

ASIGNATURA: INGENIERÍA DEL SOFTWARE (2º CURSO)

CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

CÓDIGO CARRERA:

Plan de estudios en extinción: 40=SISTEMAS y 41=GESTIÓN

Plan de estudios NUEVO: 53=SISTEMAS y 54=GESTIÓN

MATERIAL PERMITIDO: NINGUNO

MODELO: NACIONAL 1ª SEMANA



Departamento de Ingeniería de  
Software y Sistemas Informáticos

---

**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. ¿Qué dimensión del proceso de desarrollo de software añade el modelo en V al modelo en cascada? ¿Qué implica respecto a la comprobación de la corrección en las distintas fases del ciclo de vida?

### Solución

En el modelo en V se contempla **el nivel de detalle** sobre el que se trabaja en cada una de las fases. El SRD, documento producido en la fase de análisis, contempla el **sistema** en su totalidad. Tras la fase de diseño, en la que se lleva a cabo una descomposición del sistema para abordar su codificación, se desciende al nivel de **módulo**. Tras la codificación de todos los módulos se realiza su integración, ascendiendo de nuevo al nivel de sistema completo.

El modelo en V pone de manifiesto que la salida de una fase del ciclo de vida no sólo afecta a la siguiente. Existe una relación entre salidas y entradas de fases no consecutivas en el tiempo pero situadas en el mismo nivel de detalle. Este hecho se observa a la hora de realizar comprobaciones de la corrección. La **validación** se hace tras la fase de integración, donde nos encontramos en el nivel del sistema completo, comprobándose que se cumple lo estipulado en el documento producido tras la fase de análisis (SRD). La **verificación** se lleva a cabo tras la fase de codificación, realizándose una comprobación de la corrección del sistema a nivel de módulo.

2. Defina pruebas alfa, beta, de caja negra y de caja transparente. Razone si las pruebas alfa y beta se pueden considerar como de caja negra o de caja transparente.

### Solución

Las pruebas alfa y beta sirven para comprobar si un sistema completo satisface las especificaciones en un entorno real de trabajo. Mientras que las pruebas alfa se realizan en un entorno controlado donde el usuario tiene el apoyo de algún desarrollador, durante las pruebas beta no existe dicho apoyo.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

Previas a las “pruebas de sistema”, se realizan las “pruebas de unidades”, que comprueban por separado cada módulo constituyente del sistema. Dentro de las pruebas de unidades pueden distinguirse las de “caja negra”, donde se ignora por completo la estructura interna de un módulo, y las de “caja transparente”, donde se conoce y tiene en cuenta dicha estructura.

En las pruebas alfa y beta los usuarios desconocen la estructura interna del sistema, luego pueden considerarse como de caja negra.

## SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

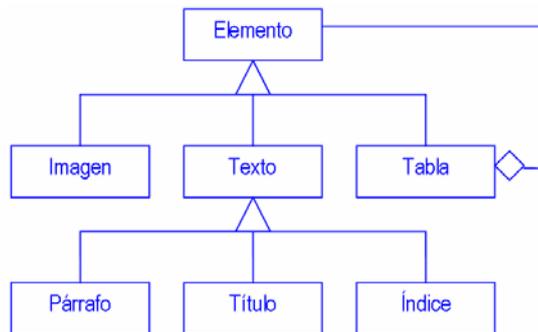
3. Se desea construir un procesador de textos capaz de manipular los siguientes elementos: imágenes, texto y tablas. En principio, sólo se contemplan tres tipos de texto: párrafos, títulos e índices. Por otro lado, una tabla puede albergar cualquier tipo de texto, imagen e incluso otras tablas.

*Utilizando un diagrama de objetos (sin incluir las operaciones ni los atributos), elabore un diseño para los elementos que maneja el procesador de textos.*

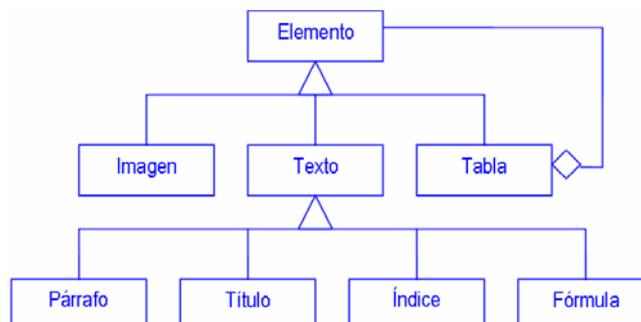
*¿Cómo afectaría a su diseño la inclusión de fórmulas matemáticas como nuevo tipo de elemento de texto?*

### Solución

Diseño inicial:



Diseño contemplando “fórmulas matemáticas”:



**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

ASIGNATURA: INGENIERÍA DEL SOFTWARE (2º CURSO)

CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

CÓDIGO CARRERA:

Plan de estudios en extinción: 40=SISTEMAS y 41=GESTIÓN

Plan de estudios NUEVO: 53=SISTEMAS y 54=GESTIÓN

MATERIAL PERMITIDO: NINGUNO

MODELO: NACIONAL 2ª SEMANA = UE ORIGINAL



Departamento de Ingeniería de  
Software y Sistemas Informáticos

---

**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. A. Describa la diferencia entre los factores de calidad de *corrección* y *fiabilidad*.

**Solución**

*Corrección*: Es el grado en que un producto software cumple con sus especificaciones. Podría estimarse como el porcentaje de requisitos que se cumplen adecuadamente.

*Fiabilidad*: Es el grado de ausencia de fallos durante la operación del producto software. Puede estimarse como el número de fallos producidos o el tiempo que permanece inutilizable durante un intervalo de operación dado.

Pág. 27 del libro

- B. Para evaluar la *corrección* de un sistema ¿qué tipo de prueba debería utilizarse, de caja negra o de caja transparente? Razone la respuesta.

**Solución**

Como se trata de evaluar si el sistema satisface sus especificaciones, lo correcto es hacer pruebas de caja negra, ya que se trataría de observar la respuesta del sistema (resultados) a determinadas entradas (casos de prueba) y comprobar que son los esperados.

Pág. 274 del libro

2. Explique brevemente los distintos niveles de comprobación de tipos que presentan los lenguajes de programación.

**Solución**

Véase pág. 255 y 256 del libro de texto.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

3. Se ha recibido una petición, por parte de un usuario final, para desarrollar un sistema automático de acceso a un garaje. En ella dice:

“...Existe un único portón que sirve de entrada y salida, pero su ancho sólo permite el paso de un vehículo. Se instalarán sensores de peso que detecten la presencia de un vehículo y semáforos tanto en el interior como en el exterior del garaje. Para controlar el acceso se utilizarán mandos a distancia codificados.”

**Analice el sistema y realice una descripción de su modelo utilizando lenguaje natural estructurado. Incluya los elementos que considere necesarios para controlar el acceso de vehículos autorizados así como para evitar daños físicos en los vehículos.**

**Realice el DFD de contexto del modelo descrito.**

**Solución**

La “descripción del modelo” es un apartado fundamental del SRD. En él se define un modelo conceptual del sistema que se va a desarrollar. Este modelo establecerá las propiedades y restricciones del sistema, dando una visión de alto nivel sin descender a detalles concretos del mismo. Es decir, indicará QUÉ debe hacer, y no CÓMO lo debe hacer. Debe ser completo, conciso, sin ambigüedades, **sin detalles de diseño o implementación, fácilmente entendible por el cliente...** Para ello se puede emplear cualquiera de las notaciones para la especificación, siendo habitual el uso de varias de ellas, de forma que se facilite su entendimiento por parte de los que van a participar en el desarrollo del sistema.

En este caso, la descripción mediante lenguaje natural estructurado se complementará con el DFD de contexto, en el que se detallan los elementos (entidades externas) que facilitan datos o reciben órdenes del sistema software.

