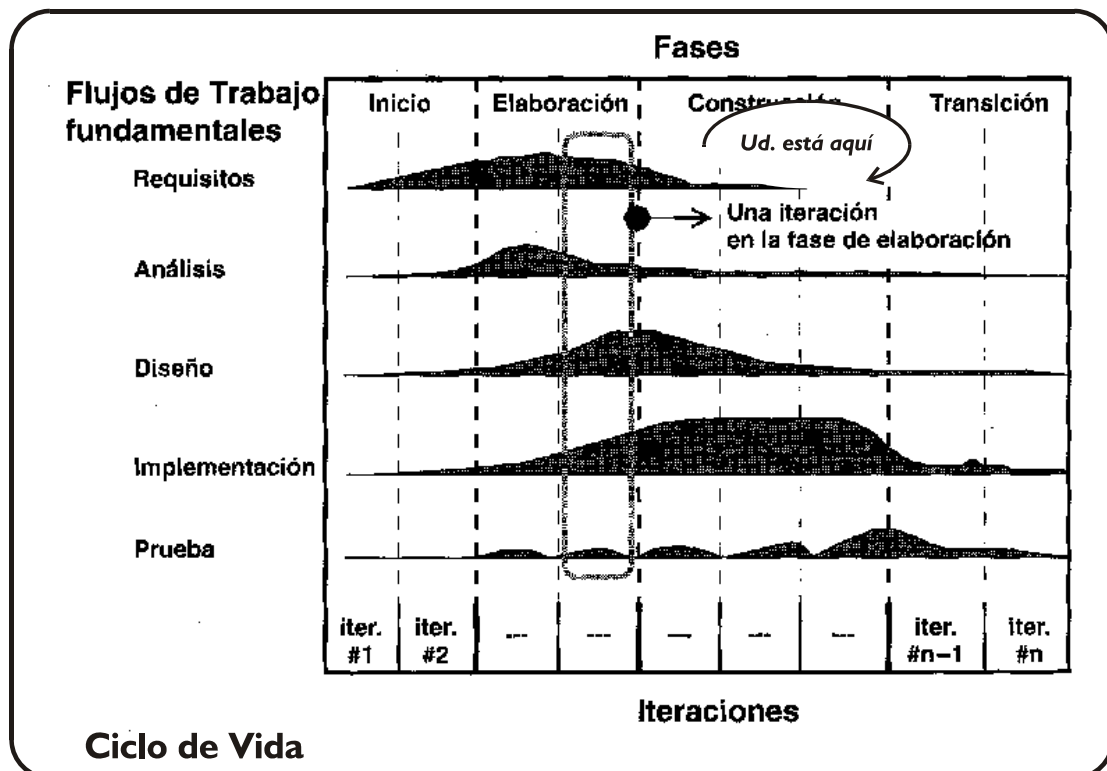
Evolución del conjunto de modelos

Plan de Aceptación del Producto

## Mapa Conceptual



## Proceso Unificado de Desarrollo de Software





## Introducción

En esta unidad retomamos la idea de la iteración genérica expresada en el módulo de Sistemas III, unidad II. La intención de esta unidad es encontrar el patrón común que caracteriza a todas las iteraciones a partir de las distintas iteraciones que ocurren durante las cuatro fases.

Utilizamos este patrón genérico como base sobre la que construir las iteraciones concretas; el contenido de una iteración cambia para acomodarse a los objetivos particulares de cada fase.

El flujo de trabajo de iteración genérica incluye los cinco flujos de trabajo: requisitos, análisis, diseño, implementación y prueba. Éste también incluye la planificación, que precede a los flujos de trabajo, y la evaluación, que va detrás de ellos.

En esta unidad nos centraremos en la planificación, en la evaluación y en otras actividades comunes a todos los flujos de trabajo, buscando integrar todos los contenidos vistos.

La planificación es necesaria a lo largo de todo el ciclo de desarrollo, pero antes de que podamos planear es necesario saber qué es lo que tenemos que hacer; los cinco flujos de trabajo fundamentales proporcionan un punto de partida.

Por supuesto, un plan no estará completo sin la estimación de recursos que éste requerirá y, por último, también llevarse a cabo la evaluación de la ejecución de cada iteración y fase.

## El flujo de trabajo de iteración genérico

### *La necesidad de equilibrio*

En cada uno de los momentos en el ciclo de vida de un proyecto de desarrollo software están activas muchas secuencias de actividades diferentes: trabajamos en nuevas funciones, diseñamos la arquitectura, recibimos comentarios de los usuarios, mitigamos riesgos, planificamos el futuro, etc. Debemos equilibrar y sincronizar en todo momento estas secuencias de actividades diferentes para controlar así esta complejidad.

Los desarrolladores dividen el trabajo, que es abrumadoramente complejo como un todo, en partes más pequeñas y comprensibles. A lo largo del ciclo de vida de desarrollo dividen el trabajo en fases, y cada una de estas fases en iteraciones. Dentro de cada iteración el proyecto se esfuerza en alcanzar un equilibrio entre las secuencias de actividad que se ejecutan a lo largo de la iteración, lo que significa que deberíamos trabajar en las cosas apropiadas en cada iteración. Las cosas correctas en las que trabajar en cada momento dependen del punto del ciclo de vida en que nos encontramos; la tarea de un proyecto es seleccionar las cosas correctas en las que trabajar en cada secuencia de actividad.

En las primeras iteraciones trabajamos con riesgos críticos, casos de usos fundamentales, cuestiones arquitectónicas, la elección del entorno de desarrollo, todas las cuestiones orientadas a la investigación, mientras que en las últimas iteraciones trabajamos en actividades orientadas al desarrollo, como la implementación y la prueba, problemas de evaluación del rendimiento y el despliegue del sistema.

El relacionar todas estas actividades entre sí es una cuestión de equilibrio bastante delicada, lo que hace que el desarrollo de software sea una tarea extraordinariamente difícil.

Lo que hacemos en una iteración es entender estas secuencias de actividades diferentes y buscar el equilibrio entre ellas.

## Las fases son la primera división del trabajo

---

### ► La fase de inicio establece la viabilidad

---

El objetivo principal de esta fase es establecer el análisis de negocio -el análisis para decidir si se sigue adelante con el proyecto-, aunque este análisis se seguirá desarrollando en la fase de elaboración conforme se vaya disponiendo de más información. La fase de inicio no es un estudio completo del sistema propuesto, sino que en ella buscamos el porcentaje de casos de uso necesarios para fundamentar el análisis de negocio inicial.

Para realizar este análisis seguimos cuatro pasos:

**1-** Delimitar el ámbito del sistema propuesto, es decir, definir los límites del sistema y empezar a identificar las interfaces con sistemas relacionados que están fuera de los límites.

