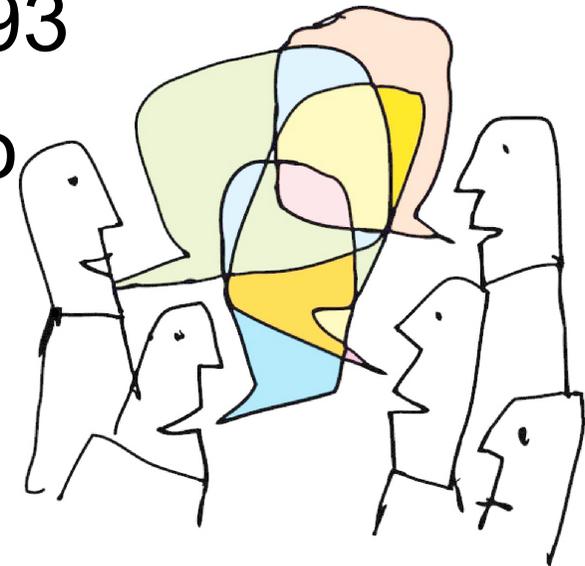


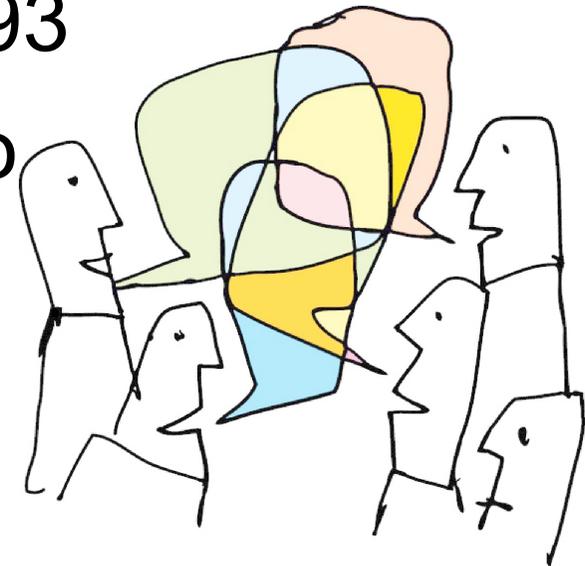
Requerimientos y Casos de Uso

- Contenido
 - Introducción
 - Técnicas de Educación de Requerimientos
 - Modelo IEEE/ANSI 830-1993
 - Modelado de Casos de Uso



Requerimientos y Casos de Uso

- Contenido
 - **Introducción**
 - Técnicas de Educación de Requerimientos
 - Modelo IEEE/ANSI 830-1993
 - Modelado de Casos de Uso



Introducción (1)

- Para que un esfuerzo de desarrollo de software tenga éxito, es esencial comprender perfectamente los requisitos del software.
- El análisis de requisitos es un proceso de:
 - Descubrimiento
 - Refinamiento
 - Modelización
 - Especificación
- El análisis de requisitos puede parecer una tarea relativamente sencilla, pero las apariencias engañan.
- El éxito de un producto depende de gran manera de la correcta realización de esta etapa.



Introducción (2)

- El análisis de requisitos facilita al ingeniero de software:
 - La especificación de la función y el rendimiento del software.
 - La descripción de la interfaz con otros elementos del sistema.
 - El establecimiento de las restricciones de diseño del software.
- Son tareas del análisis:
 - Reconocimiento del problema.
 - Evaluación y síntesis.
 - Modelización.
 - Especificación.
 - Revisión.



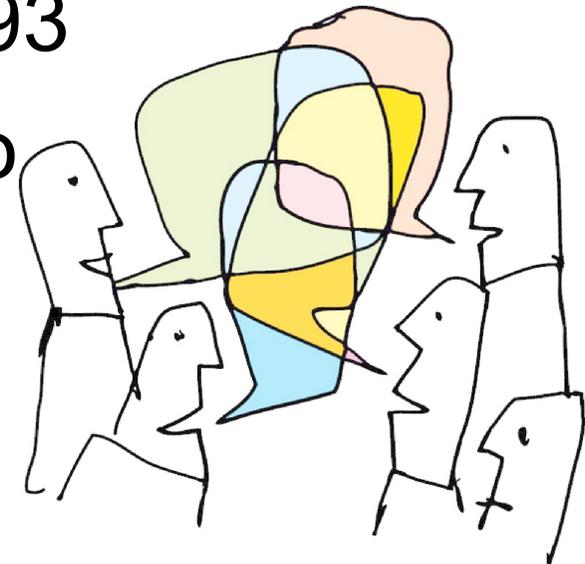
Introducción (3)

- El analista debe exhibir los siguientes rasgos de carácter:
 - Habilidad para comprender conceptos abstractos, reorganizarlos en divisiones lógicas y sintetizar soluciones basadas en cada división.
 - Habilidad para entresacar hechos importantes de fuentes conflictivas o confusas.
 - Habilidad para comprender entornos de usuario/cliente.
 - Habilidad para aplicar elementos del sistema de hardware y/o software a entornos de usuario/cliente.
 - Habilidad para comunicarse bien de forma escrita y verbal.



Requerimientos y Casos de Uso

- Contenido
 - Introducción
 - **Técnicas de Educación de Requerimientos**
 - Modelo IEEE/ANSI 830-1993
 - Modelado de Casos de Uso



Técnicas de Educación de Requisitos (1)

- Fuentes de requisitos:
 - **Libros y manuales:** son útiles para obtener conocimientos básicos del dominio y las clasificaciones existentes en él.
 - **Documentación formal:** documentos que contienen las políticas y procedimientos, los estándares, las normas y regulaciones, las leyes, etc., de un dominio.
 - **Documentación informal:** notas manuscritas, memos internos, ayudas de trabajo, etc., que circulan dentro de las organizaciones.
 - **Registros internos:** las empresas registran casos que se les presentan en formas de órdenes de reparación, fichas de clientes o pacientes, estudios o almacenamiento de casos, etc.



Técnicas de Educación de Requisitos (2)

- Fuentes de requisitos:

- **Presentaciones:** todo el material para formación, ya sea impartida o recibida, suele contener conocimientos expuestos de un modo especialmente claro.
- **Publicaciones especializadas:** son las versiones más actualizadas de los conocimientos del dominio del problema, expresadas en publicaciones especializadas.
- **Investigación:** son los resultados de las investigaciones que se estén llevando a cabo en forma de: datos empíricos, estudios, informes, etc.
- **Visitas:** son llevadas a cabo en los centros de trabajo para observar los casos “in situ” del proceso de resolución de problemas.
- **Humanos:** son las fuentes de conocimiento imprescindibles de donde se obtiene la mayor parte de los conocimientos.



Técnicas de Educación de Requisitos (3)

- Las técnicas de educación de requisitos son:
 - Estudio de Documentación
 - Entrevistas
 - Observación de Tareas Habituales
 - Incidentes Críticos
 - Cuestionarios



Técnicas de Educación de Requisitos (3)

- Las técnicas de educación de requisitos son:
 - **Estudio de Documentación**
 - Entrevistas
 - Observación de Tareas Habituales
 - Incidentes Críticos
 - Cuestionarios



Estudio de Documentación (1)

- Partes de los conocimientos relevantes del dominio de la aplicación está presente en manuales, libros u otra forma escrita.
- Normalmente, el analista debe ser capaz de usar los documentos sin asistencia externa, pero es conveniente contar con un experto en el dominio que:
 - Explique la terminología usada en la documentación.
 - Proporcione detalles omitidos en los documentos.
 - Indique el material relevante de toda la colección de manuales.
 - Señale los puntos donde la práctica real difiere de los procedimientos.