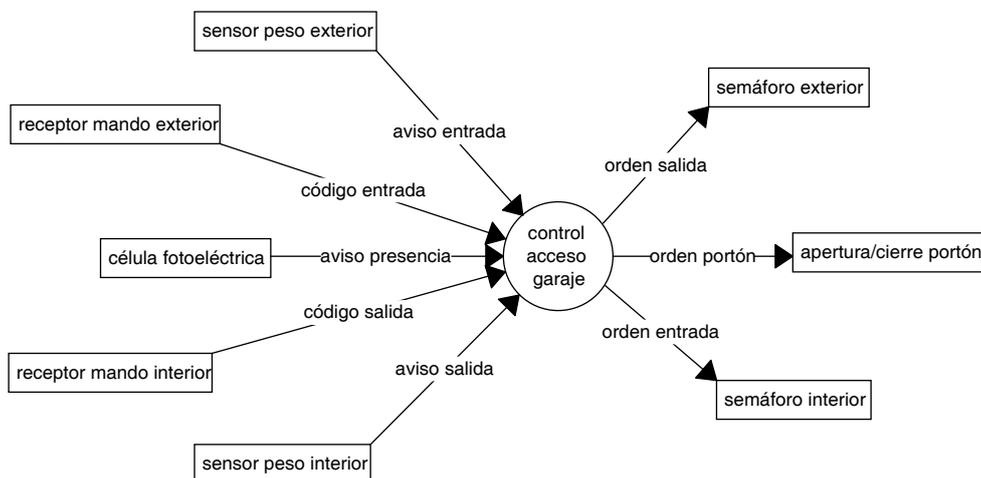
**Sistema de control de acceso a garaje. Descripción del modelo**

SI un coche autorizado quiere salir ENTONCES se pondrá en rojo el semáforo exterior, en verde el interior, y se abrirá el portón durante 30 segundos.

SI un coche autorizado quiere entrar ENTONCES se pondrá en rojo el semáforo interior, en verde el exterior, y se abrirá el portón durante 30 segundos.

SI la célula fotoeléctrica detecta un coche ENTONCES el portón permanecerá abierto durante 30 segundos.

SI un coche no autorizado quiere entrar o salir ENTONCES se pondrán en rojo los semáforos interior y exterior.



**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

ASIGNATURA: INGENIERÍA DEL SOFTWARE (2º CURSO)

CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

CÓDIGO CARRERA:

Plan de estudios en extinción: 40=SISTEMAS y 41=GESTIÓN

Plan de estudios NUEVO: 53=SISTEMAS y 54=GESTIÓN

MATERIAL PERMITIDO: NINGUNO

MODELO: EUROPA RESERVA



Departamento de Ingeniería de  
Software y Sistemas Informáticos

---

**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. ¿Qué criterios se deben emplear para utilizar más de una notación en la construcción del modelo de análisis?

### Solución

Sea cual sea la opción elegida, la notación o notaciones empleadas deberán ser fáciles de entender por el *cliente*, el usuario y, en general, por todos aquellos que puedan participar en el análisis o el desarrollo del sistema.

Si el objetivo 'global' del análisis es la comprensión del comportamiento del producto que se va a construir, la utilización de una o más notaciones para elaborar el modelo deberá decidirse de acuerdo a los criterios que debe cumplir el propio modelo:

- Completo y sin omisiones.
- Conciso y sin trivialidades.
- Sin ambigüedades.
- Sin detalles de diseño o implementación.
- Fácilmente entendible por el *cliente*.
- Separar los requisitos funcionales de los no funcionales.
- Dividido y jerarquizado.

2. Una aplicación de animación gráfica para la decoración y diseño de interiores utiliza un elemento denominado "Asiento". ¿Qué aspectos y mecanismos del diseño de software podría utilizar para la especialización de dicho elemento y su uso en un ambiente concreto (silla, butaca, banqueta, sofá, etc.)?

### Solución

La especialización se puede conseguir mediante la abstracción y a través de la herencia. En este caso, lo más inmediato parece ser el uso del mecanismo de la herencia de manera que la especialización de cada instancia en su ambiente (silla, butaca, etc.) se consigue mediante el

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

polimorfismo hereditario, es decir, adecuando cada cuál sus características específicas que han heredado de la clase “Asiento”.

SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

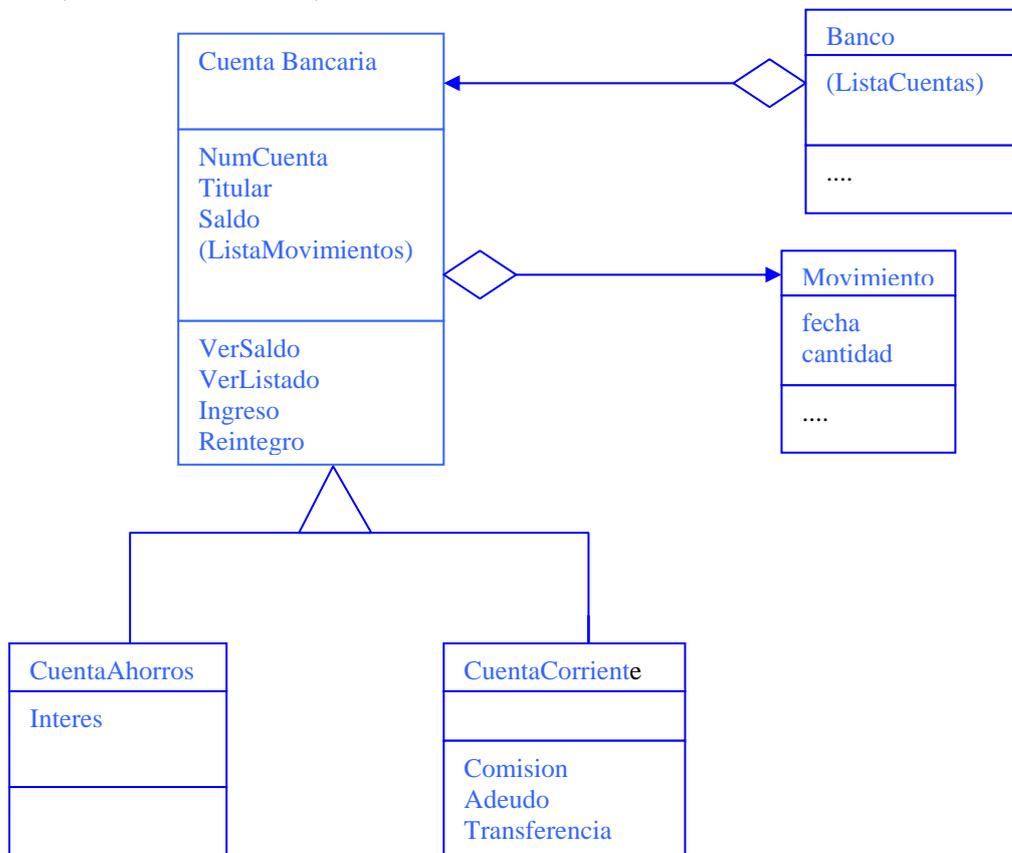
3. Un sistema informático de gestión bancaria opera con dos tipos de cuentas bancarias: la corriente y la de ahorros. Las cuentas disponen de un número que las identifica, tienen asociado un titular, una lista de los movimientos realizados hasta la fecha y almacenan el saldo disponible. Así mismo, las cuentas soportan operaciones como pedir el saldo y hacer ingresos o reintegros. Las cuentas de ahorros producen un interés que se calcula cada mes. Las cuentas corrientes no producen interés sino que reciben la carga de una comisión anual. Dichas cuentas corrientes facilitan diversas operaciones como adeudos domiciliarios y transferencias de dinero. Cada movimiento bancario consta de la fecha, la cantidad y el concepto de que se trata.

*Modele la situación anterior utilizando un diseño orientado a objetos. Utilice herencia y composición cuando se pueda.*

**Solución**

CLASES:

- Cuenta de ahorros
- Cuenta corriente
- Cuenta bancaria
- Movimientos
- Banco (Sistema de Gestión)



Entregue la hoja de lectura óptica con sus datos junto con su examen.