**2-** Describir o esbozar una propuesta de la arquitectura del sistema y en especial de aquella parte del sistema que sea nueva, arriesgada o difícil. En este paso sólo, llegamos hasta una descripción de la arquitectura, raramente hasta un prototipo ejecutable; esta descripción de la arquitectura consiste en unas primeras versiones de los modelos. Esta arquitectura no es construida en esta fase, simplemente hacemos posible el que se pueda crear una.

**3- Identificar riesgos críticos**, es decir, los que afectan a la capacidad de construir el sistema, y determinar si podemos encontrar una forma de mitigarlos, quizás en una etapa posterior. En esta fase consideramos sólo los riesgos que afectan la viabilidad, es decir, aquellos que amenazan el desarrollo con éxito del sistema. Cualquier riesgo no crítico que se identifique es colocado en la lista de riesgos para su posterior consideración detallada en la fase siguiente.

**4-** Demostrar a usuarios o clientes potenciales que el sistema propuesto es capaz de solucionar sus problemas o de mejorar sus objetivos de negocio **construyendo un prototipo**. En la fase de inicio podemos construir un prototipo para mostrar una solución al

problema de los clientes o usuarios potenciales, el cual demuestra las ideas básicas del nuevo sistema haciendo énfasis en su uso -interfaces de usuario o algún algoritmo nuevo interesante-.

Continuamos con estos esfuerzos hasta el momento en que desarrollar el sistema parece ser rentable económicamente, es decir, hasta que se concluye que el sistema proporcionará ingresos u otros beneficios proporcionales a la inversión necesaria con un margen suficiente para construirlo. En otras palabras, hemos realizado una primera versión del análisis de negocio, el cual será refinado en la fase siguiente, la de elaboración.

La intención es minimizar los gastos en tiempo de planificación, esfuerzo y fondos en esta fase hasta que decidamos si el sistema es viable o no.

- » En el caso de un sistema completamente nuevo en un dominio poco explorado esta consideración puede llevar un tiempo y un esfuerzo considerables y puede extenderse a varias iteraciones.
- » En el caso de un sistema bien conocido en un dominio establecido o en el caso de tratarse de la extensión de un sistema a una, nueva versión, los riesgos y casos desconocidos pueden ser mínimos, permitiendo que esta primera fase se complete en pocos días.

### ► **La fase de elaboración se centra en la factibilidad**

El resultado principal de la fase de elaboración es una arquitectura estable para guiar el sistema a lo largo de su vida futura. Esta fase también lleva el estudio del sistema propuesto al punto de planificar la fase de construcción con gran precisión.

Con estos dos grandes objetivos: la arquitectura y la estimación de costes con gran precisión, el equipo hace lo siguiente:

**1-** Crea una línea base para la arquitectura que cubre la funcionalidad del sistema significativa arquitectónicamente y las características importantes para las personas involucradas.

**2- Identifica los riesgos** significativos, es decir, los riesgos que podrían perturbar los planes, costes y planificaciones de fases posteriores, y los reduce a actividades que pueden ser medidas y presupuestadas.

**3-** Especifica los niveles a alcanzar por los atributos de calidad, como la fiabilidad (porcentajes de defectos) y los tiempos de respuesta.

**4- Recopila casos de uso** para aproximadamente el 80 por ciento de los requisitos funcionales, suficiente para planificar la fase de construcción.

**5-** Prepara una propuesta de la planificación cubierta, personal necesario y coste dentro de los límites establecidos por las prácticas de negocio.

## ► **La fase de construcción construye el sistema**

---

El objetivo general de la fase de construcción viene indicado por su tarea fundamental: la capacidad de operación inicial. Es decir, un producto listo para ser distribuido como versión beta y ser sometido a pruebas.

Esta fase emplea más personal a lo largo de un periodo de tiempo más largo que ninguna otra fase, y es por esto por lo que es tan importante tener todos los detalles importantes bien preparados antes de empezar con la construcción. Generalmente esta fase requiere un mayor número de iteraciones que las fases anteriores.

Una de las grandes ventajas de construir software utilizando un enfoque que use múltiples fases y un desarrollo iterativo e incremental es que nos permite equilibrar la asignación de recursos y de tiempo a lo largo del ciclo de vida

Entre las actividades de la fase de construcción tenemos:

**1-** La **extensión de la identificación, descripción y realización de casos de uso** a todos los casos de uso.

**2-** La **finalización del análisis** (posiblemente quedan todavía más de la mitad de los casos de uso por ser analizados cuando comienza la fase de construcción), del diseño, de la implementación y de la prueba (posiblemente queda un 90 por ciento).

**3-** El **mantenimiento de la integridad de la arquitectura**, modificándola cuando sea necesario.

**4-** La **monitorización de los riesgos críticos y significativos**, arrastrados desde las dos primeras fases, y su mitigación si se materializan.

## ► **La fase de transición se mete dentro del entorno del usuario**

---

La fase de transición comienza a menudo con la entrega de una versión beta del sistema es decir, la organización distribuye un producto software capaz ya de un funcionamiento inicial a una muestra representativa de la comunidad de usuarios. El funcionamiento del producto entorno de los usuarios es frecuentemente una prueba del estado de desarrollo del producto, mas severa que el funcionamiento en el entorno del que lo desarrolla.

Entre las actividades de la transición se incluyen:

**1-** Preparar las actividades, como la **preparación del lugar**.

**2- aconsejar al cliente** sobre la actualización del entorno (hardware, sistemas operativos, protocolos de comunicaciones, etc.) en los que se supone que el software va a funcionar.

**3- Preparar los manuales** y otros documentos para la entrega del producto. En la fase de construcción se prepara una documentación preliminar para los usuarios de las versiones beta

**4- Ajustar el software** para que funcione con los parámetros actuales del entorno del usuario.

**5- Corregir los defectos** encontrados a lo largo de las pruebas realizadas a la versión beta

**6- Modificar el software** al detectar problemas que no habían sido previstos.

**7-** La fase de transición termina con la **entrega del producto final**. Sin embargo, antes de que el equipo del proyecto abandone el proyecto, los líderes del equipo llevan a cabo un estudio del sistema con los siguientes objetivos:

- ▶ Encontrar, discutir, evaluar y registrar las "lecciones aprendidas" para referencias futuras
- ▶ Registrar asuntos útiles para la entrega o versión siguiente.

## **“El planificar” precede “al hacer”**

Cuando comenzamos la fase de inicio sabemos tres cosas:

- » Vamos a llevar a cabo el proyecto en una serie de iteraciones en cuatro fases.
- » Tenemos la información sobre el sistema propuesto que nuestros predecesores recopilamos (y que llevaron a comenzar un proyecto).
- » Tenemos nuestra propia información básica sobre el dominio y sobre sistemas similares en los que trabajamos en el pasado.

Partiendo de esta información hemos de planear:

**El proyecto (plan de proyecto)**

**Cada iteración (plan de iteración)**

Al principio, con información limitada con la que trabajar, ambos planes contienen poco detalle, pero conforme trabajamos en las fases de inicio y de elaboración los planes llegan a estar más detallados.