Estudio de Documentación (2)

- Con la información es bien obtenida en el estudio de la documentación, el tiempo del análisis disminuye drásticamente.
- No existen técnicas o herramientas aceptadas para llevar a cabo de modo automático el estudio de la documentación.
- Una técnica muy difundida es la del **análisis estructural de textos**, que extrae conceptos fundamentales del dominio buscando estructuras preestablecidas.
- Para funcionar, el análisis estructural de textos necesita tener descriptas las estructuras textuales.



Estudio de Documentación (3)

- Las estructuras textuales fundamentales son:
 - **Definiciones:** introducción de un concepto nuevo en el texto.
 - **Afirmaciones:** una afirmación es una frase que establece una verdad.
 - **Leyes:** las leyes de un dominio establecen sus principios básicos, así como las reglas que fijan el funcionamiento de objetos del dominio.
 - **Procedimientos:** son los pasos para la resolución de problemas en el dominio.



Técnicas de Educación de Requisitos (3)

- Las técnicas de educación de requisitos son:
 - Estudio de Documentación
 - **Entrevistas**
 - Observación de Tareas Habituales
 - Incidentes Críticos
 - Cuestionarios



Entrevistas (1)

- La entrevista es el método más común y familiar para educir requerimientos.
- La entrevista consiste en la interacción sistemática de un analista con un usuario/experto del dominio para extraer los conocimientos de éste.
- Este método de educción puede estructurarse en varios grados y de distintas maneras, pero los que nos interesan son:
 - Entrevista Abierta
 - Entrevista Estructurada



Entrevistas (1)

- La entrevista es el método más común y familiar para educir requerimientos.
- La entrevista consiste en la interacción sistemática de un analista con un usuario/experto del dominio para extraer los conocimientos de éste.
- Este método de educción puede estructurarse en varios grados y de distintas maneras, pero los que nos interesan son:
 - **Entrevista Abierta**
 - **Entrevista Estructurada**



Entrevista Abierta (1)

- También llamada no estructurada; el analista plantea, más o menos espontáneamente, preguntas al usuario.
- Que se pregunte espontáneamente no significa que esta técnica no necesite planificación y control.
- Las preguntas típicas para comenzar una entrevista abierta son:
 - Qué es lo que espera que el producto haga.
 - Cómo resuelve actualmente los problemas.
 - Cuáles son las personas involucradas.
 - Quién va a utilizar la solución.
 - Puede mostrarme el entorno donde se resolverá el problema.



Entrevista Abierta (2)

- Es correcto que el analista intercale preguntas, aunque sin interrumpir, del tipo:
 - Qué hace a continuación.
 - Puede describir lo que quiere decir con eso.
 - Por qué hace eso.
- El comienzo de la mayoría de los sistemas comienza con entrevistas abiertas, generalmente con usuarios de administración.
- La principal ventaja de la entrevista abierta es que, de una manera muy simple, genera rápidamente una gran cantidad de conocimientos sobre la terminología y componentes principales de un dominio.



Entrevista Abierta (3)

- Lo que distingue a la entrevista abierta de los demás métodos es que permite educir información imprevista, en el orden que se desee y con el detalle que se quiera.
- Por el contrario, tiene como inconveniente que es muy consumidora de tiempo y no sirve para refinar las versiones iniciales de un sistema.



Entrevistas (1)

- La entrevista es el método más común y familiar para educir requerimientos.
- La entrevista consiste en la interacción sistemática de un analista con un usuario/experto del dominio para extraer los conocimientos de éste.
- Este método de educción puede estructurarse en varios grados y de distintas maneras, pero los que nos interesan son:
 - Entrevista Abierta
 - **Entrevista Estructurada**



Entrevista Estructurada (1)

- Es una alternativa potente y conceptualmente simple.
- La entrevista estructurada combina una Observación de tareas habituales con una entrevista abierta.
- En las entrevistas estructuradas, el analista, una vez marcado el tema y la profundidad con que se desea tratarlo, planifica todas las preguntas que debe plantear al usuario durante la sesión.
- Esta planificación debe incluir una formulación y agrupación de cuestiones de manera lógica.



Entrevista Estructurada (2)

- Las preguntas típicas en una entrevista estructurada son:
 - Podría explicarme este concepto en mayor detalle.
 - Qué ocurre en este punto.
 - No entiendo porque se hizo esto.
 - Por qué se planteó ese problema.
 - Es correcta esta secuencia.
 - Le parece que este diagrama ilustra el concepto que explicó.
 - Están incluidos aquí todos los conceptos relacionados con este tema.
- El tipo de respuestas que el analista debe esperar son:
 - Si
 - No
 - Si, hago esto aquí.
 - Se ha olvidado ésta.



Entrevista Estructurada (3)

- La entrevista estructurada se usa normalmente en la segunda parte del proceso de educación de requisitos.
- Esta técnica permite adquirir conocimientos específicos, de grano fino, y rellenar partes que le falten al analista.
- El modo de controlar la sesión es, fundamentalmente, a través de las preguntas. El analista guía al usuario al nivel de detalle deseado mediante las cuestiones que plantea.
- Si a través de las preguntas de una entrevista estructurada aparece un nuevo conjunto de conocimientos, se deben suspender las preguntas que faltan y comenzar una entrevista abierta sobre el tema nuevo.



Entrevistas (1)

- La entrevista es el método más común y familiar para educir requerimientos.
- La entrevista consiste en la interacción sistemática de un analista con un usuario/experto del dominio para extraer los conocimientos de éste.
- Este método de educción puede estructurarse en varios grados y de distintas maneras, pero los que nos interesan son:
 - Entrevista Abierta
 - Entrevista Estructurada



Entrevistas (2)

- El ciclo de educación en una entrevista es el siguiente:
 - Preparación de la sesión:
 - Información a tratar
 - Amplitud, profundidad.
 - Técnica adecuada.
 - Preparación de preguntas.
 - Sesión:
 - Repaso del análisis de la última sesión.
 - Explicación al usuario de los objetivos de la nueva sesión.
 - Educación.
 - Resumen del usuario.



Entrevistas (3)

- Transcripción
- Análisis de la Sesión:
 - Lectura para obtención de una visión general.
 - Extracción de conocimientos concretos.
 - Lectura para recuperar detalles olvidados.
 - Crítica para mejoras por parte del analista.
- Evaluación:
 - ¿Se han conseguido los objetivos?
 - ¿Es necesario volver sobre el mismo objetivo?.
 - Número y tipo de sesiones necesarias para cubrir el área.



Entrevistas (4)

- Cada entrevista debe ser identificada de manera única, detallando fecha, hora y lugar de realización, personas involucradas, duración, resumen del tema tratado y el medio de registro (cinta, papel, etc.).
- Es importante transcribir el contenido de la cinta a papel, ya que permite un análisis detallado del contenido.
- Consejos para la educación satisfactoria:
 - Conseguir la cooperación del usuario.
 - No imponer el entendimiento del analista sobre el quehacer del usuario.
 - Limitar las sesiones en duración y en contenido.
 - Intentar comprobar la información.
 - No limitarse al diálogo. Exigir documentación
 - No espaciar demasiado las entrevistas.
 - Elegir adecuadamente el lugar de la entrevista.