ASIGNATURA: INGENIERÍA DEL SOFTWARE (2º CURSO)

CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

CÓDIGO CARRERA:

Plan de estudios en extinción: 40=SISTEMAS y 41=GESTIÓN

Plan de estudios NUEVO: 53=SISTEMAS y 54=GESTIÓN

MATERIAL PERMITIDO: NINGUNO

MODELO: ORIGINAL AMÉRICA, GUINEA Y TÁNGER



Departamento de Ingeniería de  
Software y Sistemas Informáticos

---

**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. ¿Qué ventajas e inconvenientes tiene el uso de prototipos en el ciclo de vida y en qué situaciones se producen dichas ventajas e inconvenientes?

**Solución**

Véase apartado 1.5 del libro, en páginas 16 a 21.

2. ¿Se puede conseguir el polimorfismo sin utilizar la herencia? Si es así, ¿de qué manera?

**Solución**

“El concepto de genericidad (...) es una manera de lograr que un elemento genérico pueda adquirir distintas formas cuando se particulariza su utilización” (pág. 118 del libro).

Por otro lado, existe otro tipo de polimorfismo, que no está ligado a la herencia, en el que quienes adquieren múltiples formas son los operadores, funciones o procedimientos. Es el polimorfismo de **sobrecarga**.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

3. Se pretende hacer una aplicación para mejorar la gestión de convalidaciones. El procedimiento vigente es que el solicitante elabore un escrito con sus datos, la titulación de procedencia y la lista de asignaturas que quiere convalidar. Con esta información se construye un dossier en el que aparecen los datos mencionados. Según el tipo de convalidación solicitada (informativa, total, parcial, de titulación extranjera, etc.), cada dossier es revisado por el equipo docente correspondiente a cada asignatura cuya convalidación se solicita. Cada departamento imparte un conjunto de asignaturas de la titulación. Periódicamente, se reparten los dosieres entre los departamentos para la revisión de las asignaturas que les correspondan. El problema es que el listado de asignaturas no aparece ordenado ni agrupado por departamentos, lo cuál, dificulta el movimiento de las carpetas y su control.

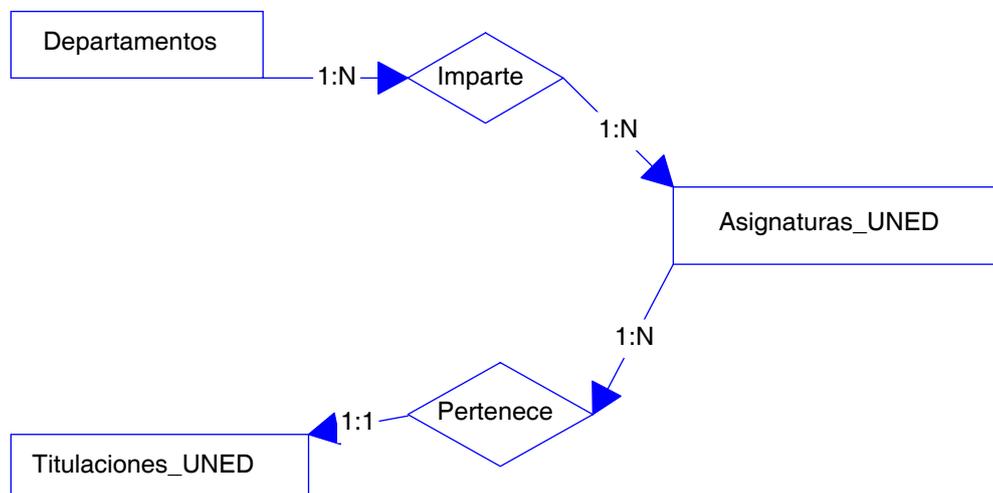
La aplicación que se pretende desarrollar permitiría a un administrativo introducir los datos del solicitante, centro de procedencia, tipo de convalidación, titulación (Sistemas, Gestión ó 2º Ciclo) y seleccionar, en una lista, las asignaturas que se solicita convalidar. El resultado debe ser un documento en el que aparezcan impresos los datos anteriores y, lo más importante, las asignaturas agrupadas por departamentos, con el nombre del departamento que imparte cada grupo.

*Realice el diseño de la base de datos con las titulaciones, departamentos y asignaturas. A continuación, construya un modelo de análisis, para la aplicación descrita, mediante DFD con dos o tres niveles.*

**Solución**

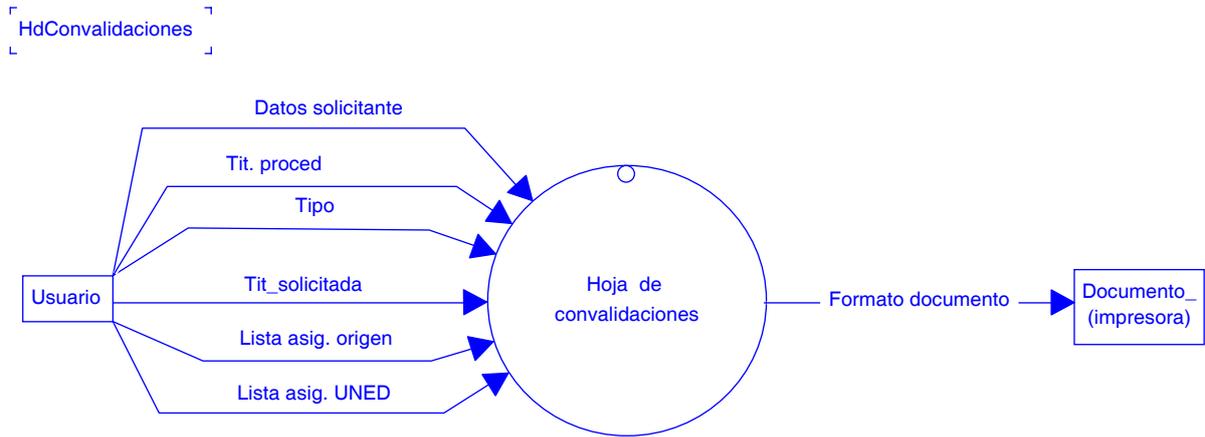
Diseño de la base de datos (E-R):