## ***Planear las cuatro fases***


Gracias a las indicaciones del Proceso Unificado sabemos qué hay que hacer en cada una de las fases. Nuestra tarea, en el plan del proyecto, es reducir estas indicaciones a términos concretos:



Asignaciones de tiempo. Decidimos cuánto tiempo asignar a cada fase y la fecha en la que cada fase ha de haber sido completada. Estos tiempos, aunque establecidos con

precisión, pueden ser bastante inestables al principio de la fase de inicio, pero se irán fijando conforme vayamos aprendiendo más.

✓ **Hitos principales.** Una fase termina cuando ha sido alcanzado el criterio actual. Al principio estos criterios pueden ser poco más que creencias basadas en alguna información, información que puede venir de nuestra experiencia pasada en el dominio, de las similitudes del sistema propuesto con sistemas anteriores, de las especificaciones del rendimiento que se pretende que el nuevo sistema alcance y de la capacidad de nuestra organización de desarrollo de software.

 **Plan de proyecto.** El plan de proyecto esboza un "mapa de carreteras" global, que cubre la planificación, las fechas y criterios de los objetivos principales y la división de las fases en iteraciones.

Uno puede esperar que la primera iteración en la fase de inicio sea difícil, ya que además de la iteración misma uno ha de cubrir lo siguiente:

- » Ajustar el Proceso Unificado al proyecto y seleccionar herramientas para automatizar el proceso.
- » Empezar a reunir a gente con el talento necesario para el proyecto. Crear las relaciones que dan lugar a un equipo efectivo.
- » Entender el dominio, que a menudo es desconocido para el equipo.
- » Percibir la naturaleza del proyecto, que será más difícil si se trata de un desarrollo nuevo, que si se trata de la extensión de un producto ya existente.
- » Familiarizar al equipo con las herramientas apropiadas para el proceso y el proyecto.

## ***Plan de iteraciones***

Cada fase está formada por una o más iteraciones. La planificación de las iteraciones pasa por un conjunto de pasos comparables con los seguidos en la planificación de las fases:

**Planificación de iteración.** Decidimos cuánto tiempo puede requerir cada iteración y su fecha de terminación; al principio a *grosso modo* y después cada vez con más precisión conforme aprendemos

**Contenido de iteración.** Mientras que el plan del proyecto esboza las iteraciones planeadas en términos generales, cuando la fecha de comienzo de una iteración se acerca, planeamos lo que ha de hacerse más en detalle. El contenido de una iteración se basa en lo siguiente:

- ▶ Los casos de uso que tienen que ser completados al menos parcialmente durante la iteración
- ▶ Los riesgos técnicos que han de ser identificados, transformados en casos de uso y mitigados en ese momento.

- ▶ Los cambios que han sufrido los requisitos o los defectos que pueden haber sido corregidos.
- ▶ Los subsistemas que han de ser implementados parcial o completamente. Este punto varía dependiendo de la fase en que se considere. Por ejemplo, en la fase de elaboración identificamos la mayoría de los subsistemas y todas las clases significativas arquitectónicamente; en cambio, en la fase de construcción completamos los subsistemas con más comportamiento, resultando en componentes más completos.

### ***Pensar a largo plazo***

Durante el largo periodo de tiempo que el sistema puede durar, éste puede presenciar grandes cambios, como nuevas tecnologías, nuevas interfaces con el entorno del sistema o plataformas avanzadas.

Además, los planificadores deberían examinar si el negocio necesitaría ser adaptado a otras organizaciones, como filiales o socios resultantes de fusiones. Sin embargo, no se debe llegar al absurdo especulando con el futuro. Uno no quiere construir una arquitectura rígida para el sistema si es posible prever que ésta cambiará en el futuro, pero tampoco queremos introducir una flexibilidad innecesaria en el sistema, es decir, una flexibilidad que no será utilizada nunca.

### ***Planear los criterios de evaluación***

A grandes rasgos, los criterios de evaluación pueden clasificarse en dos categorías:

- Requisitos verificables
- Criterios más generales

#### **▶ Requisitos verificables**

Cada una de las construcciones creadas en una iteración es sometida a prueba. Los ingenieros de pruebas aumentan y refinan las pruebas que se ejecutan sobre cada construcción, acumulándose así una cantidad de pruebas que serán utilizadas como pruebas de regresión en etapas posteriores.

Las primeras iteraciones introducen más funciones nuevas y más pruebas que las últimas iteraciones. Conforme avanza la integración de construcciones disminuye el número de nuevas pruebas y se ejecuta un número creciente de pruebas de regresión para validar la implementación del sistema realizada hasta ese momento. Por tanto, las primeras construcciones e iteraciones requieren más planificación y diseño de pruebas, mientras que en las últimas se realiza más ejecución y evaluación de pruebas.

---

### ► **Criterios más generales.**

---

Los criterios generales no se pueden reducir a caminos dentro del código que los ingenieros de pruebas pueden probar; puede que éstos hayan de ser percibidos en prototipos primero y en las series de construcciones en funcionamiento y de iteraciones más tarde. Los usuarios, clientes y desarrolladores pueden ver pantallas e interfaces gráficas de usuario con mayor perspicacia de la que pueden emplear en estudiar la información estática contenida en los artefactos de los modelos.

Los criterios de evaluación dicen cómo verificar que los requisitos para una iteración han sido desarrollados correctamente. Especifican en términos que pueden ser observados o verificados hasta dónde intenta el jefe del proyecto que llegue la iteración. Al principio los criterios son un tanto vagos, pero van siendo más detallados conforme los casos de uso, escenarios de los casos de uso, requisitos de rendimiento y casos de prueba, concretan cómo han de ser los incrementos sucesivos.

---

## **Evaluar las iteraciones y las fases**

---

Si se quieren conocer completamente los beneficios de la forma de trabajo iterativo, el proyecto debe evaluar la parte del trabajo que se ha completado al final de cada iteración o fase. El jefe de proyecto es responsable de esta evaluación.

Este análisis se hace no sólo para evaluar la iteración en sí, sino para impulsar estos otros dos objetivos:

- El primer objetivo de una evaluación es examinar lo que ha sido realizado en términos del criterio de evaluación actual.
- El segundo objetivo es el de comparar el progreso realizado con el plan de la iteración o del proyecto.

¿Avanza el proyecto dentro del presupuesto y siguiendo la planificación?

¿Está alcanzando los requisitos de calidad, de acuerdo con los resultados de las pruebas o la observación de los prototipos, componentes y construcciones?

Idealmente, el proyecto cumplirá los criterios. El jefe del proyecto distribuye los resultados de la evaluación a las personas relacionadas con el proyecto y archiva el documento correspondiente. Este documento no se actualizará, cuando se informe de la siguiente evaluación se creará un nuevo documento.



## Criterios no alcanzados

Las evaluaciones raramente van tan bien. Frecuentemente alguna iteración no alcanza los criterios satisfactoriamente. El proyecto puede tener que prolongar este trabajo durante la siguiente iteración (o hasta la iteración posterior apropiada).