Entrevistas (5)

- Uno de los principales problemas de las entrevistas son los equívocos, ya que en la comunicación entre dos personas puede darse que falle alguna de las partes o ambas, y produzcan malentendidos.
- El analista debe tener en cuenta que en toda interacción con el usuario aparecen probabilidades muy altas de que se produzcan:
 - Asunciones falsas
 - Tendenciosidad
 - Malentendidos
 - Fallos



Técnicas de Educción de Requisitos (3)

- Las técnicas de educación de requisitos son:
 - Estudio de Documentación
 - Entrevistas
 - **Observación de Tareas Habituales**
 - Incidentes Críticos
 - Cuestionarios



Observación de Tareas Habituales (1)

- Con frecuencia, la mejor forma de descubrir cómo un usuario realiza una tarea o función del sistema es observarlo trabajar en un problema real habitual.
- La primera decisión que se debe tomar al respecto es cómo registrar las prestaciones del usuario (observar, tomar notas, grabar, etc.)
- En la OTH, el analista no interfiere en la actuación del usuario en la solución de sus tareas reales cotidianas.
- El propósito es discernir acerca de la complejidad de la tarea y qué características debe presentar el interfaz de usuario final.



Observación de Tareas Habituales (2)

- Su mayor inconveniente proviene de que el analista actúa como un observador pasivo, lo que hace que sea poco práctica por consideraciones de tiempo.
- En algunos dominios es inútil, por cuestiones de privacidad.
- La OTH suele realizarse en la primera fase del proceso de educación de requisitos y proporciona conocimientos básicos del dominio y ayuda a que el analista comprenda la tarea del usuario.



Técnicas de Educción de Requisitos (3)

- Las técnicas de educción de requisitos son:
 - Estudio de Documentación
 - Entrevistas
 - Observación de Tareas Habituales
 - **Incidentes Críticos**
 - Cuestionarios



Incidentes Críticos



- En esta técnica, se le pide al usuario que describa casos especialmente interesantes o difíciles que se le hayan presentado y deberá contar cómo los resolvió.
- Esta forma de preguntar puede proporcionar muy rápidamente información útil y sentar bases para futuras discusiones.
- El usuario puede olvidar detalles esenciales a la hora de resolver casos normales, pero los casos complejos le estimulan de modo que los detalles salen a la luz.
- Una variación menos interesante es plantear al usuario casos imaginarios, inventando alternativas sobre un caso concreto.

Técnicas de Educación de Requisitos (3)

- Las técnicas de educación de requisitos son:
 - Estudio de Documentación
 - Entrevistas
 - Observación de Tareas Habituales
 - Incidentes Críticos
 - **Cuestionarios**



Cuestionarios (1)

- La técnica de cuestionarios consiste en realizar una Entrevista Estructurada al experto de forma indirecta, a través de cuestionarios.
- Esta técnica tiene la ventaja de ser una forma eficiente de acumular información, ya que el usuario puede rellenar los cuestionarios en un ambiente todo lo placentero y agradable que desee.
- Los cuestionarios son particularmente adecuados para educir información incierta, ya que las respuestas verbales de la gente no son muy dignas de confianza.



Cuestionarios (2)

- La gente no es buena en estimar probabilidades pues, generalmente, se tienden a sobreestimar los valores altos.
- Educar la certidumbre estimada usando escala de respuestas preimpresas puede producir estimaciones mucho más exactas.
- La elaboración del cuestionario debe hacerse previamente por el analista y debe estar destinado a un usuario o conjunto de usuarios específico.
- Debe adjuntarse fecha y hora, resumen y personas que intervienen.



Técnicas de Educación de Requisitos (3)

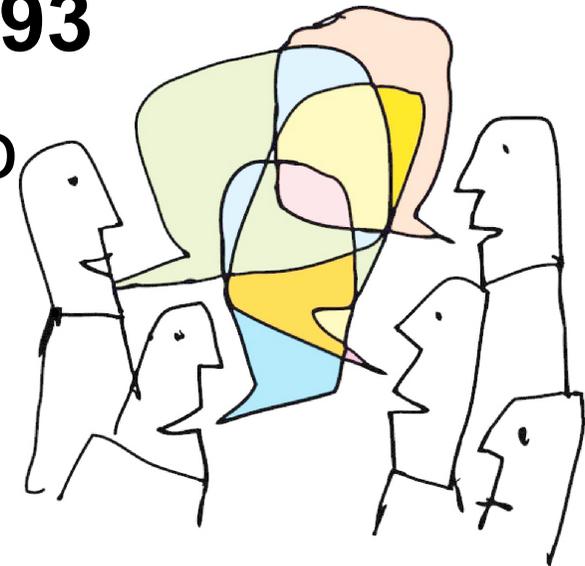
- Las técnicas de educación de requisitos son:
 - Estudio de Documentación
 - Entrevistas
 - Observación de Tareas Habituales
 - Incidentes Críticos
 - Cuestionarios



Requerimientos y Casos de Uso

- Contenido

- Introducción
- Técnicas de Educación de Requerimientos
- **Modelo IEEE/ANSI 830-1993**
- Modelado de Casos de Uso



Modelo IEEE/ANSI 830-1993 (1)

- El IEEE/ANSI 830-1993 es un formato de especificación de requisitos software propuesto por la oficina nacional de estándares de E.U. Y el IEEE.
- El título es:
 - Recommended Practice for Software Requirements Specifications.
 - Práctica recomendada para la especificación de requisitos de software.
- **Ámbito:**
 - Dentro del desarrollo de sistemas de información, en los procesos del ciclo de vida del software, corresponde a la etapa de especificación de requisitos



Modelo IEEE/ANSI 830-1993 (2)

- Es un enfoque recomendado de cómo especificar los requisitos del software, con el fin de conseguir un documento completo y sin ambigüedades que ayude:
 - A los clientes o compradores a describir con precisión lo que quieren obtener.
 - A los suministradores a comprender exactamente lo que el cliente quiere.
 - A los individuos a cargo de desarrollar una especificación de requisitos del software (ERS) normalizada para su organización, de definir el formato y contenido de esas especificaciones, o bien de comprobar su calidad.



Modelo IEEE/ANSI 830-1993 (3)

- Los beneficios de una buena ERS son evidentes:
 - Servir como base para el acuerdo entre cliente y proveedor sobre lo que el software ha de hacer.
 - Reducir el esfuerzo de desarrollo.
 - Proporcionar la base para la estimación de costes y plazos.
 - Proporcionar el punto de partida para las actividades de validación y verificación.
 - Facilitar la transferencia del software, a nuevos usuarios o nuevas máquinas.
 - Servir de base para ampliaciones o mejoras.



IEEE

Modelo IEEE/ANSI 830-1993 (4)

- La norma está orientada fundamentalmente a la especificación de requisitos para software cuyo desarrollo se va a contratar, pero puede aplicarse también como ayuda en la selección de productos software comerciales o desarrollados internamente.
- Cuando el software está integrado en sistemas más amplios, como por ejemplo equipos de medicina, es probable que haya que tratar aspectos adicionales a los contemplados por esta norma.
- Esta "práctica recomendada" describe el proceso de creación de un producto (la especificación de requisitos del software o ERS) y el contenido del producto en sí.
- Puede utilizarse directamente o como modelo para un norma más específica.