ER\_Dep\_Asi\_Tit

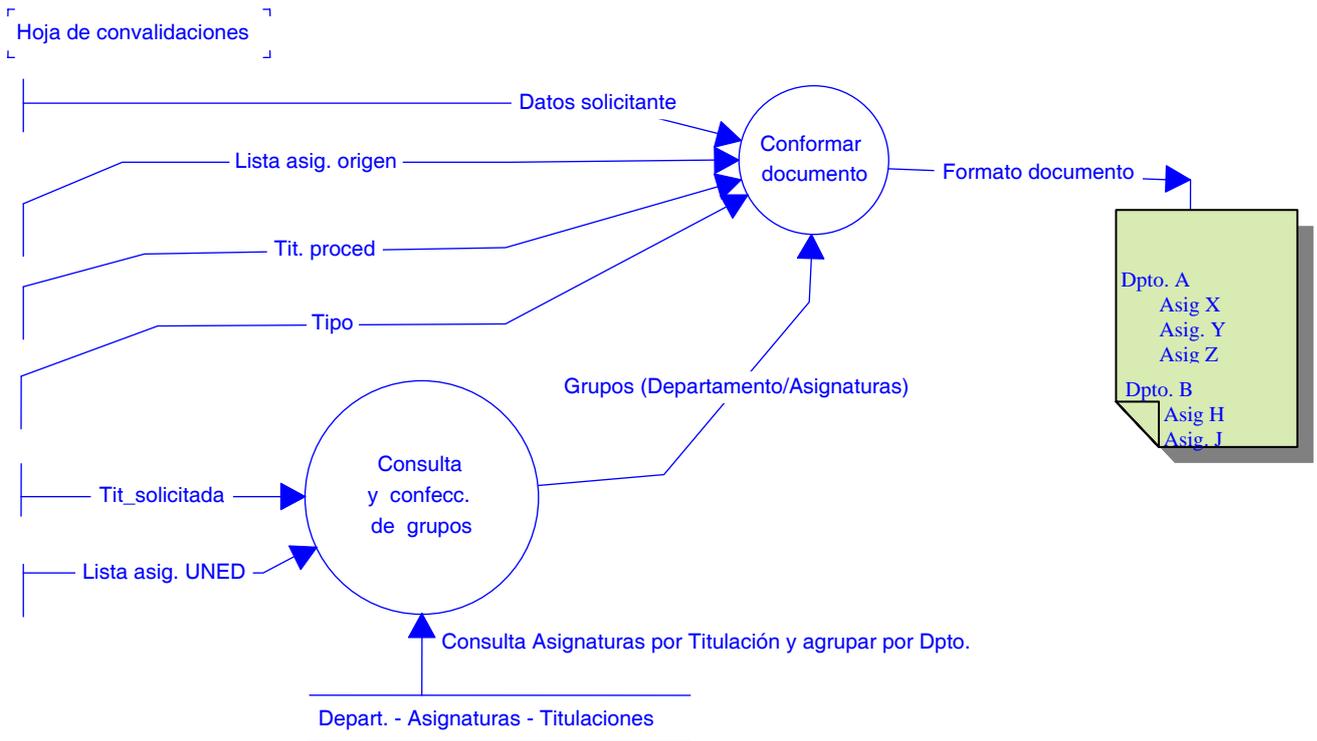


**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

### Modelado con DFD. Contexto (Nivel 0):



### Nivel 1:



**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

ASIGNATURA: INGENIERÍA DEL SOFTWARE (2º CURSO)

CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

CÓDIGO CARRERA:

Plan de estudios en extinción: 40=SISTEMAS y 41=GESTIÓN

Plan de estudios NUEVO: 53=SISTEMAS y 54=GESTIÓN

MATERIAL PERMITIDO: NINGUNO

MODELO: RESERVA AMÉRICA, GUINEA Y TANGER



Departamento de Ingeniería de  
Software y Sistemas Informáticos

---

**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. Razone la causas por las que se descomponen los DFDs (Diagramas de Flujo de Datos). ¿Tendría sentido aplicar la descomposición en otras notaciones de análisis como DTEs (Diagramas de Transición de Estados) o DERs (Diagramas Entidad-Relación)?

### Solución

Para mejorar la legibilidad del DFD de un sistema complejo, facilitar la construcción del DFD, permitir el desarrollo en paralelo del DFD e incluso la reutilización parcial del DFD, éste se divide en diferentes niveles de abstracción. Así, el DFD de contexto da la visión más abstracta del sistema: cuales son los agentes externos que interactúan con el sistema y que datos le suministran o solicitan. El DFD de contexto se concreta en el DFD de nivel 1 y, a su vez, cada proceso complejo de dicho DFD se concreta o “explota” en otros DFDs. Así sucesivamente, hasta alcanzar procesos elementales.

Por las razones antes mencionadas, tiene sentido aplicar esta estrategia, que en el diseño y la codificación suele denominarse “refinamiento progresivo”, en la representación de sistemas complejos con cualquier otra notación de análisis. Por ejemplo:

- Un DTE podría descomponerse “explotando” sus estados.
- Un DER podría descomponerse “explotando” sus entidades.

2. ¿Cómo afecta la independencia funcional al mantenimiento de un sistema?

### Solución

Los dos criterios básicos para medir la independencia funcional entre los módulos constituyentes de un sistema son la cohesión y el acoplamiento.

La cohesión se refiere a la coherencia del contenido de un módulo. Cuanto mayor sea la cohesión de los módulos, más fácil será localizar los cambios durante el mantenimiento.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

El acoplamiento entre módulos se refiere a la interrelación existente entre ellos. Cuanto menor sea el acoplamiento, menor será la propagación de los cambios durante el mantenimiento (la modificación de un módulo no implicará el cambio de los módulos con los que interactúa).

SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

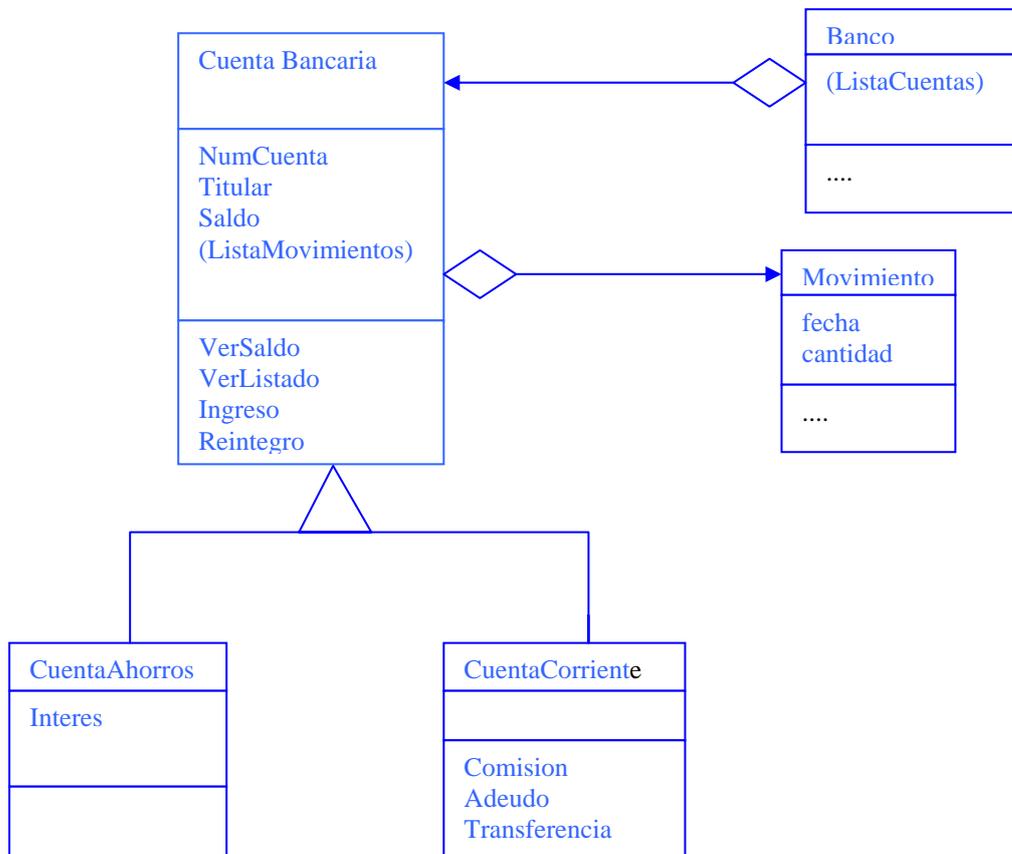
3. Un sistema informático de gestión bancaria opera con dos tipos de cuentas bancarias: la corriente y la de ahorros. Las cuentas disponen de un número que las identifica, tienen asociado un titular, una lista de los movimientos realizados hasta la fecha y almacenan el saldo disponible. Así mismo, las cuentas soportan operaciones como pedir el saldo y hacer ingresos o reintegros. Las cuentas de ahorros producen un interés que se calcula cada mes. Las cuentas corrientes no producen interés sino que reciben la carga de una comisión anual. Dichas cuentas corrientes facilitan diversas operaciones como adeudos domiciliarios y transferencias de dinero. Cada movimiento bancario consta de la fecha, la cantidad y el concepto de que se trata.

*Modele la situación anterior utilizando un diseño orientado a objetos. Utilice herencia y composición cuando se pueda.*

**Solución**

CLASES:

- Cuenta de ahorros
- Cuenta corriente
- Cuenta bancaria
- Movimientos
- Banco (Sistema de Gestión)



**Entregue la hoja de lectura óptica con sus datos junto con su examen.**



Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. ¿Qué se entiende por análisis del dominio? ¿Que ventajas produce en el desarrollo del producto software?