Este trabajo puede implicar:

- » Modificar o extender el modelo de casos de uso
- » Modificar o extender la arquitectura
- » Modificar o extender los subsistemas desarrollados hasta entonces.
- » Buscar otros riesgos
- » Incorporar ciertas habilidades al equipo

También es posible que, simplemente, se necesite más tiempo para poder llevar a cabo el plan existente. Si éste es el caso, el proyecto podría extender la planificación de la primera iteración, aunque si esto ocurre entonces se debería especificar una fecha límite firme.

## La siguiente iteración



Un hito principal marca la terminación de una fase

El punto en el que no sólo el equipo del proyecto sino todas las personas involucradas en él, en particular las autoridades que proporcionan los fondos y los representantes de los usuarios, coinciden en que el proyecto ha alcanzado el criterio del hito y que está justificado el paso a la siguiente fase.

Sobre la base de la evaluación, el jefe de proyecto (asistido por algunas de las personas que trabajaron en la iteración o en la fase y por algunas de las personas que participarán en la siguiente fase) hace lo siguiente:

- » Determina que el trabajo está listo para pasar a la siguiente iteración.
- » Si se necesita rehacer algún trabajo asigna en cuáles de las siguientes iteraciones debería realizarse.
- » Planea en detalle la siguiente iteración.
- » Actualiza el plan, en menos detalle, de las iteraciones posteriores a la siguiente.
- » Actualiza la lista de riesgos y el plan del proyecto.
- » Compara el coste y la planificación de la iteración con los planeados.

Observamos que, en el desarrollo basado en componentes, el número de líneas de código escritas no es un indicador fiable del progreso. Puesto que un desarrollador puede

reutilizar bloques (subsistemas, clases y componentes) ya diseñados, se puede avanzar mucho.

## Evolución del conjunto de modelos

Una característica clave del desarrollo iterativo en fases es la evolución del conjunto de modelos. Esta evolución contrasta con lo que ocurre en el modelo en cascada, donde imaginamos que los requisitos se completan primero, a continuación el análisis y así sucesivamente.



En el desarrollo iterativo, los modelos crecen juntos a través de las fases. En las primeras iteraciones unos modelos van por delante de otros, por ejemplo, el modelo de casos de uso va por delante del modelo de implementación

En lugar de que un modelo evolucione casi independientemente del siguiente modelo pensamos en términos de un estado del sistema entero que evoluciona a un estado más avanzado del sistema entero. Cada iteración -quizás cada construcción dentro de una iteración- representa un avance en el estado del sistema completo, el cual se refleja en el movimiento gradual hacia la compleción del conjunto de modelos.

Considerar el grado en que esta evolución ha progresado en cada evaluación será un indicador importante para el grupo evaluador.

---

## < Nombre del proyecto > Plan de Aceptación del Producto Versión <1.0>

*[Nota: el texto adjuntó entre [ ] y desplegado en las cursivas grises es incluido para proporcionar la guía al autor y debe anularse antes de publicar el documento.]*

### Historia de la revisión

Fecha	Versión	Descripción	Autor
<día/mm/año>	<x.x>	<detalles>	<nombre>

## Tabla de Contenidos

### I. Introducción

- I.1 Propósito
- I.2 Alcance
- I.3 Definiciones, Siglas y Abreviaturas

- 1.4 Referencias
- 1.5 Apreciación global
- 2. Responsabilidades**
- 3. Tareas de Aceptación de producto**
  - 3.1 Criterio de Aceptación de producto
  - 3.2 Auditoría de la Configuración física
  - 3.3 Auditoría de la Configuración funcional
  - 3.4 Horario
- 4. Requisitos de recursos**
  - 4.1 Requisitos del hardware
  - 4.2 Requisitos del software
  - 4.3 Requisitos de la documentación
  - 4.4 Requisitos del personal
  - 4.5 Pruebe los Requisitos del Datos
  - 4.6 Otros Requisitos
- 5. Resolución del problema y Acción correctiva**
- 6. Ambiente de Aceptación de producto**
- 7. Identificación de Evaluaciones del Artefacto Requeridas**
- 8. Herramientas, Técnicas y Metodologías**

# Plan de Aceptación del Producto

## Introducción

*[La introducción del Plan de Aceptación de Producto debe proporcionar una apreciación global del documento entero. Debe incluir el propósito, alcance, definiciones, siglas, abreviaturas, referencias y apreciación global de este Plan de Aceptación de Producto.]*

### Propósito

*[Especifique el propósito de este Plan de Aceptación de Producto.]*

### Alcance

*[Una descripción breve del alcance de este Plan de Aceptación de Producto; con los que el Proyecto está asociado, y lo que es afectado o influenciado por este documento.]*

### Definiciones, Siglas y Abreviaturas

*[Esta subdivisión debe proporcionar las definiciones de todas las condiciones, las siglas, y abreviaturas exigidas para interpretar la Aceptación del Producto propiamente dicho. Esta información puede proporcionarse por referencia al Glosario del proyecto.]*

### Referencias

*[Esta subdivisión debe proporcionar una lista completa de todos los documentos referenciados en otra parte en el Plan de Aceptación de Producto. Cada documento debe identificarse por el título, número del informe (si es aplicable), fecha, y publicando la organización. Especifique las fuentes y referencias. Esta información puede proporcionarse por la referencia a un apéndice o a otro documento.]*

### Apreciación global

*[Esta subdivisión debe describir lo que el resto del Plan de Aceptación de Producto contiene y explica cómo el documento está organizado.]*

## Responsabilidades

*[Identifique las responsabilidades de ambos el cliente explícitamente, y el desarrollo unce, preparando para y realizando las actividades de aceptación de producto.]*

## Tareas de Aceptación de producto

### **Criterio de Aceptación de producto**

*[Identifique aquí el criterio objetivo por determinar aceptabilidad de los artefactos entregables de este proyecto. Éstos el criterio debe estarse de acuerdo formalmente por el cliente y el desarrollo unce.]*

### **Auditoría de la Configuración física**

*[Identifique y liste aquí los artefactos a que serán entregados, y aceptó por, el cliente que es el resultado del trabajo en este proyecto.]*

### **Auditoría de la Configuración funcional**

*[Para cada artefacto identificado en la Configuración Física, identifique el método de la evaluación y el nivel de detalle que se usará para determinar si se encuentra el criterio de aceptación de producto.]*

*Los métodos pueden incluir software ejecución probando, la demostración del producto, la documentación repasa etc.]*

### **Horario**

*[Un horario que indica la salida y tiempos del fin por cada uno de la aceptación del producto atarea, incluso la preparación y actividades del arreglo.]*

## Requisitos de recursos

### **Requisitos del hardware**

*[por ejemplo hardware ítems, equipamiento de interface, ítems de firmware]*

### **Requisitos del software**

*[por ejemplo sistemas operativos, compiladores, drivers de testeo, generadores de tests de datos]*

