|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | **5** |
|  | TEMA |

Modelado de los requisitos: escenarios, información y clases de análisis

[5.1] Análisis de los requisitos

[5.2] Modelado basado en escenarios

[5.3] Modelos UML que proporcionan el caso de uso

[5.4] Conceptos de modelado de datos

[5.5] Modelado basado en clases

[5.6] Diagramas de clases

[5.7] Referencias

Esquema



Ideas clave

5.1. Análisis de los requisitos

El modelado de requisitos puede ser:

* Modelado basado en el **escenario de los requisitos**: centrándose en los actores del sistema a modelar.
* Modelado de **datos**: desde el punto de vista de la información del sistema.
* Modelado **orientado a clases**: para modelar basándose en el paradigma orientado a objetos. Representa a las clases del sistema, con sus atributos y métodos, y las interacciones entre ellas.
* Modelado **orientado al flujo**: representan las funcionalidades del sistema y la transformación que realizan de los datos.
* Modelado del **comportamiento**: muestra el comportamiento del sistema en base a eventos externos.

La fase de análisis y modelado de requisitos se centra «¿qué hay que hacer?». La fase de diseño que veremos en temas posteriores tratará el «¿cómo hay que hacerlo?».

Durante el análisis de requisitos se debe lograr tres objetivos:

1. Describir lo que quiere el cliente
2. Establecer una base para el posterior diseño del software
3. Definir el conjunto de requisitos que habrá que validar una vez construido el software.

En análisis de requisitos puede ser:

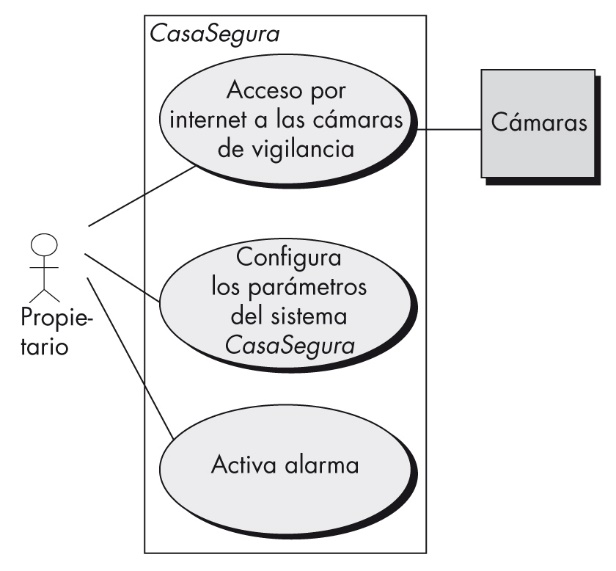


5.2. Modelado basado en escenarios

El modelado de requisitos basado en escenarios tiene como objetivo representar **cómo desea interactuar el usuario final con el sistema**.

Para llevarlo a cabo:

1. Se crean los casos de uso preliminares. Cada uno de ellos, define una funcionalidad del sistema, identificando, por un lado, a los actores implicados y, por otro lado, enumerando las características que debe cumplir esa funcionalidad.
2. Se refina cada caso de uso preliminar. Se analiza en profundidad cada caso de uso para identificar escenarios adicionales, condiciones de error y comportamientos alternativos.
3. Se escriben formalmente los casos de uso con al menos el siguiente contenido: actor principal y actores secundarios, objetivo, precondiciones, lista de características del escenario, excepciones y prioridad.
4. Se representan gráficamente los casos de uso con diagramas UML de casos de uso.