**SOLUCIÓN:**

Por **dominio** entenderemos el campo de aplicación en el que se encuadra el sistema que se construye. En cada campo o dominio existe desde siempre una manera específica de realizar las cosas y una terminología ya acuñada que debe ser respetada y tenida en cuenta. Esto es lo que denominaremos “*realizar un análisis del dominio de la aplicación*”.

Si bien las peculiaridades de cada aplicación hacen que necesariamente deba ser estudiada como un caso único, es importante analizar el dominio de la aplicación para situarla dentro de un entorno mucho más global. Para realizar este análisis es aconsejable estudiar los siguientes aspectos:

- Normativa que afecte al sistema
- Otros sistemas semejantes
- Estudios recientes en el campo de la aplicación
- Bibliografía clásica y actualizada: libros y artículos sobre el tema
- ... etc. ...

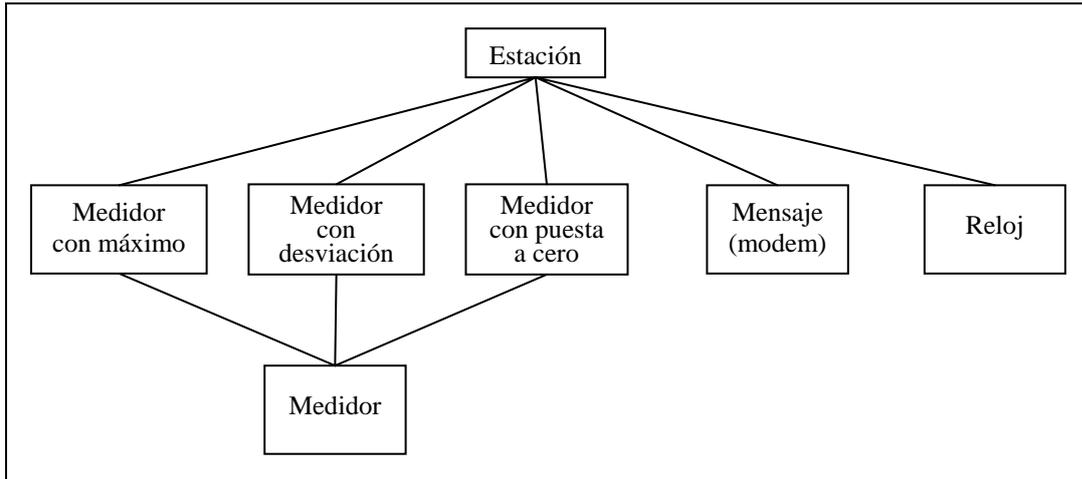
Este estudio facilitará la creación de un modelo más universal. Como ventajas de este enfoque se pueden citar las siguientes:

1. Facilitar la comunicación entre analista y usuario del sistema.
2. Creación de elementos realmente significativos del sistema.
3. Reutilización posterior del software desarrollado.

Apartado 2.1.2.5 del libro de texto, páginas 40 a 42.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

2. Dado el siguiente diagrama de arquitectura (construido con las relaciones de uso entre los módulos), correspondiente al ejemplo de una “estación meteorológica”; desarrolle el diagrama orientado a objetos equivalente.



**SOLUCIÓN:**

Uno posible es el correspondiente a la figura 4.14, de la página 187, del libro de texto.

#### SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

3. Se desea comprobar la corrección de un programa que calcula el precio de la estancia en un hotel. Los datos de entrada al programa son el día y el mes de la primera noche de estancia y el número de noches. El precio por noche es de:
- 100 € del 7 de enero al 30 de junio y del 1 de septiembre al 23 de diciembre.
  - 140 € del 1 de julio al 31 de agosto.

El hotel permanece cerrado del 24 de diciembre al 6 de enero. La salida que genera el programa es el precio en euros, siempre que se le hayan proporcionado valores correctos a la entrada. Si los datos de entrada no son adecuados, el programa devuelve el texto “datos no válidos”.

*Se pide desarrollar un juego de pruebas de error del programa, justificando la elección de los casos escogidos.*

**SOLUCIÓN:**

Puesto que sólo se conoce la especificación entrada-salida del programa, y no su estructura interna, el juego de pruebas será de tipo “caja negra”. Para ello emplearemos, de forma conjunta y complementaria, los métodos de “partición en clases de equivalencia” y “análisis de valores límite”.

A grandes rasgos podemos dividir el espacio de ejecución en tres subespacios:

- temporada baja (2 periodos: 7/1 al 30/6 y 1/9 al 23/12)
- temporada alta (1/7 al 31/8)
- temporada de cierre (24/12 al 7/1)

Y podemos establecer como valores límite las fechas de transición entre estas clases de equivalencia:

- 6/1 y 7/1
- 30/6 y 1/7
- 31/8 y 1/9

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

- 23/12 y 24/12

Un juego de casos de prueba podría ser:

CASOS VÁLIDOS		
ENTRADA	COMENTARIO	SALIDA
7/1, 5 noches	Valor límite temporada baja	500 euros (5 x 100)
24/3, 11 noches	1 <sup>er</sup> periodo temporada baja	1100 euros (11 x 100)
28/6, 3 noches	Valor límite temporada baja	300 euros (3 x 100)
25/6, 7 noches	Transición baja/alta, valor límite alta	740 euros (6 x 100 + 1 x 140)
1/7, 2 noches	Valor límite temporada alta	280 euros (2 x 140)
12/7, 20 noches	Temporada alta	2800 euros (20 x 140)
20/8, 12 noches	Valor límite temporada alta	1680 euros (12 x 140)
31/8, 2 noches	Transición alta/baja, valores límite	240 euros (100 + 140)
15/11, 6 noches	2 <sup>o</sup> periodo temporada baja	600 euros (6 x 100)
21/12, 3 noches	Valor límite temporada baja	300 euros (3 x 100)
28/6, 70 noches	Transición baja/alta/baja	9480 euros (8 x 100 + 62 x 140)
CASOS NO VÁLIDOS		
ENTRADA	COMENTARIO	SALIDA
3/1, 7 noches	Temporada de cierre	“datos no válidos”
6/1, 2 noches	Valor límite temporada de cierre	“datos no válidos”
19/12, 11 noches	Temporada de cierre	“datos no válidos”
24/12, 4 noches	Valor límite temporada de cierre	“datos no válidos”

Otro caso de estudio, que complica bastante el problema, sería la comprobación de la corrección de la fecha. También se deberían preparar casos de prueba según se trate o no de año bisiesto, etc.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**



Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. En la especificación de requisitos del “Videojuego de las Minas”, indique cuál (o cuáles) de los siguientes requisitos son “de recursos”, funcionales, “de capacidad” o “de operación”:
  - R.a En todo momento, el jugador estará informado de los segundos transcurridos y de las minas que todavía quedan por marcar del total de las minas ocultas inicialmente.
  - R.b Para moverse de una casilla a otra de las que la rodean, sólo será necesario pulsar una tecla una sola vez.
  - R.c Tiempo para situar inicialmente las minas  $\leq 1$  segundo.

**SOLUCIÓN:**

R.a: Requisito funcional.

R.b: Requisito de operación.

R.c: Requisito de capacidad.