### **Requisitos de la documentación**

*[por ejemplo documentación de testing, referencias técnicas]*

### **Requisitos del personal**

*[por ejemplo miembros del equipo de desarrollo, representantes del cliente, terceras partes con autoridad]*

### **Pruebe los Requisitos del Datos**

*[por ejemplo tamaño, tipo y composición de datos para soportar los tests de aceptación]*

### **Otros Requisitos**

*[por ejemplo equipamiento especial]*

## Resolución del problema y Acción correctiva

*[Esta sección describe los procedimientos por informar para ocuparse de problemas que se identificaron durante las actividades de Aceptación de Producto. Normalmente esto se dirige adjuntando el Plan de Resolución de Problemas por la referencia.]*

## Ambiente de Aceptación de producto

*[Describe los planes para preparar el ambiente de aceptación de producto.]*

## Identificación de Evaluaciones del Artefacto Requeridas

*[Basado en la descripción de Auditoría de Configuración Funcional, identifique cada evaluación del artefacto individual externo que se llevará. Para cada uno, liste el tipo de evaluación (la prueba, la revisión), y su objetivo]*

*La nota: UN artefacto de la Causa instrumental se preparará para cada prueba identificada aquí.]*

## Herramientas, Técnicas y Metodologías

*[Una lista de cualquier herramienta específica, técnicas y metodologías que serán usadas al realizar las actividades de Aceptación de Producto.]*

## Modelo de examen

- ▶ Lea con detenimiento y analice el siguiente enunciado, antes de comenzar a desarrollar su examen.
- ▶ Justifique todas sus posiciones.

### PARTE PRÁCTICA

El Contador de una mediana empresa es responsable de preparar un presupuesto anticipado y la proyección de las ventas, costos de mano de obra, costos de los suministros y gastos de operación. Los presupuestos se preparan tres semanas antes de la semana a la cual pertenecen y se envían a gerente de cada departamento, quien, a su vez, programa a los empleados para cubrir las ventas esperadas y permanecer de esta forma dentro del presupuesto de mano de obra y de guías para los gastos.

Primero se prepara el pronóstico de ventas; después se determinan los costos de mano de obra y de materiales, utilizando un porcentaje fijo de nivel esperado de ventas. En forma similar, una parte predeterminada del presupuesto de ventas se destina a los costos extra.

Desdichadamente, el Contador comete muchos errores en el cálculo. Después de que se termina la semana a la cual pertenecía el presupuesto, el Contador corrige a mano las ventas actuales y los gastos, además de los montos del presupuesto. De vez en cuando, se preparan informes especiales para acumular los pronósticos y las ventas actuales, así como los gastos sobre períodos de 13, 26 y 52 semanas.

Todo el proceso se realiza en forma manual con la ayuda de una calculadora, incluso los resúmenes anuales se preparan a mano.

El Contador ha solicitado la compra de una PC, y el desarrollo de un software especial para llevar a cabo los cálculos de presupuesto y de análisis, y almacenar las ventas presupuestadas y actuales, así como la información de los gastos de la semana. El sistema también debe tener la capacidad para utilizarlo como terminal conectada al sistema central de la compañía, en donde se almacenan los registros de las cuentas. Esa capacidad mejorará el acceso del contador a la información que actualmente pierde alguna utilidad debido a que llega tarde.

1. Enumere las posibles oportunidades o alcances que tendría este proyecto de sistemas. ¿Cómo armaría la justificación de su propuesta de sistemas para este ejemplo, qué incluiría?
2. ¿Qué informes programados regularmente deben ser proporcionados a los gerentes? Especifique.

### **PARTE TEÓRICA**

1. ¿Por qué es necesario contar con distintos métodos o medios de conversión para implementar un sistema?
2. ¿Qué resultado se obtiene (o para que sirve) realizar la prueba de caja negra sobre un sistema?
3. ¿Qué problemas o aspectos deben ser tratados o cubiertos durante la etapa de entrenamiento del usuario?
4. ¿Por qué es importante evaluar la implementación de un sistema?

# Índice

<b>Fundamentos de la materia .....</b>	<b>1</b>
Organización del Módulo .....	1
<b>Objetivos Generales.....</b>	<b>2</b>
<b>Contenidos.....</b>	<b>2</b>
<b>Mapa Conceptual.....</b>	<b>4</b>
<b>Propuesta Metodológica.....</b>	<b>5</b>
<b>Formas de evaluación.....</b>	<b>6</b>
<b>Material soporte de información de todas las unidades.....</b>	<b>6</b>
<b>Evaluación Diagnóstica .....</b>	<b>7</b>
<b>Iconos de Referencia.....</b>	<b>7</b>
 <b>UNIDAD I</b>	
<b>“Introducción al Análisis” .....</b>	<b>9</b>
Objetivos .....	9
Temas de la Unidad.....	9
Mapa Conceptual .....	10
Introducción .....	11
RECORDAMOS: Ciclo de Vida del Proceso Desarrollo Unificado .....	11
Fases dentro de un ciclo .....	12
Iteraciones.....	14
Ciclos de desarrollo.....	14
Flujos de trabajo del proceso .....	14
Modelos.....	15
RECORDAMOS: UML .....	16
Qué es el análisis? .....	17
Esencia del Análisis .....	17
¿Por qué hay que hacer Análisis? .....	18
¿Qué significa hacer Análisis con objetos? .....	18
Principales actividades del Proceso de Análisis .....	19
I Encontrar los objetos .....	19
II Organizar los objetos .....	19
III Especificar las operaciones de los objetos .....	20
IV Definir la estructura interna de los objetos .....	20
Reusabilidad en el análisis .....	21
No debemos Confundir Análisis con Diseño .....	21
El análisis en el Proceso de Desarrollo Unificado de Software, en pocas palabras...21	
Descripción del análisis en términos estáticos .....	22
11 Clases Análisis .....	22
12 Paquete del análisis.....	25
Descripción del análisis en término Dinámicos.....	27
Análisis de un caso de uso – .....	27
Identificar clases de análisis.....	27
El papel del análisis en el ciclo de vida del software .....	29
Autoevaluación .....	30

## UNIDAD II

<b>Introducción al Diseño .....</b>	<b>31</b>
Objetivos .....	31
Temas de la Unidad .....	31
¿Qué hacemos en el diseño? .....	32
Diseño en el ciclo de vida (vista estática) .....	33
El papel del “Diseño” en el ciclo de vida del Proceso Unificado .....	33
Modelo de Diseño .....	34
Clases de diseño .....	34
Realización de un caso de Uso del diseño .....	35
Diagramas de clases .....	36
Subsistema de Diseño .....	38
Interfaz .....	38
Modelo de despliegue .....	39
<b>Diseño en el ciclo de vida (vista dinámica) .....</b>	<b>41</b>
Diseño de la arquitectura .....	41
a- Identificación de clases del diseño .....	41
b- Identificación de nodos y configuraciones de red .....	46
c- Identificación de subsistemas y sus interfaces .....	46
<b>Comparación entre el modelo de Análisis y el modelo de Diseño .....</b>	<b>48</b>
<b>Autoevaluación .....</b>	<b>49</b>