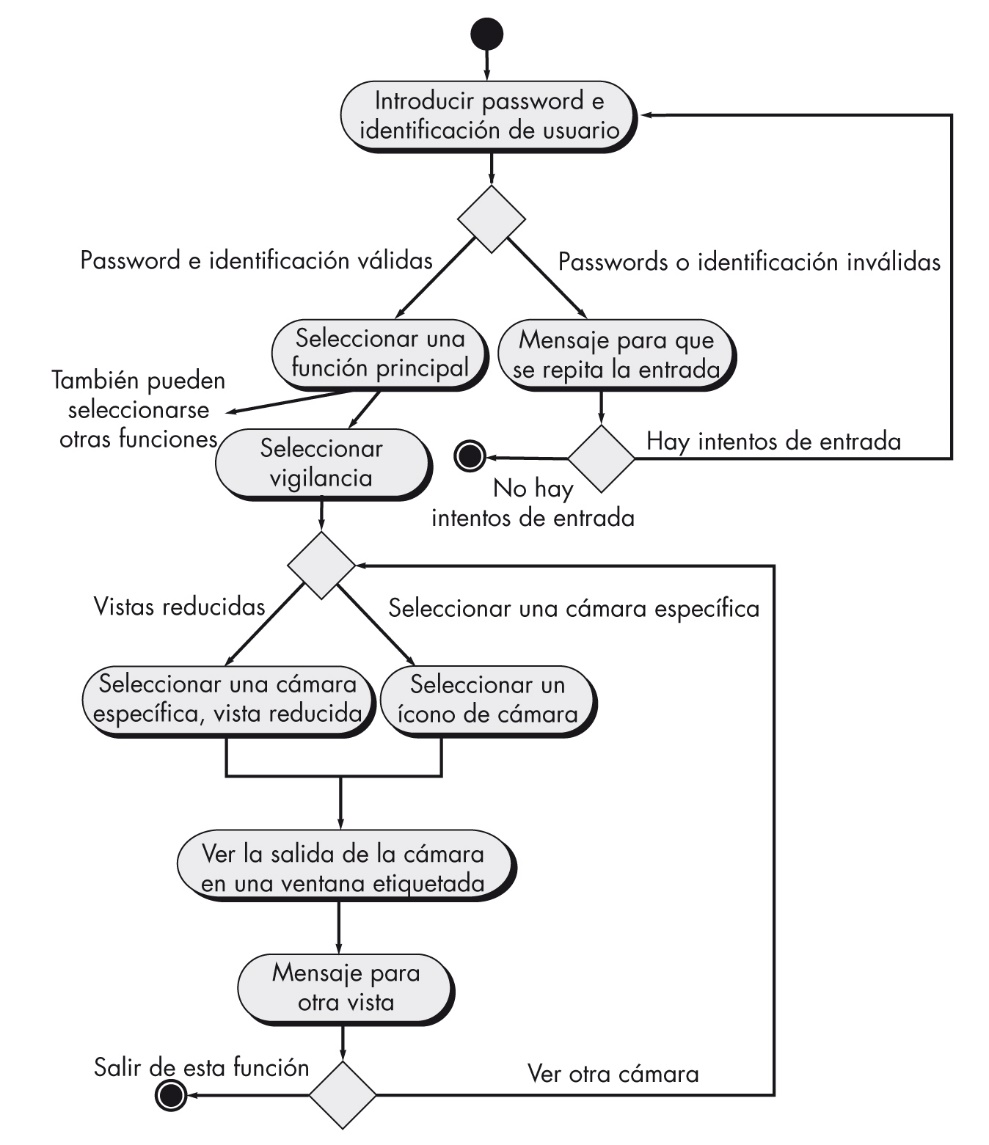


Fuente: Pressman, R. S. (2010). *Ingeniería del software.   
Un enfoque práctico* (7ª ed.) (p. 137). México: McGraw Hill.