Modelo IEEE/ANSI 830-1993 (5)

- La estructura básica que vamos a utilizar es la siguiente:

1. Análisis de Requisitos del Sistema.

1.1 Identificación de los usuarios participantes.

1.2. Planificación y Realización de entrevistas.

1.3. Catálogo de Requisitos del Sistema.

1.3.1. Objetivos y Alcance del Sistema.

1.3.2. Definiciones, Acrónimos y Abreviaturas.

1.3.3. Descripción general

1.3.4. Requisitos funcionales (A integrar con el Modelado de Casos de Uso)

1.3.5. Requisitos de usuario y tecnológicos

1.3.6. Requisitos de interfaces externas

1.3.7. Requisitos de rendimiento

1.3.8. Requisitos de desarrollo y restricciones de diseño



Modelo IEEE/ANSI 830-1993 (6)

1. Análisis de Requisitos del Sistema

- Corresponde a la etapa de análisis del ciclo de vida.
- Este módulo tiene como objetivo analizar y documentar las necesidades funcionales que deberán ser soportadas por el sistema a desarrollar.
- Para ello, se identificarán los requisitos que ha de satisfacer el nuevo sistema mediante entrevistas, el estudio de los problemas de las unidades afectadas y sus necesidades actuales.
- Además de identificar los requisitos se deberán establecer las prioridades, lo cual proporciona un punto de referencia para validar el sistema final que compruebe que se ajusta a las necesidades del usuario.



Modelo IEEE/ANSI 830-1993 (7)

1.1. Identificación de los usuarios participantes

- Los objetivos de esta tarea son identificar a los responsables de cada una de las unidades implicadas y a los principales usuarios implicados. Para ello se consideran los siguientes aspectos:
 - Incorporación de usuarios al equipo de proyecto.
 - Conocimiento de los usuarios de las funciones a automatizar.
 - Repercusión del nuevo sistema sobre las actividades actuales de los usuarios.
 - Implicaciones legales del nuevo sistema
- Es de destacar la necesidad de una participación activa de los usuarios del futuro sistema en las actividades de desarrollo del mismo, con objeto de conseguir la máxima adecuación del sistema a sus necesidades y facilitar el conocimiento paulatino de dicho sistema, permitiendo una rápida implantación.



Modelo IEEE/ANSI 830-1993 (8)

1.2. Planificación y Realización de Entrevistas.

- Esta tarea tiene como finalidad capturar los requisitos de usuarios para el desarrollo del sistema. Se deben realizar entrevistas de tipo abierta y estructurada y el estudio de la documentación asociada. El proceso comprende:
 - Planificar las entrevistas a realizar: en la planificación se incluirá fecha, hora y lugar de la entrevista, duración estimada y guión de la entrevista.
 - Realizar las entrevistas y documentarlas debidamente.
 - Documentar los requisitos identificados con sus prioridades.
- A partir de las entrevistas realizadas con los responsables y usuarios, se identifican los requisitos que debe cumplir el sistema y se establecerá una prioridad para los mismos, de acuerdo a las necesidades expresadas por los usuarios y a los objetivos a cubrir por el nuevo sistema.



Modelo IEEE/ANSI 830-1993 (9)

1.3. Catálogo de Requisitos del Sistema

- El objetivo de la especificación es definir en forma clara, precisa, completa y verificable todas las funcionalidades y restricciones del sistema que se desea construir.
- Esta documentación está sujeta a revisiones por el grupo de usuarios que se recogerán por medio de sucesivas versiones del documento, hasta alcanzar su aprobación por parte del grupo de usuarios.
- Una vez aprobado, servirá de base al equipo de desarrollo para la construcción del nuevo sistema.



Modelo IEEE/ANSI 830-1993 (10)

1.3.1. Objetivos y alcance del sistema

- En esta etapa se detallan los objetivos del sistema, describiendo brevemente QUÉ es lo que el sistema debe hacer.
- En el alcance del sistema se describe en lenguaje natural el ámbito del sistema, su dominio y sus límites.

1.3.2. Definiciones, acrónimos y abreviaturas

- Esta etapa tiene como fin establecer el vocabulario de términos que forman parte del sistema, de manera que TODOS los participantes "hablen el mismo idioma".



Modelo IEEE/ANSI 830-1993 (11)

1.3.3. Descripción general

- Esta sección nos presenta una descripción general a grandes rasgos del sistema con el fin de conocer las principales funciones que debe soportar, los datos asociados, las restricciones impuestas y cualquier otro factor que pueda influir en la construcción del mismo.
- Una buena manera de realizar la descripción es plantearla con un enfoque descendente; es decir, a nivel subsistemas, detallando las funciones por debajo de los mismos.

1.3.4. Requisitos funcionales



- Descripción en lenguaje natural de las funciones desglosadas en la etapa anterior, detallando las entradas, las salidas y la descripción del proceso desde el punto de vista del usuario.
- Las descripciones de entradas y salidas deben ser en lo posible gráficas, o bien los documentos que se usan corrientemente.

Modelo IEEE/ANSI 830-1993 (12)

1.3.5. Requisitos de usuario y tecnológicos

- **Requisitos de usuario:** debe describirse el nivel de conocimiento de cada usuario (novato, intermedio, experto) para la realización de interfaces, manuales de usuario, ayuda y capacitación de los mismos.
- **Requisitos tecnológicos:** se describen las necesidades desde el punto de vista tecnológico, es decir equipos de clientes y servidores, velocidades de transmisión de datos, características que debe tener el sistema operativo y el sistema gestor de base de datos, y cualquier equipo que forma parte del sistema.
- Este documento es solamente un conjunto de criterios que me permite luego elegir el software y hardware adecuado para el sistema.



Modelo IEEE/ANSI 830-1993 (13)

1.3.6. Requisitos de interfaces externas

- En esta etapa se capturan los requerimientos que describen cómo debe ser la comunicación del sistema con el usuario y el mundo exterior.
- Se deben capturar las interfaces con el usuario, interfaces hardware, interfaces software e interfaces de comunicación.

1.3.7. Requisitos de rendimiento



- Pretende definir una serie de parámetros MENSURABLES del sistema que imponen restricciones sobre el mismo.
- Generalmente están asociados a tiempos de respuesta, tiempos de espera y duración de tareas batch.
- Estos requerimientos son muy importantes ya que la no-satisfacción de los mismos implica un fracaso del sistema, por lo que deben tener una prioridad alta.

Modelo IEEE/ANSI 830-1993 (14)

1.3.8. Requisitos de desarrollo y restricciones de diseño

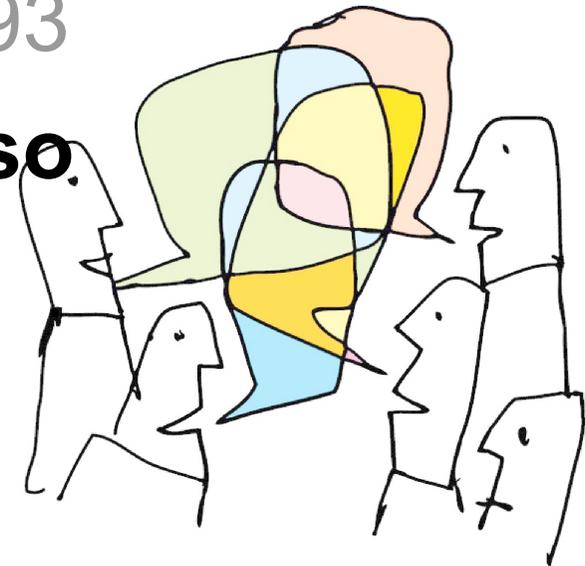
- **Requisitos de desarrollo:** se definen los requerimientos planteados por el equipo de trabajo: qué metodología se seguirá, qué ciclo de vida, qué herramientas se utilizarán, etc.
- **Restricciones de diseño:** son requisitos que nos impone la naturaleza del dominio del problema. Estos son:
 - Ajuste a estándares (p.e. una determinada manera de codificar un dato).
 - Limitaciones hardware (por los equipos disponibles).
 - Seguridad (por los distintos niveles de acceso a la información que deben tener los usuarios)
 - Mantenimiento (se debe tener en cuenta la ampliación del sistema)
 - Adaptación al entorno
 - Políticas de borrado.