## UNIDAD III

<b>Cómo construimos los modelos .....</b>	<b>51</b>
Objetivos .....	51
Contenidos de la Unidad .....	51
Mapa Conceptual .....	52
Introducción .....	53
Caso de Estudio .....	53
Premisas del caso .....	53
Enunciado y Resolución de un caso .....	53
<b>Propuesta .....</b>	<b>57</b>
<b>1. Modelado del Sistema de Negocio .....</b>	<b>57</b>
1.1. Diagrama de Casos de Uso del Sistema de Negocios .....	57
1.2. Descripción de casos de uso del Sistema de Negocios .....	59
1.3. Descripción de objetos activos trabajadores .....	74
<b>2. Modelado del Sistema de Información .....</b>	<b>74</b>
2.1 Diagrama de Casos de Uso .....	74
2.2 Descripción de cada Caso de Uso y Prototipos de Interfaz .....	76
2.3. Modelo de Objetos del Dominio .....	93
<b>3. Flujo de trabajo del Análisis .....</b>	<b>95</b>
3.1 Diagramas de Colaboración .....	95
3.2. Subsistemas .....	98
1. Subsistema Archivos .....	98
2. Subsistema Horarios de Viaje .....	98
3. Subsistema Reserva y Ventas de pasaje .....	98
4. Subsistema Encomiendas .....	98
5. Subsistema Reparación de Unidades .....	98
6. Subsistema Control de Ingresos y Egresos .....	99
<b>4. Flujo de trabajo de Diseño .....</b>	<b>99</b>
4.1. Ambiente de implementación .....	99



4.2 Diagrama de Despliegue .....	100
4.2 Diagrama de Clases .....	101
<b>4.4. Diagramas de Estado.....</b>	<b>103</b>
<b>4.4. Diagramas de Estado.....</b>	<b>103</b>
Objeto Unidad.....	103
Objeto Pasaje.....	104
Objeto Encomienda .....	105
Objeto Factura.....	106
<b>Documentación para desarrollo o construcción .....</b>	<b>107</b>
Desarrollo .....	107
Capacitación de Programadores .....	107
Especificación de las necesidades de Formularios .....	107
Desarrollo de los Programas.....	108
<b>Diseño de la migración y carga inicial de datos .....</b>	<b>109</b>
Especificación del Entorno de Migración .....	110
Diseño de Procedimientos de Migración y Carga Inicial.....	110
Diseño Detallado de Componentes de Migración y Carga Inicial .....	111
Productos .....	111
Participantes .....	111
Autoevaluación.....	112
 <b>UNIDAD IV</b>	
<b>Implementación.....</b>	<b>115</b>
Objetivos .....	115
Contenidos de la Unidad.....	115
Mapa conceptual .....	116
Introducción a la Implementación .....	117
Definición y Establecimiento del Plan de implementación .....	118
(a) Definición del Plan de Implantación .....	119
Plan, contiene.....	119
Operación del nuevo sistema .....	119
b) Especificación del equipo de Implantación.....	121
<b>Formación o capacitación para la implementación .....</b>	<b>122</b>
Objetivos de la Capacitación o Adiestramiento .....	122
Consideraciones al proceso de capacitación.....	122
Esquema .....	124
<b>Incorporación del Sistema a entorno de operación.....</b>	<b>125</b>
Preparación de la instalación .....	125
Realización de la Instalación.....	125
Carga de datos al entorno de operación .....	126
<b>Pruebas de implementación del sistema .....</b>	<b>126</b>
<b>Pruebas de aceptación del sistema .....</b>	<b>127</b>
<b>Evaluación y ajuste .....</b>	<b>128</b>
<b>Preparación del Mantenimiento del Sistema.....</b>	<b>128</b>
<b>Presentación y aprobación del sistema .....</b>	<b>129</b>
Informe Final .....	129
Modelo de Informe Final .....	130
Aprobación.....	130
Operación .....	130
Entrega.....	130
Recepción.....	131

Paso a entorno de producción .....	131
Documentación entregable .....	131
<b>Elaboración de Manuales de Usuario .....</b>	<b>132</b>
Manuales de Sistemas .....	132
Proceso de elaboración de un manual de sistema .....	133
Proceso de aprobación, emisión y control del manual .....	133
Esquema del contenido de un manual .....	134
¿Que incluir en un manual? .....	134
<b>Evaluación de la Implementación .....</b>	<b>135</b>
Mantenimiento .....	136
Autoevaluación .....	136

## UNIDAD V

<b>Prueba .....</b>	<b>137</b>
Objetivos .....	137
Contenidos de la Unidad .....	137
Mapa Conceptual .....	138
Introducción a la Prueba .....	139
La Prueba .....	139
Propósito de la prueba .....	139
Tipos de tests .....	140
Niveles de Prueba .....	141
Estrategias de Prueba .....	144
El Proceso de Prueba .....	145
Planificación de la prueba .....	145
Identificación de la Prueba .....	146
Especificación de la prueba .....	146
Ejecución de las Pruebas .....	147
Autoevaluación .....	147

## ANEXO A

<b>Integramos los Flujos de trabajo del PDUS .....</b>	<b>149</b>
Objetivos .....	149
Contenido de la Unidad .....	149
Mapa Conceptual .....	150
Introducción .....	151
El flujo de trabajo de iteración genérico .....	151
La necesidad de equilibrio .....	151
Las fases son la primera división del trabajo .....	152
“El planificar” precede “al hacer” .....	155
Planear las cuatro fases .....	155
Plan de iteraciones .....	156
Pensar a largo plazo .....	157
Planear los criterios de evaluación .....	157
Evaluar las iteraciones y las fases .....	158
Criterios no alcanzados .....	159
La siguiente iteración .....	159
Evolución del conjunto de modelos .....	160
<b>Modelo de examen .....</b>	<b>163</b>

---

**Área Informática**

*guía de actividades*

---

# **Sistemas IV**

---

Material elaborado por  
**Mgtr. Ing. Cecilia Savi**

---

**INSTITUCION CERVANTES**





## Guía de Actividades

### Actividad 1

Identifique, cuáles serían las actividades, artefactos, modelos o diagramas a considerar cuando realizamos una visión estática del Diseño; Y cuáles, cuando realizamos una visión Dinámica del modelo de Diseño.

¿Hay diferencias?. ¿En que se basa la diferencia?

t1 

### Actividad 2

Desarrolle el modelo de Despliegue completo, (gráfica y ambiente Implementación) en la siguiente situación.