2. Explique brevemente los conceptos de herencia y polimorfismo empleados en el diseño de software.

**SOLUCIÓN:**

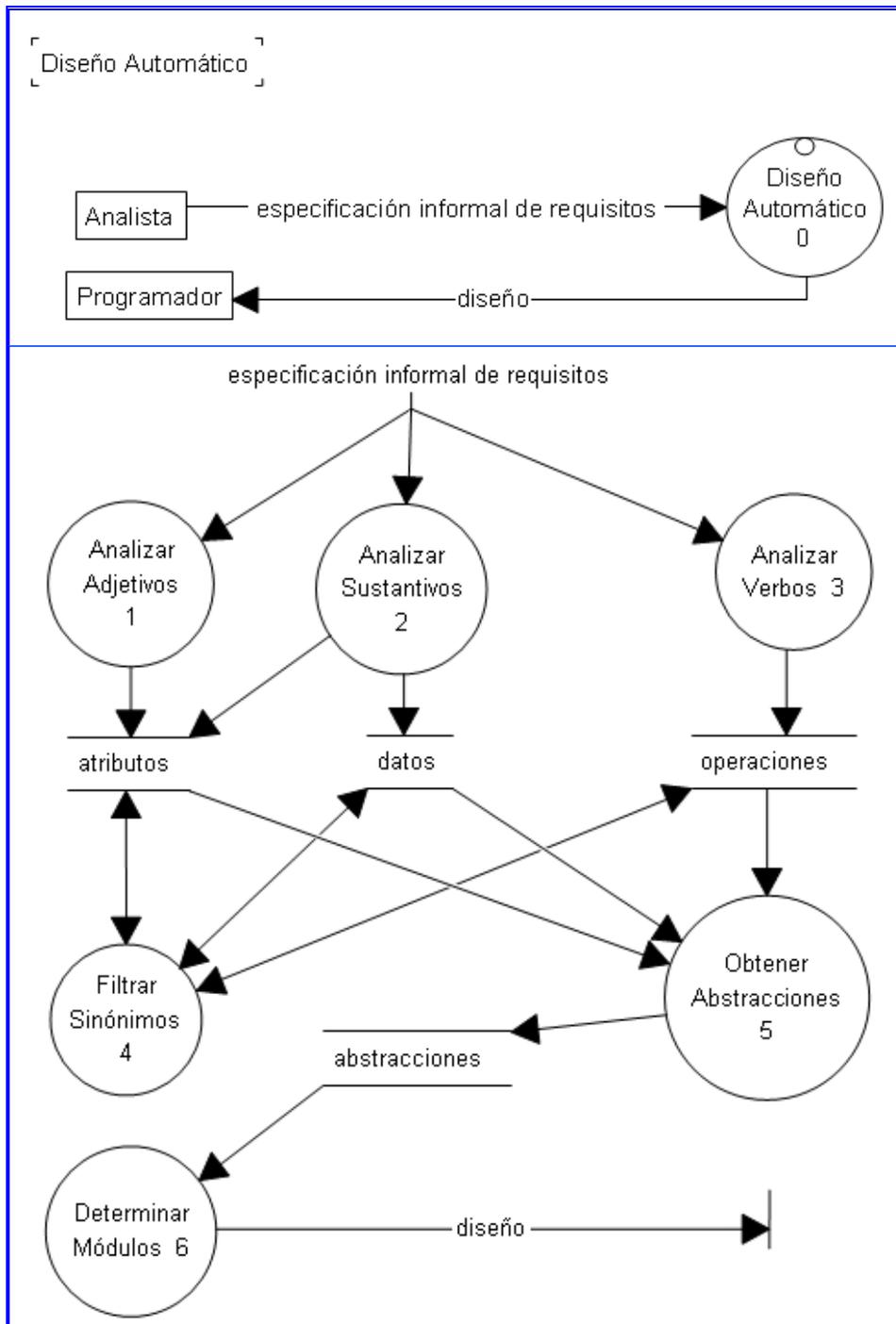
Apartados 3.2.7 y 3.2.8 del libro de texto.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

3. Se desea desarrollar un programa informático que, a partir de una especificación informal de requisitos, sea capaz de producir automáticamente un diseño aplicando el método de Abbott. **Modele el programa utilizando un Diagrama de Flujo de Datos.**

SOLUCIÓN:



Entregue la hoja de lectura óptica con sus datos junto con su examen.



**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. ¿Cuáles son las tareas o etapas fundamentales en el análisis de requisitos?

**SOLUCIÓN:**

1. Estudio del sistema en su contexto.
  2. Identificación de necesidades.
  3. Análisis de alternativas. Estudio de viabilidad.
  4. Establecimiento del modelo del sistema.
  5. Elaboración del documento de especificación de requisitos.
  6. Revisión continuada del análisis.
2. ¿Cuáles son las cualidades deseables de una descomposición modular en la fase de diseño de software?

**SOLUCIÓN:**

Apartado 4.1 del libro de texto.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

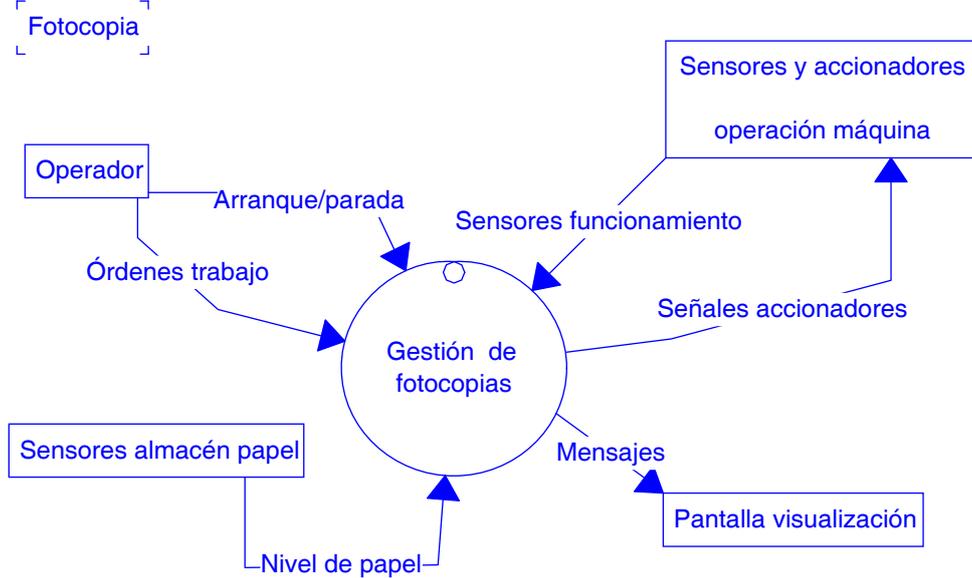
SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

3. Desarrolle un modelo de análisis en el que se refleje con claridad y sencillez el comportamiento del software de una fotocopidora. **Represente dicho modelo mediante DFD (hasta el nivel 1 ó 2) Y Diagramas de Transición de Estados.**