Requerimientos y Casos de Uso

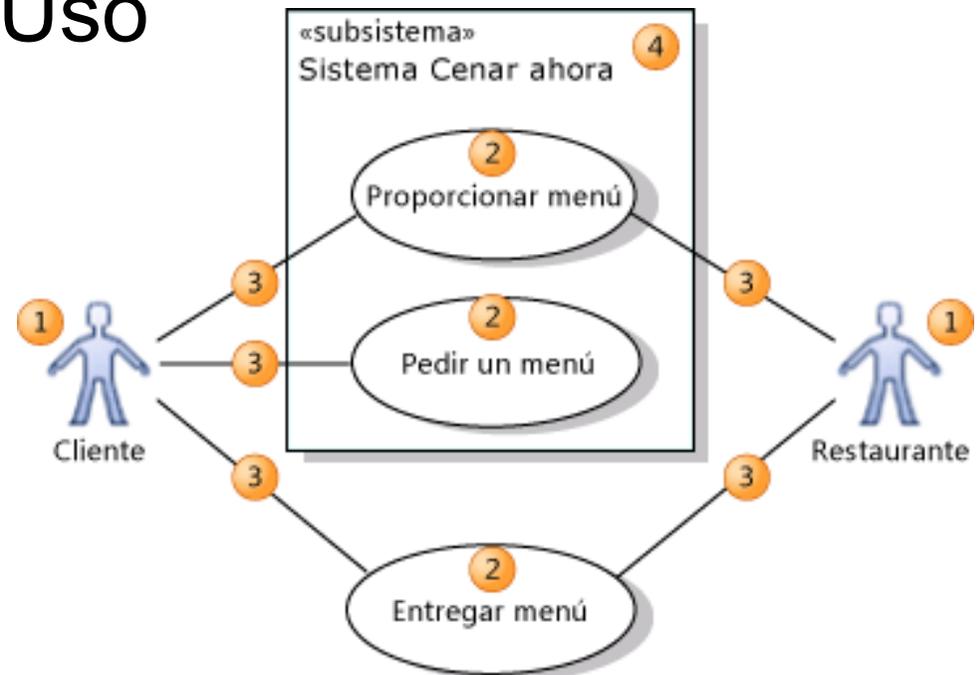
- Contenido

- Introducción
- Técnicas de Educación de Requerimientos
- Modelo IEEE/ANSI 830-1993
- **Modelado de Casos de Uso.**



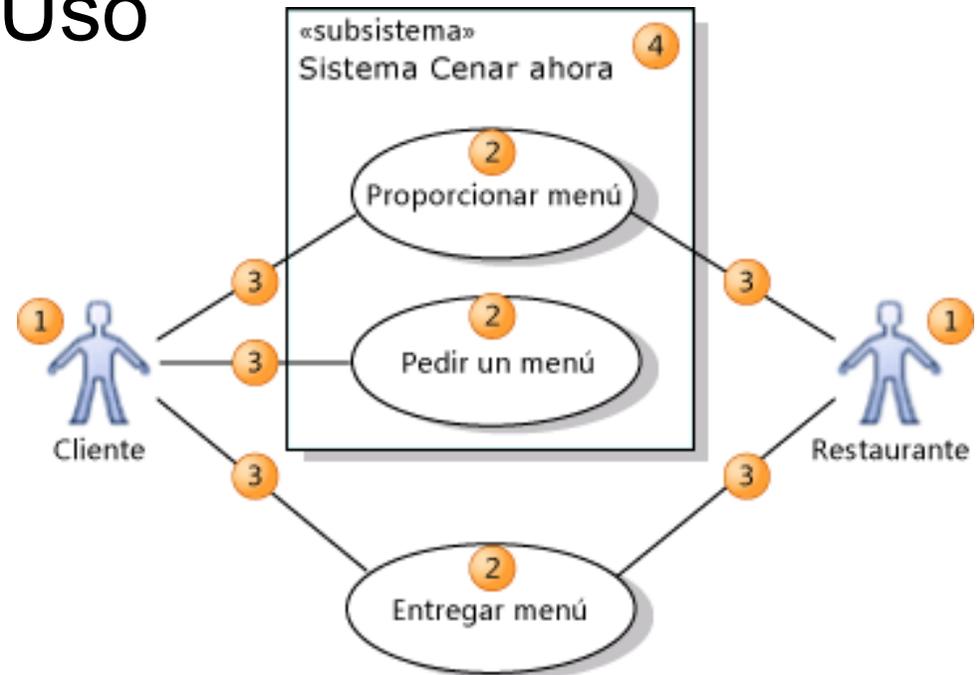
Modelado de Casos de Uso

- La Disciplina de Requisitos
- Vista de Casos de Uso
- Modelado de Casos de Uso
- Diagramas de Actividad



Modelado de Casos de Uso

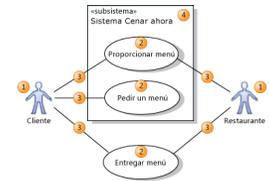
- **La Disciplina de Requisitos**
- Vista de Casos de Uso
- Modelado de Casos de Uso
- Diagramas de Actividad