*El Centro de Educación y Nutrición, se encuentra ubicado en la calle 9 de Julio, en el centro de la ciudad. Nace luego de que dos Licenciadas en Nutrición, realizaran un estudio de mercado sobre distintos planes de salud existentes en el área de nutrición que se ofrecen en la ciudad de Córdoba. Este centro se dedica, a brindar novedosos planes de salud, entre los cuales se encuentra el programa de control de crecimiento del niño “Crecer”. El objetivo del plan crecer, es vigilar el normal crecimiento del niño, a fin de detectar tempranamente posibles desviaciones, para corregirlas a tiempo. El programa no intenta sustituir el control pediátrico periódico, simplemente lo complementa.*

*La organización, intenta brindar un servicio educativo y asistencial ambulatorio privado en el área de nutrición, con carácter especializado y diferente a todo lo que ya existe en el mercado.*

*Como estrategias, se han pensado y creado algunos paquetes de servicios, con bajo costo, que son ofrecidas en las guarderías, bajo el nombre de club crecer, la implementación de un Web-site, en las que se pudiesen realizar consultas, por medio de e-mails o por Chat con especialistas, tests de preguntas indicativas, sobre la conveniencia de visitar el centro, etc.*

t2 

## Actividad 3

---

*GE. es un distribuidor de productos eléctricos de todo tipo, que van desde lámparas para casas hasta lámparas para sistemas de iluminación industriales. La compañía también almacena diversos artículos eléctricos como timbres para puertas, unidades de calefacción eléctrica y una amplia variedad de conectores electrónicos.*

*Durante los pasados cuatro años, el negocio ha aumentado más del doble y las utilidades se han incrementado en mas de 115%.*

*Las transacciones de los clientes son una combinación de compras de menudeo en efectivo y a crédito, compras de mayoreo a los contratistas de productos y muchas compras de las empresas que venden a las industrias y a los comercios.*

*Control de inventario informa semanalmente los productos ingresados como nuevos y los precios a los que deben ser vendidos los diferentes productos.*

*Las ventas al menudeo en efectivo requieren, que las notas de venta tengan tres copias (una para el cliente, otra para el departamento de contabilidad y una más para el control de inventario).*

*Las ventas a crédito son a través del Bco. C&O o de la tarjeta MCV. Para estas se llena una forma de crédito bancario estándar junto con el recibo de ventas en efectivo.*

*Las compras de los contratistas, en productos eléctricos y de los clientes de las industrias y los comercios requieren una forma de orden de compra por parte del cliente, la cual va al archivo de la oficina.*

*Por otra parte, se escribe la factura que incluye la relación de la mercancía y, desde luego, el nombre y la dirección del cliente. Una copia se envía por correo al cliente, una segunda copia se incluye con la orden como copia de remisión, y las otras dos copias se envían al control del inventario y del Departamento de Contabilidad.*

*Mensualmente es necesario informar al Dpto. de Contabilidad sobre los totales vendidos por mayoristas y minoristas, según las diferentes formas de ventas.*

Aplique alguna herramienta que Ud. considere significativa para analizar y diseñar una propuesta en base a la situación planteada.

En el caso resuelto de la unidad III, Sistemas IV se han utilizado algunos modelos y diagramas, que son los más comúnmente aplicados. Pero no son los únicos, existen otros que tienen características muy útiles y que pueden ser aplicados al momento de desarrollar un sistema de Información. Seleccione a su criterios los artefactos que utilizará.

**t3** 

## Actividad 4

**a-** Realizar una lectura comprensiva del capítulo IV Implementación, del módulo de estudio y responder el cuestionario que se presenta a continuación:

1. ¿Hay dos aspectos relevantes a tener en cuenta en el establecimiento de un plan de implementación: Definición del Plan y Equipo de implantación. Establezca a su criterio cual es el objetivo de cada aspecto y seleccione 3 ítems que contemple cada uno.
2. ¿Quiénes deben ser capacitados al momento de implementar un sistema?. Construya un cuadro comparativo con el objetivo de cada tipo de capacitación y los temas que involucra cada uno.
3. ¿Qué problemas o aspectos deben ser tratados o cubiertos durante la etapa de entrenamiento del usuario?
4. ¿Qué diferencia hay entre Prueba de Implementación y Prueba de aceptación? ¿Es lo mismo que Plan de implementación y Plan de aceptación?
5. Enumere los pasos que determinan la aceptación de un sistema
6. ¿Qué tipos de manuales son necesarios?. ¿Qué objetivos cumple cada uno?. ¿Cuál es su contenido?
7. ¿Por qué es necesario contar con distintos tipos de medios de conversión para implementar un sistema?
8. ¿Por qué es importante evaluar la implementación de un sistema?

**b-** Aplique los contenidos trabajados en el punto (a), en una situación práctica

1. Basándose en el enunciado, de un ejemplo de cómo aplicaría a este caso, una estrategia de implementación
2. Teniendo en cuenta la documentación que acompaña al sistema, elabore una muestra de cada tipo de manual.

La imprenta TODO ESCRITO se dedica a la impresión y comercialización de volantes de todo tipo, así como de tarjetas personales.

La misma cuenta con el Administración dividida en: Recepción de Pedidos y Cobros, Producción propiamente dicha, y Expedición (servicios a domicilio).

El proceso de venta se realiza de la siguiente manera: el encargado de recepción recibe los pedidos de impresión de diversos clientes, dicho encargado registra los mismos en un cuaderno (en caso de ser telefónico sólo se aceptan pedidos únicamente de clientes con cuenta corriente), y los clientes deben señalar los mismos entregando un 50% del monto total, ante esto el encargado de recepción entrega dos Ticket de Señas y el cliente se dirige a Cobros para pagar la señal, y el encargado de cobros sella un Ticket con la leyenda 'Entrega Señal' y el otro lo archiva. El encargado de recepción corrobora el pago de la señal y comunica luego el pedido realizado al área producción.

Ante cada pedido cumplido, el encargado de producción entrega los ítems terminados junto con un remito interno a Cobros. El remito detalla el pedido realizado, en cuanto a tipo de impresión y cantidad (no especifica el importe total pactado, ni fecha de entrega)

Cuando los clientes retiran sus pedidos, el encargado de cobros busca el remito, solicita el Ticket de Señas, controla las cantidades detalladas en el mismo con las realizadas y si esta todo OK cobra al cliente el 50% restante (no controla con el pedido anotado en el cuaderno a no ser que el cliente presente alguna queja) y genera la correspondiente factura. En caso de ser un cliente con C/C se envía el pedido al encargado de expedición para que lo entregue al domicilio del cliente junto con el remito, previo registro del monto en la cuenta del mismo.

Últimamente se han presentado inconvenientes en la entrega de los pedidos, dado que las cantidades solicitadas no son las entregadas, demoras en las fechas de entregas (esto en caso de clientes con C/C) y pérdidas económicas para la imprenta, dado que ante un determinado pedido a veces se realiza más cantidad de la solicitada y el cliente no siempre abona los ítems de más, por lo que se deben rezagar los ítems sobrantes. Otro inconveniente es que ante un determinado pedido de tarjetas personales, el tamaño de letra, formato de la tarjeta y color no siempre se respetan a gusto del cliente, y por esto se debe otorgar un descuento el mismo, asumiendo de ésta manera los errores en la recepción de pedidos o en la comunicación a producción.