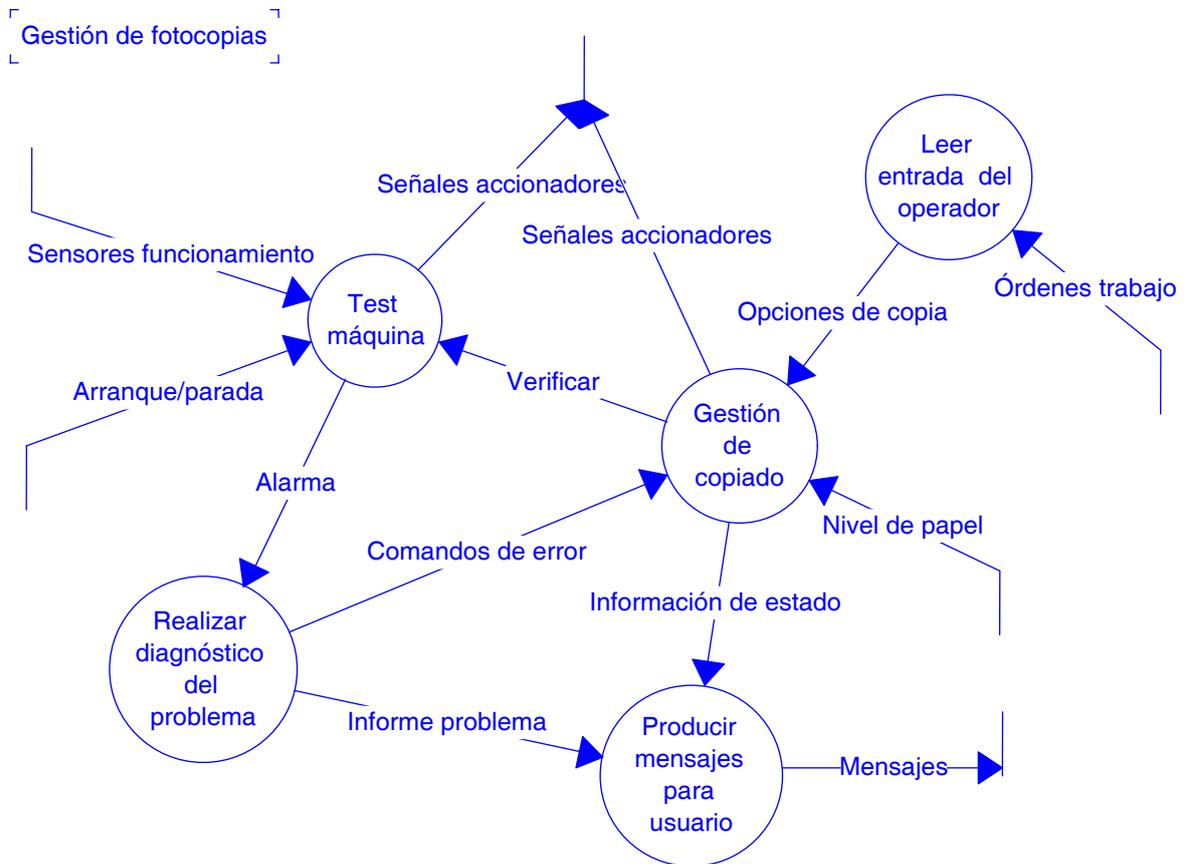
SOLUCIÓN:

Con Diagramas de Flujo de Datos:

Diagrama de contexto (nivel 0):

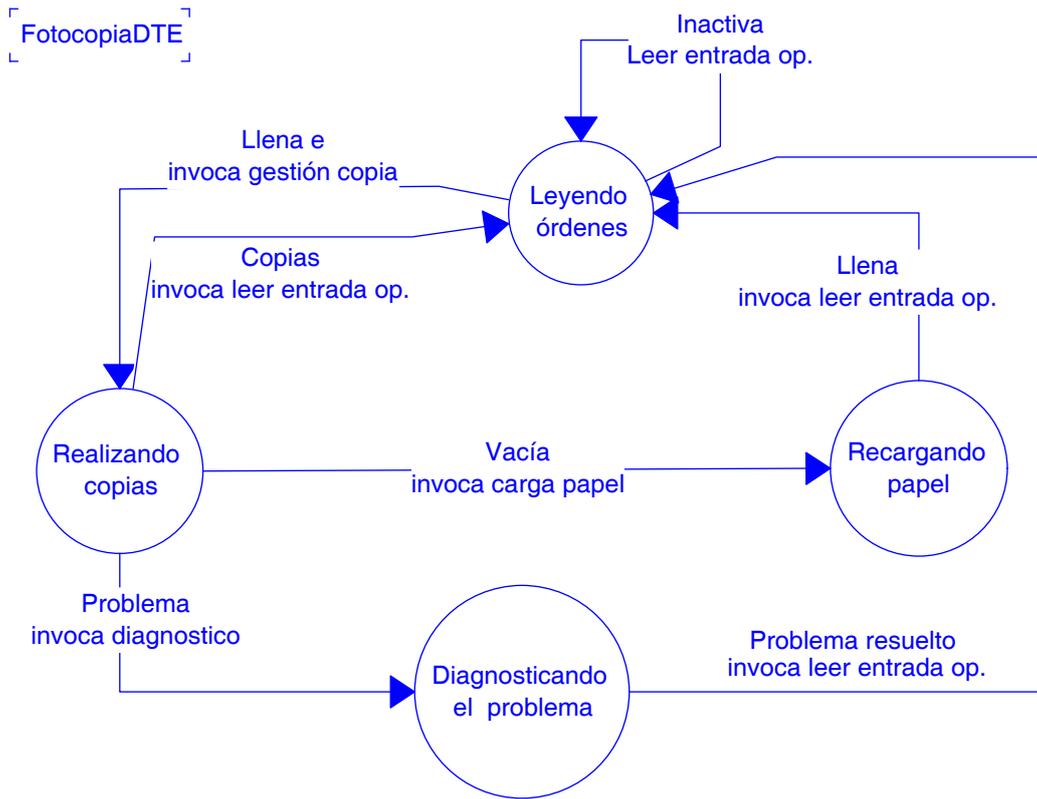


DFD0 (nivel 1):



Entregue la hoja de lectura óptica con sus datos junto con su examen.

Diagrama de Transición de Estados:



Entregue la hoja de lectura óptica con sus datos junto con su examen.



**Todas las preguntas de este ejercicio son eliminatorias en el sentido de que debe obtener una nota mínima en cada una de ellas. En cada pregunta teórica, que se valora con 2'5 puntos, la nota mínima es 1 punto; en la segunda parte (ejercicio de teoría aplicada que se valora con 5 puntos) la nota mínima que debe obtener es de 2 puntos.**

**¡ATENCIÓN! PONGA SUS DATOS EN LA HOJA DE LECTURA ÓPTICA QUE DEBERÁ ENTREGAR JUNTO CON EL RESTO DE LAS RESPUESTAS.**

Conteste a las preguntas teóricas, en cualquier orden, en hojas diferentes a las que utilice para la contestación de la segunda parte. En cada parte, la cantidad MÁXIMA de papel (de examen, timbrado) que puede emplear ESTÁ LIMITADA al equivalente a DOS (2) HOJAS de tamaño A4 (210 x 297 mm)

---

PRIMERA PARTE. PREGUNTAS TEÓRICAS (2'5 PUNTOS CADA UNA)

1. Defina el Ciclo de Vida del Software. Justifique la importancia de este concepto y la necesidad de su utilización.

**SOLUCIÓN:**

Como aparece en la página 7 del libro de texto: la ingeniería de software amplía la visión del desarrollo de software; a partir de una actividad esencialmente de programación, incorpora un conjunto amplio de actividades adicionales cuya distribución temporal es, precisamente, el ciclo de vida. Así, en la página 10, se define como: el conjunto de actividades involucradas en el proceso de desarrollo de software, así como su organización y distribución temporal, incluyendo el mantenimiento necesario durante su explotación. La justificación de su importancia reside en que la ingeniería de software tiene como objetivo conseguir hacer un seguimiento y control del proceso de desarrollo. Para ello, es necesario disponer de un modelo que identifique qué constituye dicho proceso. Es decir, el ciclo de vida establece una opción para la visibilidad del proceso que se quiere controlar.

2. Indique razonadamente si algún tipo de pruebas de unidades asegura la ausencia de defectos. Resuma brevemente los métodos propuestos en la asignatura para la elaboración de pruebas de caja negra.

**SOLUCIÓN:**

Ningún tipo de pruebas asegura la ausencia de defectos. Esto sólo puede conseguirse mediante técnicas de verificación formal. Los métodos para la elaboración de pruebas de caja negra están descritos en las páginas 274-279 del libro de texto.