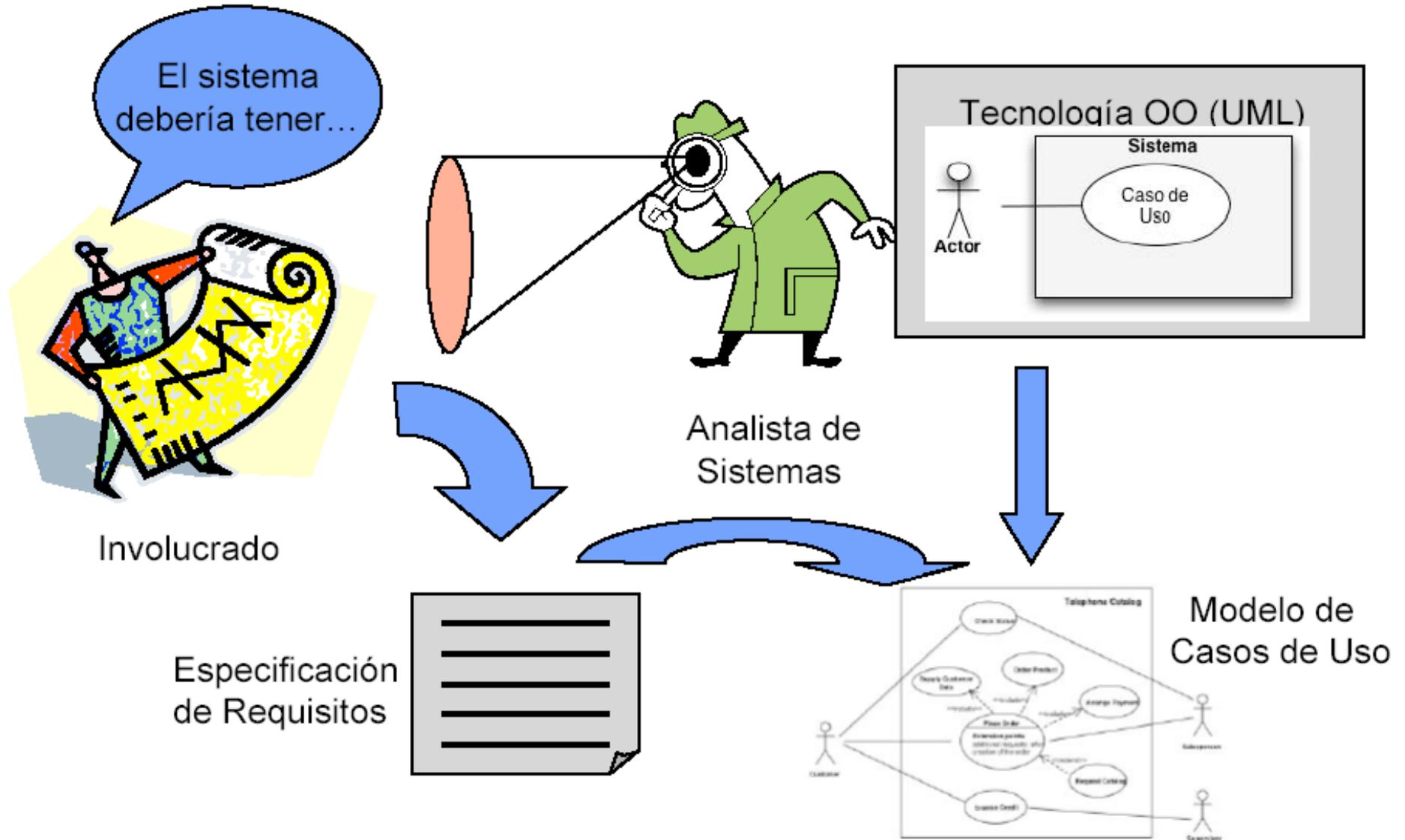
La Disciplina de Requisitos (1)

El propósito de esta disciplina es:

- Establecer y mantener un acuerdo con los involucrados respecto de lo que deberá hacer el sistema (plazos, precios).
- Suministrar a los desarrolladores del sistema una buena comprensión de los requisitos del sistema.
- Definir los límites o delimitar el sistema (alcance).
- Suministrar una base para planificar el contenido de las Iteraciones (Proceso Unificado).
- Suministrar una base para estimar el costo y el plazo de desarrollo del sistema (estimación).
- Definir una interfaz de usuario para el sistema, centrándose en las necesidades y objetivos de los usuarios (maqueta).

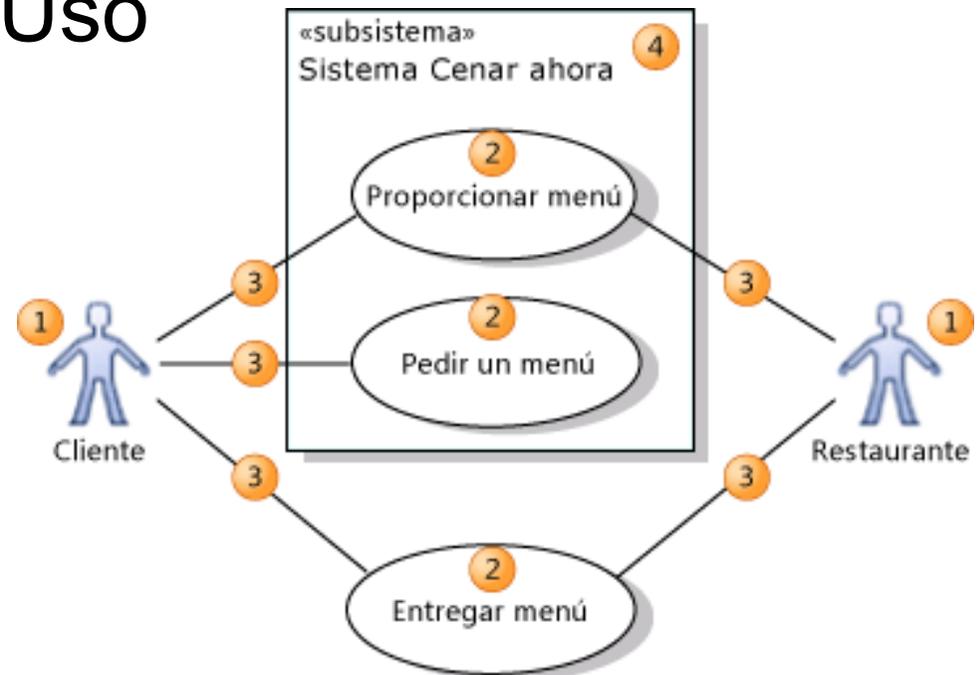


La Disciplina de Requisitos (2)



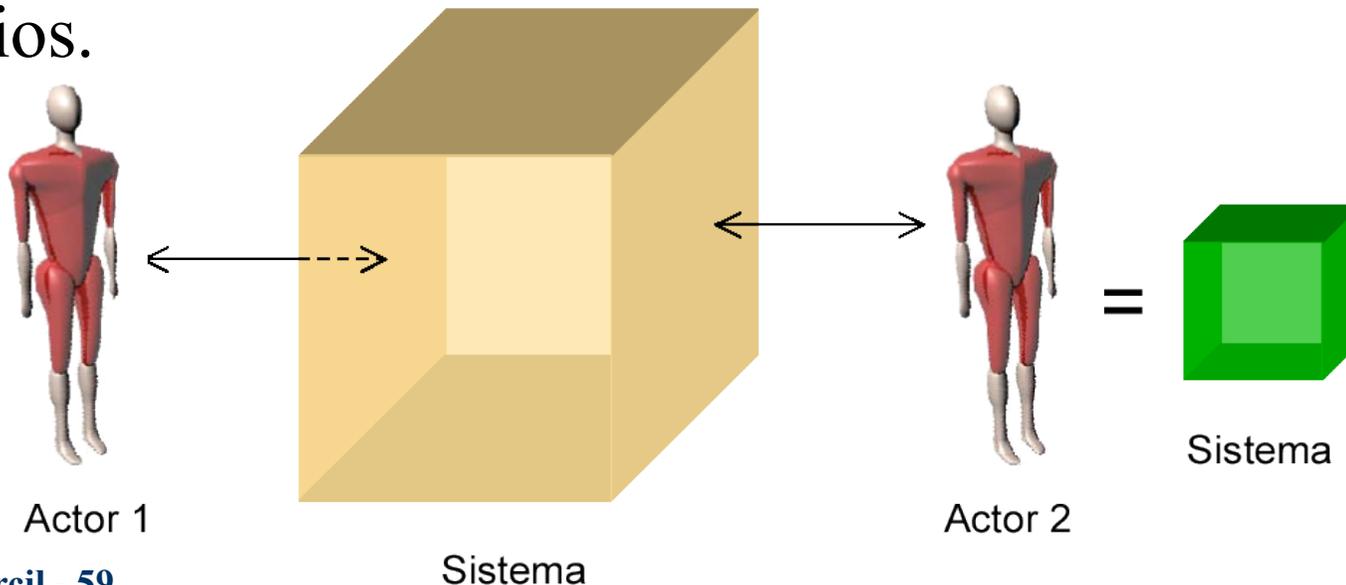
Modelado de Casos de Uso

- La Disciplina de Requisitos
- **Vista de Casos de Uso**
- Modelado de Casos de Uso
- Diagramas de Actividad