Los clientes con C/C se dirigen personalmente a abonar las cuotas (entregas de pago), el encargado realiza el cobro y le emite una factura por dicho monto que no siempre se debita de su cuenta, generando inconvenientes en los próximos cobros.

---

## Actividad 5

---

**a-** Realizar una lectura comprensiva del capítulo V Prueba, del módulo de estudio y responder el cuestionario que se presenta a continuación:

1. ¿Qué incumbencias tiene la verificación en la prueba de un proyecto?
2. ¿Porque decimos que el propósito de la prueba es encontrar fallas?
3. Armar un cuadro sinóptico con los diferentes tipos de prueba y resume en una frase los objetivos de cada prueba.
4. ¿Se puede realizar una prueba de integración, previa a una prueba de unidad?
5. Fundamente
6. ¿Cuáles son los pasos correctos y completos en un proceso de prueba y cuáles las características mas importantes de cada uno?
7. Identifique las estrategias de prueba, sus características
8. Existe alguna documentación que respalde el proceso de prueba, ¿Qué aspectos deben ser especificados?

**b-** Aplico los contenidos trabajados en el punto (a), en la misma situación practica de la actividad N°

1. Basándose en el enunciado del caso, dé un ejemplo de cómo aplicaría los diferentes tipos de test.
2. ¿Cómo realizarías una prueba de caja negra y la prueba basada en estados?.
3. Elabore un informe con las actividades de las prueba realizadas y sus resultados



## Tutores

t1 ▶

### Visión Estática y Visión Dinámica

En el desarrollo de todos los contenidos de los módulos de Sistemas III y IV, con respecto al Proceso de Desarrollo Unificado del Software, siempre se ha trabajado con ambas visiones (Estatica y Dinámica) como complementos indispensables al momento de desarrollar un Sistema de Información. La recomendación, es la lectura de la Unidad I de este módulo.

t2 ▶

### Mucha Imaginación

Para realizar el diagrama de despliegue y ambiente implementación, sin desarrollar los otros modelos anteriores, deberá poner un poco de imaginación y de predicción. Es decir actuar casi, como un profesional en actividad. ¿Por qué? Esto es porque deberá diseñar la ubicación de los equipos, de los procesos, de los software de base, y otros tantos puntos más.

t3 ▶

### Interbank – Complejo Cine Club

En base a los casos de estudios expuestos en el módulo completamente desarrollado, lea las situaciones planteadas y observe la secuencia de pasos, que faltaría completar y que sea conveniente, para desarrollar cada uno de los modelos básicos de la construcción de un sistema automatizado.

#### M. de Negocios

Diagrama de Casos de Uso del Mod. Negocios

Descripción de Caso de uso de M. Negocio

#### M. de Dominio

Diagrama de clases

#### M. de Sistemas de Información

Diagrama de Casos de Uso del Mod. Información

Descripción de Caso de uso de M. Información

Diagrama de colaboración del Análisis

División de paquetes

Modelo de diseño:

Diagrama de clases

Revisión de Subsistemas

Interfaz

Modelo de despliegue

Lay out de equipos en la Organización

Ambiente Implementación

## Trabajo Práctico Integrador IV

---

### Objetivos:

- ▶ Evaluar el grado de aprendizaje sobre los conceptos aprendidos en las asignaturas Sistemas IV, Programación IV y Base de Datos II.
- ▶ Evaluar el progreso de aprendizaje respecto de las herramientas y del vocabulario técnico adquirido, valorar terminología técnica usada.
- ▶ Evaluar la transferencia de lo aprendido a situaciones prácticas.
- ▶ Lograr la visión sistémica de un proyecto informático.

### Enunciado

En BASE al documento trabajado en el cuatrimestre anterior, trabajo practico integrador del tercer cuatrimestre, en el que se definió: Modelo de negocio, modelo de información, diseño de base de datos lógica y construcción de prototipos de Interfaz. Complementar con las herramientas **que responden al ciclo de vida de un proceso de desarrollo unificado (PDUS)**:

Revisión del Sistema de Información:

- 1- Reformular el Modelo del Sistema de Información y realizar las plantillas en base a las observaciones del nuevo modelo y a las correcciones del cuatrimestre anterior.

### Análisis:

Construir

- 2- Diagrama de Colaboración de 4 (cuatro) de los Caso de uso del Sistema de Información del práctico de Sistemas III, que incluya procesos de alta, baja y modificación, una consulta elaborada, un proceso de cálculo y una salida de documento o informe.  
Recordar: que se debe acompañar cada Diagrama de Colaboración con su correspondiente Caso de uso e Interfaz para ver su correspondencia.
- 3- Especificar como agruparía los Caso de uso que se vienen trabajando en un diseño de menú principal, agregando a estos, casos de uso de seguridad, respaldo, ayuda, informes de resumen.
- 4- Complementando la información de este flujo de trabajo Análisis, se debe agregar el Modelo de Datos para poder ver las clases de entidad.

### Diseño:

Construir

- 5- Especificar el Ambiente de Implementación del sistema completo.
- 6- Construir el modelo Conceptual lógico y la implementación física completa de la base de datos a implementar. Rediseñar los modelos con las entidades que aparezcan y se agreguen, a partir del escenario completo del sistema.

- 7-** La implementación física deberá realizarse sobre una arquitectura Cliente/Servidor.
- 8-** Definir políticas de seguridad de la Base de datos: plan de respaldo y recuperación, planificar niveles de seguridad por usuarios, tipo de seguridad en las cuentas (por ej. longitud y duración de las claves de acceso y las cuentas).
- 9-** Construcción y programación de las interfaces que surjan del nuevo modelo de datos. Se recomienda realizar el desarrollo de las aplicaciones aplicando al máximo la parametrización de valores externos a las aplicaciones.
- 10-** Construir reportes y consultas que nos permitan consultar o imprimir los datos solicitados. Se solicitarán consultas y reportes delimitados por parámetros.
- 11-** Construir el Diagrama de Despliegue completo.
- 12-** Realizar el Diagrama de Transición de Estados de dos clases.

## Prueba

Construir

- 13-** Desarrollar los diferentes tipos de tests de prueba de acuerdo al sistema.

## Implementación

Construir

- 14-** Especificar la implementación del sistema completo.
- 15-** Realizar el Manual del Usuario para la porción del sistema presentado.

## Forma de Presentación y Evaluación

La documentación lograda deberá ser única, un solo ejemplar. Generando, si es necesario y solicitado por los docentes, una copia para cada uno de las asignaturas involucradas.

Cada docente evaluará, el trabajo en forma integral y particular, específicamente sobre las consignas relacionadas a los contenidos de su asignatura.

El trabajo tendrá un seguimiento con fechas de control, de acuerdo a las pautas que el docente proponga en cada asignatura

Una vez revisado, el alumno realizará una presentación de todo el trabajo frente al resto de los compañeros, en una exposición oral y grupal.