5.3. Modelos UML que proporcionan el caso de uso

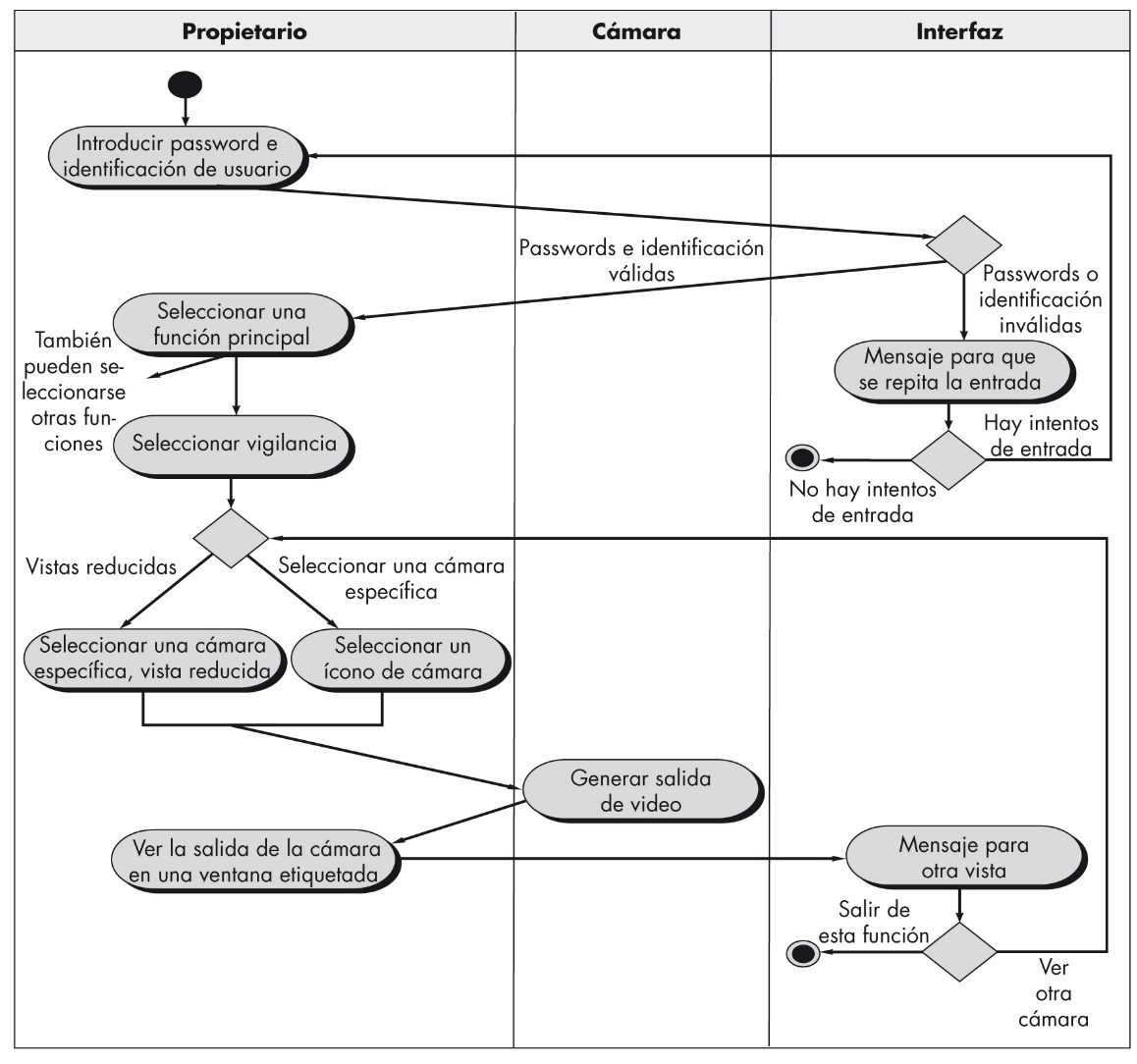
Los diagramas de actividades enriquecen la información del caso de uso:

* **Diagramas de actividades**: representa el flujo de interacción dentro del escenario.



Fuente: Pressman, R. S. (2010). *Ingeniería del software.   
Un enfoque práctico* (7ª ed.) (p. 138). México: McGraw Hill.

* **Diagramas de actividades con canal**: es un diagrama de actividad que añade información sobre el actor responsable de cada acción.



Fuente: Pressman, R. S. (2010). *Ingeniería del software.   
Un enfoque práctico* (7ª ed.) (p. 139). México: McGraw Hill.

5.4. Conceptos del modelado de datos



Los objetos de datos se interconectan entre sí mediante las relaciones.

Los **diagramas Entidad-Relación** permiten modelar los objetos de datos, sus atributos y sus relaciones.

5.5. Modelado basado en clases



El modelado de clases sigue los siguientes pasos:

1. **Identificación de las clases**, para lo que se analiza los casos de uso y se identifican los diferentes sustantivos. Posteriormente, de estos sustantivos, habrá que identificar los que realmente representan a clases del sistema, que pueden ser entidades externas, cosas, eventos, roles, unidades organizativas, lugares, estructuras, etc.
2. **Identificación de los atributos** de la clase con aquellos elementos que pertenezcan a la clase.
3. **Identificación de los métodos** de la clase con las operaciones que definen el comportamiento de la clase.
4. **Identificación de las relaciones** entre las diferentes clases para representar jerarquías, relaciones, asociaciones, agregaciones y dependencias.

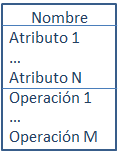
Si el sistema a modelar es muy grande, para facilitar su manejo, se pueden agrupar las clases en paquetes.

5.6. Diagramas de clases

Para modelar las clases (con sus atributos y operaciones) y las relaciones y asociaciones con otras clases se utilizan los diagramas UML de clases.

Cada clase se representa con una caja compuesta de tres partes:

* Nombre
* Atributos
* Operaciones



La **relación de generalización o herencia** se representa mediante una flecha con una punta triangular hueca que apunta al padre (o superclase).

La **relación de asociación** se representa mediante una línea que una a ambas clases. Si se pone una flecha en uno de los extremos, se indica que la relación es solamente en uno de los sentidos. También se puede indicar la multiplicidad de esta relación con un número o un rango de números.

La **relación de dependencia** se representa con una línea punteada indicando que cambios en una de ellas puede suponer cambios en la otra.

La **relación de agregación** se representa mediante un rombo hueco indicando la relación todo/parte.

La **relación de composición** es una relación de agregación que indica dependencia total de las partes con respecto al todo y se representa mediante un rombo relleno.