**Entregue la hoja de lectura óptica con sus datos junto con su examen.**

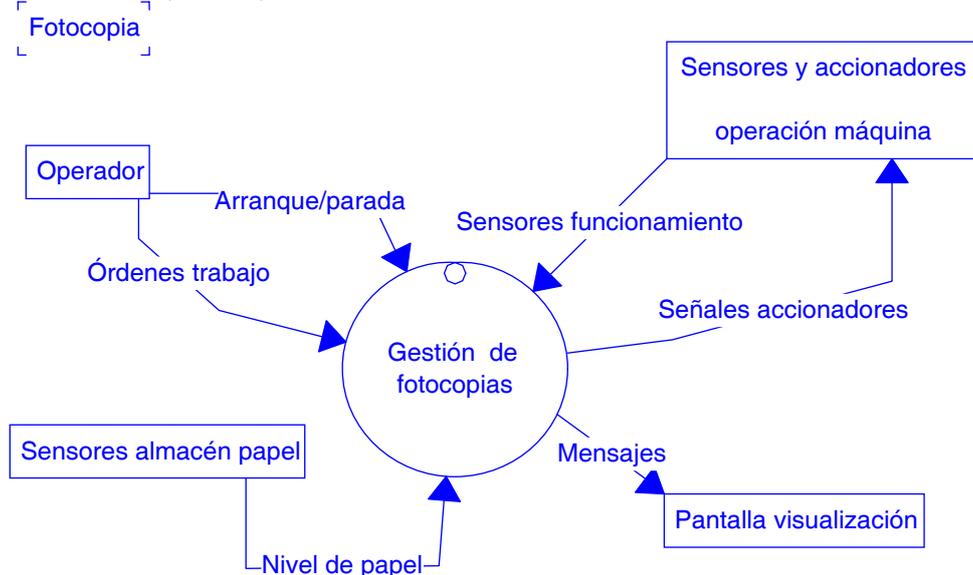
SEGUNDA PARTE. PREGUNTA DE TEORÍA APLICADA (MÁXIMO 5 PUNTOS)

3. Desarrolle un modelo de análisis en el que se refleje con claridad y sencillez el comportamiento del software de una fotocopidora (arranque y calibración, programación de tipo de copia, falta de papel, atasco, tóner, etc.) **Represente dicho modelo mediante DFD (hasta el nivel 1 ó 2) Y Diagramas de Transición de Estados.**

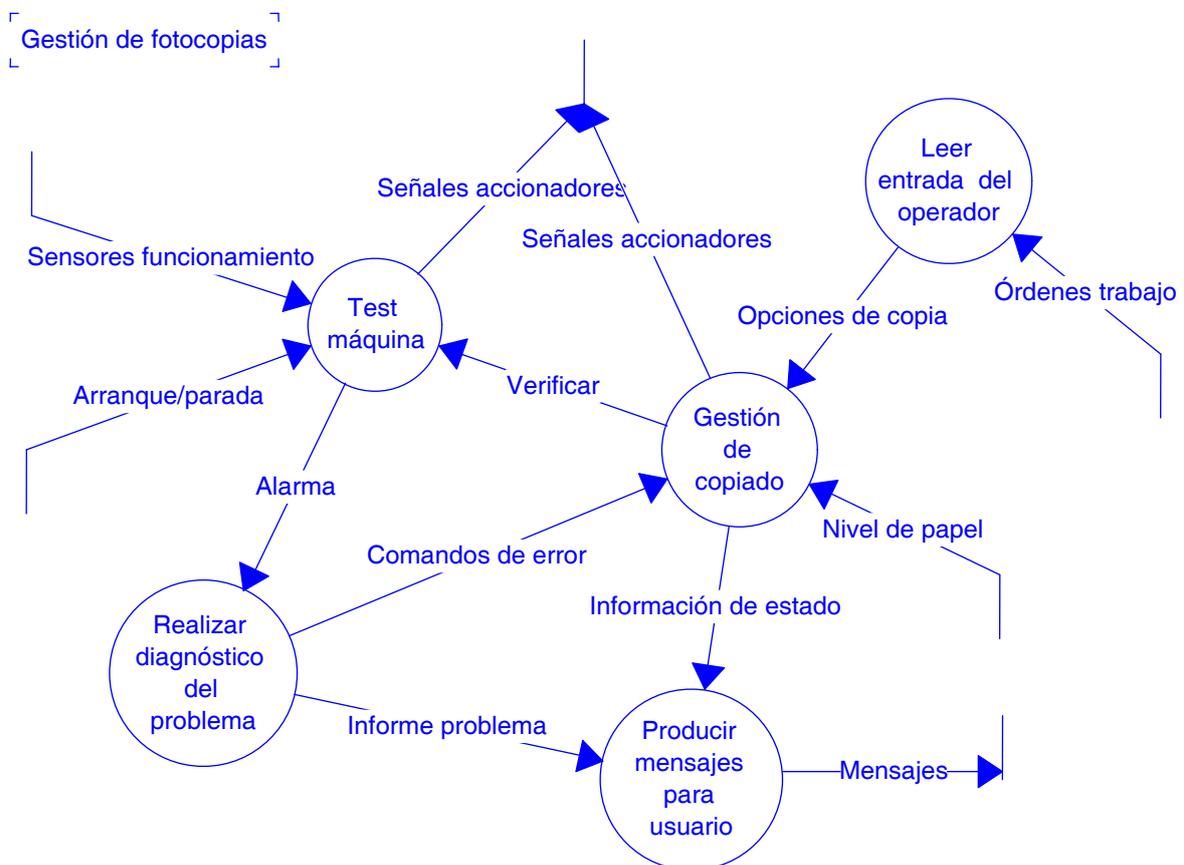
SOLUCIÓN:

Con Diagramas de Flujo de Datos:

Diagrama de contexto (nivel 0):

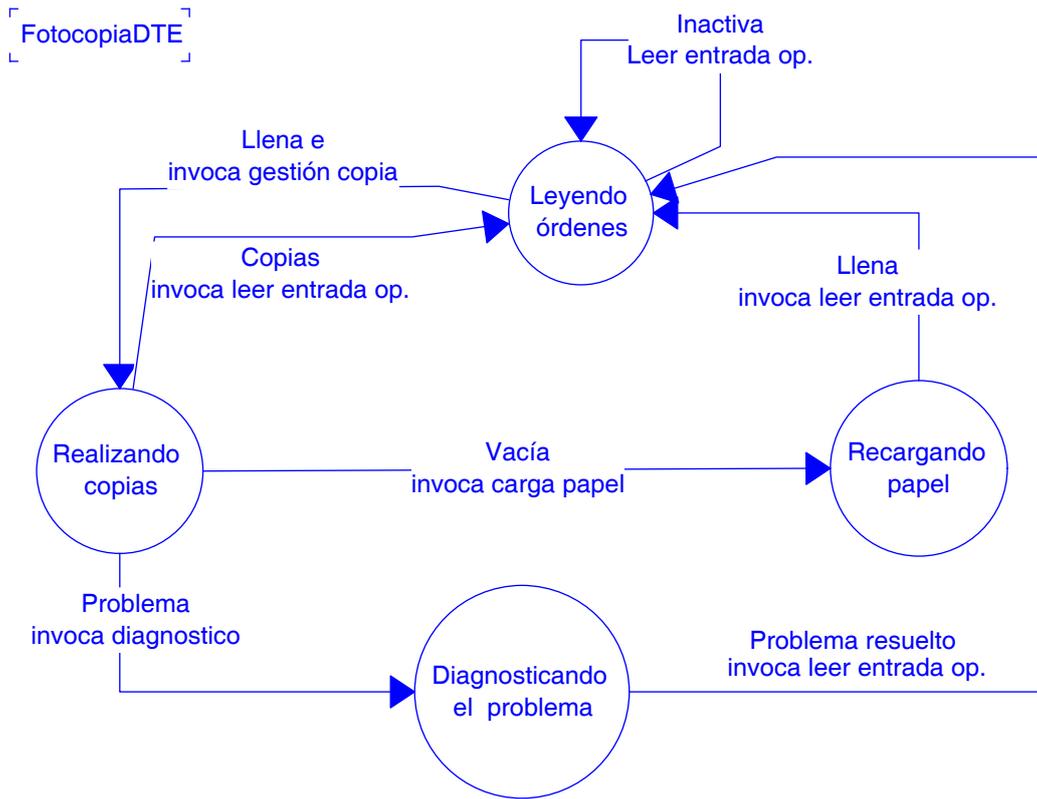


DFD0 (nivel 1):



Entregue la hoja de lectura óptica con sus datos junto con su examen.

Diagrama de Transición de Estados:



Entregue la hoja de lectura óptica con sus datos junto con su examen.