5.7. Referencias

Pressman, R. (2010). *Ingeniería del software* (7ª ed.). México: McGrawHill.

Lo + recomendado

No dejes de leer…

**Especificación de UML 2.0**

En este documento oficial de OMG encontrarás la especificación del estándar UML 2.0.

Accede al artículo desde el aula virtual o a través de la siguiente dirección web:

<http://www.omg.org/spec/UML/2.0/Superstructure/PDF>

No dejes de ver…

**Técnicas de persuasión**

Fragmento del programa El Hormiguero de Antena 3 en el que la colaboradora Elsa Punset presenta cuatro sencillas reglas de persuasión aplicables a las reuniones de captura de requisitos con un cliente.



Accede al vídeo desde el aula virtual o a través de la siguiente dirección web:

<https://www.youtube.com/watch?v=bK2vRu5gj8Q>

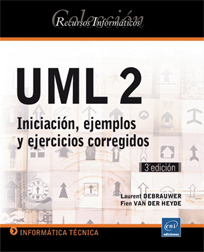
+ Información

A fondo

**UML 2. Iniciación, ejemplos y ejercicios corregidos**

Van Der Heyde, F. & Debrauwer, L. (2013). *UML 2. Iniciación, Ejemplos y Ejercicios Corregidos*. Barcelona: Ediciones Eni.

Libro sobre el modelado de software con UML 2 desde un punto de vista práctico. Presenta inicialmente una introducción a la orientación a objetos, posteriormente presenta los diferentes diagramas de UML 2, desde los diagramas de casos de uso, componentes, interacción, clases, estados, actividades, etc. de un modo muy didáctico y como multitud de ejercicios prácticos que facilita la comprensión y el aprendizaje.



Accede a una parte del libro desde el aula virtual o a través de la siguiente dirección web:

<http://books.google.es/books?id=drst-kOWZWsC&printsec=frontcover>

Enlaces relacionados

Página oficial de UML

Página web de UML de OMG (Object Management Group) que es un consorcio internacional de estándares tecnológicos que mantiene UML.



Accede a la página desde el aula virtual o a través de la siguiente dirección web:

<http://uml.org>

Recursos externos

StarUML

StarUML es una herramienta de modelado UML gratuita compatible con UML 2 que soporta once tipos de diagramas UML: clases, objetos, casos de uso, componentes, despliegue, secuencia, comunicación, actividad, etc.



Accede a la página desde el aula virtual o a través de la siguiente dirección web:

<http://staruml.io/download>

Documentación: <https://docs.staruml.io/>

Test

**1.** El modelado de requisitos que se centra en los actores del sistema a modelar es el:

A. Modelado de actores del sistema.

B. Modelado del comportamiento.

C. Modelado basado en escenarios.

D. Modelado orientado al flujo.

**2.** La escritura formal de un caso de uso:

A. No es necesaria si se realiza el diagrama UML de caso de uso.

B. Debe contener información del lenguaje de programación con el que se implementará.

C. Deberá incluir el conjunto de excepciones.

D. Se realiza en la fase de diseño.

**3.** Los diagramas de actividades:

A. Contienen la misma información que los diagramas de casos de uso.

B. Representan el flujo de interacción dentro del escenario.

C. Representan los estados por los que pasa un componente del sistema.

D. Contienen la lista de actividades a realizar por el jefe de proyecto.

**4.** Los diagramas de actividad con canal:

A. Son una variación de los diagramas de caso de uso.

B. Representa la identificación del actor o clase responsable de cada acción.

C. Se utilizan en el modelado estructurado, pero no en el modelado orientado a objetos.

D. Todas las respuestas son correctas.

**5.** Respecto al modelado de datos y sus interrelaciones:

A. Se utilizan los DFD (Diagramas de Flujo de Datos) para su modelado.

B. Se definen atributos públicos, privados y protegidos de los datos.

C. Se realiza posteriormente al diseño del sistema.

D. Se representan con los diagramas Entidad-Relación.

**6.** En referencia a las relaciones de asociación de los diagramas de clase:

A. Se representan mediante una flecha con un triángulo hueco.

B. Representan una relación jerárquica.

C. Siempre son bidireccionales.

D. Las agregaciones son un tipo especial de relaciones de asociación.

**7.** La representación gráfica de una clase en un diagrama de clase contiene:

A. Las precondiciones, las interrelaciones y las dependencias.

B. El nombre, los atributos y las operaciones.

C. Las generalizaciones, las asociaciones y las restricciones.

D. Los objetos, el tipo y la cardinalidad.

**8.** Indique cuál de las siguientes propiedades de la orientación a objetos se representa con un diagrama de clases:

A. Herencia.

B. Polimorfismo.

C. Abstracción.

D. Cohesión.