Vista de Casos de Uso (1)

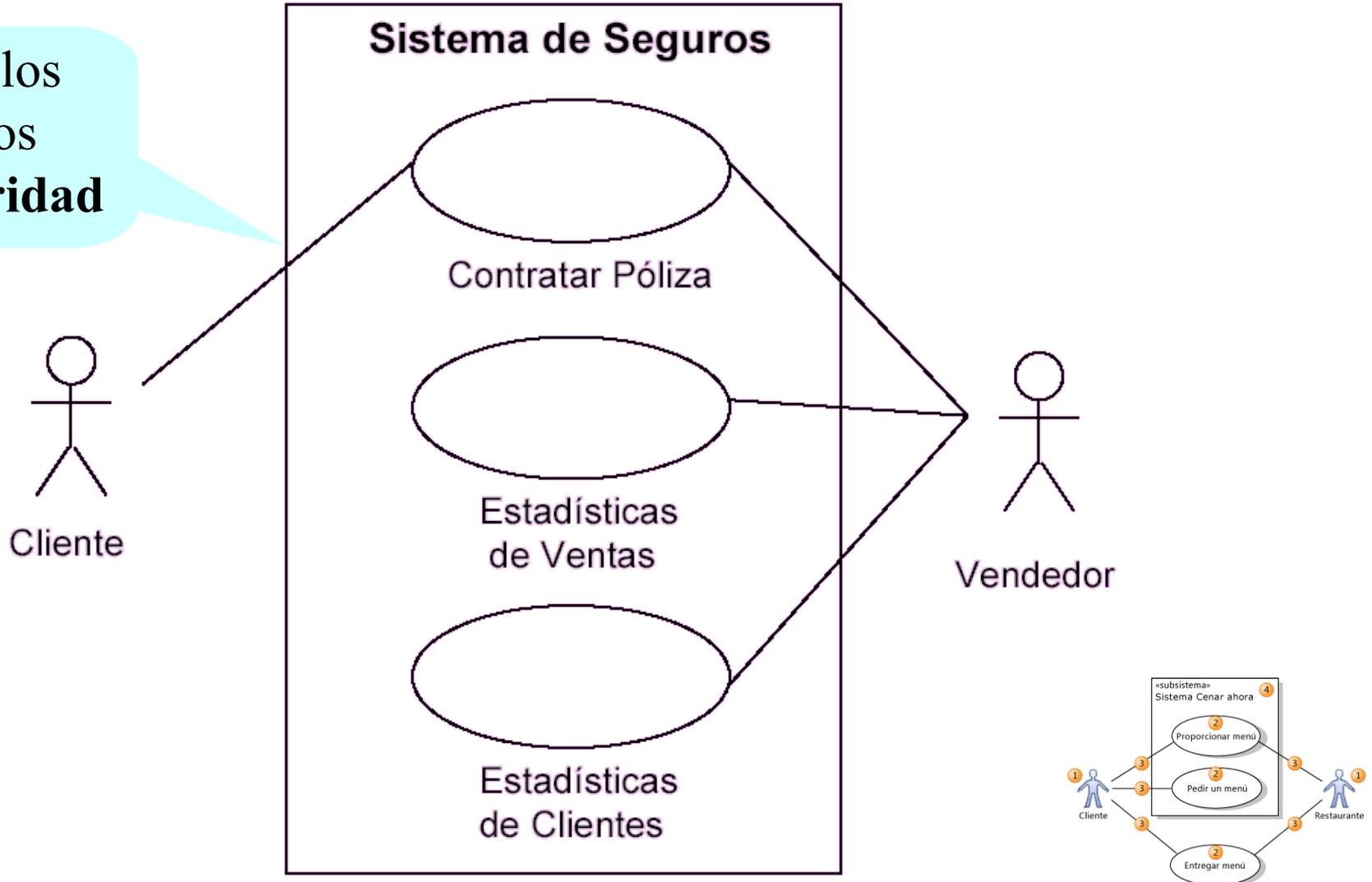
- Muestra la funcionalidad del sistema tal como la perciben los actores externos (caja negra).
- Un actor interactúa con el sistema, siendo el actor un usuario u otro sistema.
- Se describe en diagramas de casos de uso, que se usan además para **validar** el sistema probando la vista de casos de uso con los usuarios.



Vista de Casos de Uso (2)

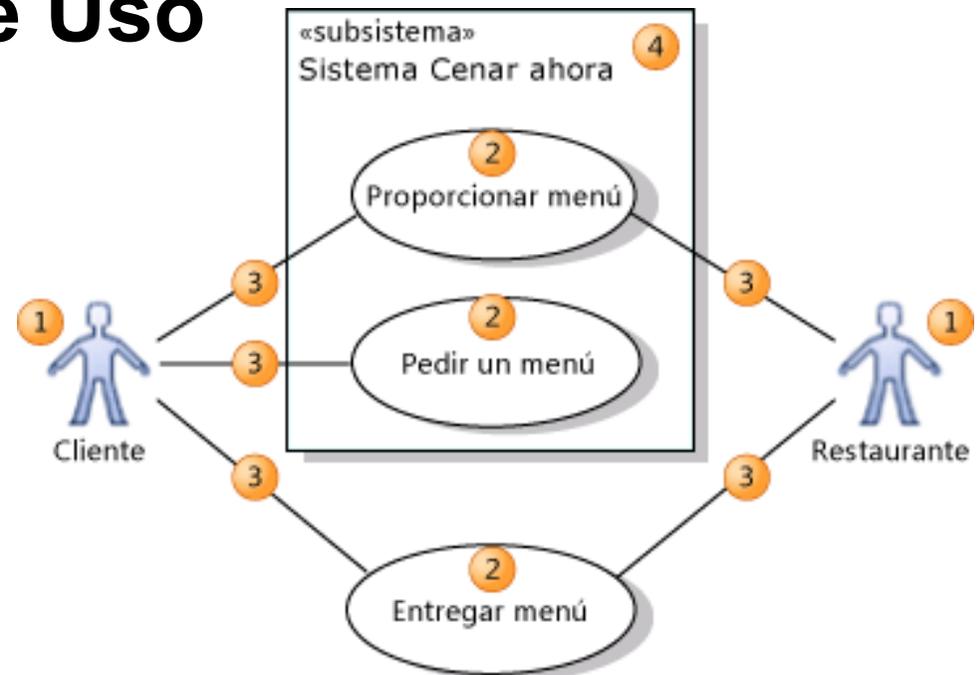
Un diagrama de casos de uso puede tener un aspecto similar al siguiente:

Dividir los dominios
Granularidad



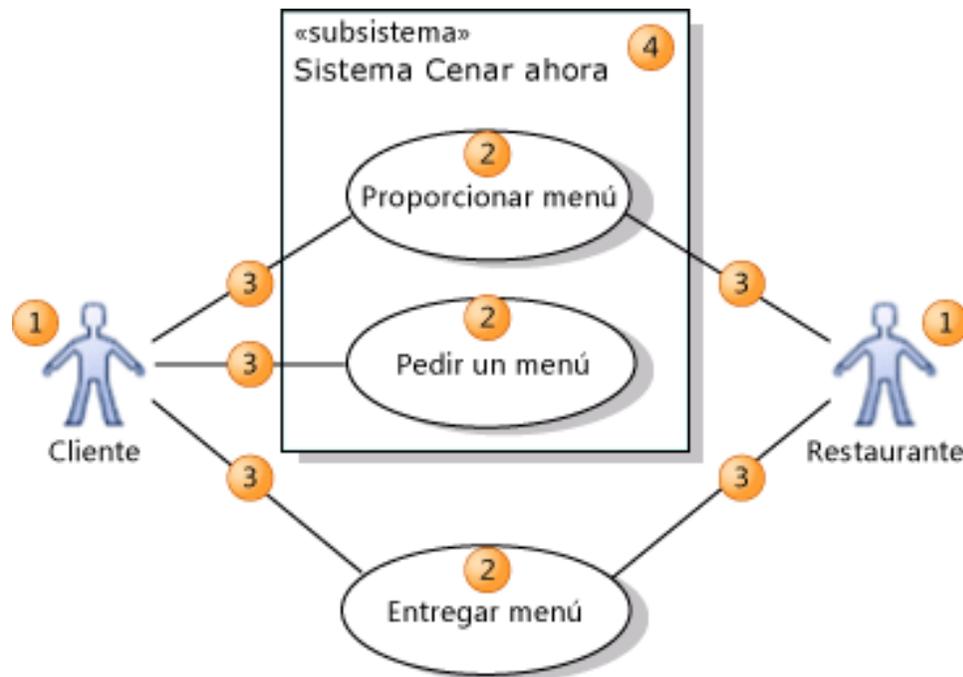
Modelado de Casos de Uso

- La Disciplina de Requisitos
- Vista de Casos de Uso
- **Modelado de Casos de Uso**
- Diagramas de Actividad



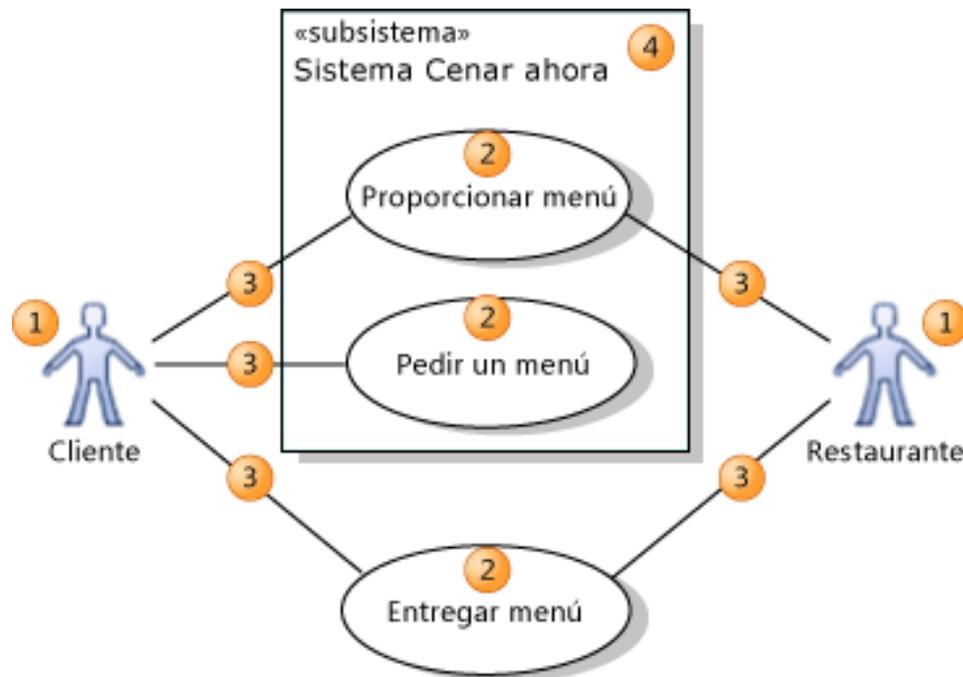
Modelado de Casos de Uso

- Diagramas de Casos de Uso
- Descripción Textual de los Casos de Uso



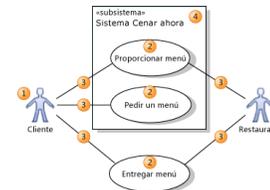
Modelado de Casos de Uso

- Diagramas de Casos de Uso
- Descripción Textual de los Casos de Uso



Diagramas de Casos de Uso (1)

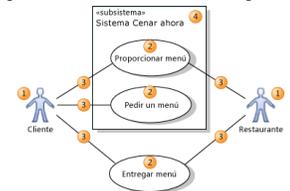
- En este tipo de modelado, el sistema es visualizado como una “caja negra” que suministra casos de uso. Por lo tanto no importa cómo lo hace el sistema, cómo están implementados los casos de uso o cómo funcionan internamente.
- Los propósitos más importantes de los casos de uso son:
 - Decidir y describir los requisitos funcionales del sistema, que serán el acuerdo entre los clientes y los desarrolladores del sistema.
 - Dar una descripción clara y consistente de lo que deberá hacer el sistema.
 - Dar una base para realizar las pruebas del sistema para verificar que funciona correctamente y validarlo.
 - Brindar la posibilidad de rastrear los requisitos funcionales hasta las clases reales y las operaciones.
 - Para simplificar los cambios y extensiones del sistema modificando el modelo de casos de uso y luego contrastando los casos de uso afectados con el diseño y la implantación del sistema.



Diagramas de Casos de Uso (2)

Ejemplo: Tarea de hacer una reserva de hotel.

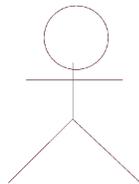
- Juan necesita hacer un viaje a Córdoba por una reunión de negocio de dos días. Viaja frecuentemente, normalmente varias veces por mes.
- En los últimos cuatro años ha ido a las mismas siete ciudades y ha creado una rutina.
- Tiene una línea aérea favorita, una cadena de hoteles y ubicaciones de hoteles en cada ciudad.
- Normalmente alquila un coche, pero si va a hacer mal tiempo prefiere un taxi.
- Prefiere hablar directamente con un agente de viajes y es muy exigente para que todo esté tal como lo ha pedido.



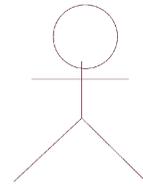
Diagramas de Casos de Uso (3)

Paso 1: Modelar los usuarios.

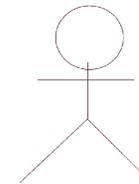
- Significa crear definiciones de grupos de usuarios cuya base de distinción se convierte en una diferenciación valiosa para los propósitos de DISEÑO.
- **Usuario:** persona que interactúa directamente con el sistema. El caso de uso debe generar valor agregado al usuario. Los clientes y los usuarios no son siempre los mismos.
- Al revisar las historias (escenarios) encontramos nombres de personas y sus características que nos dan pistas para ir definiendo actores como elementos del modelo de CU.



Turista



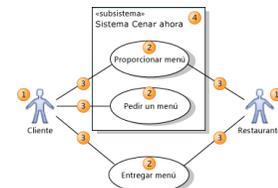
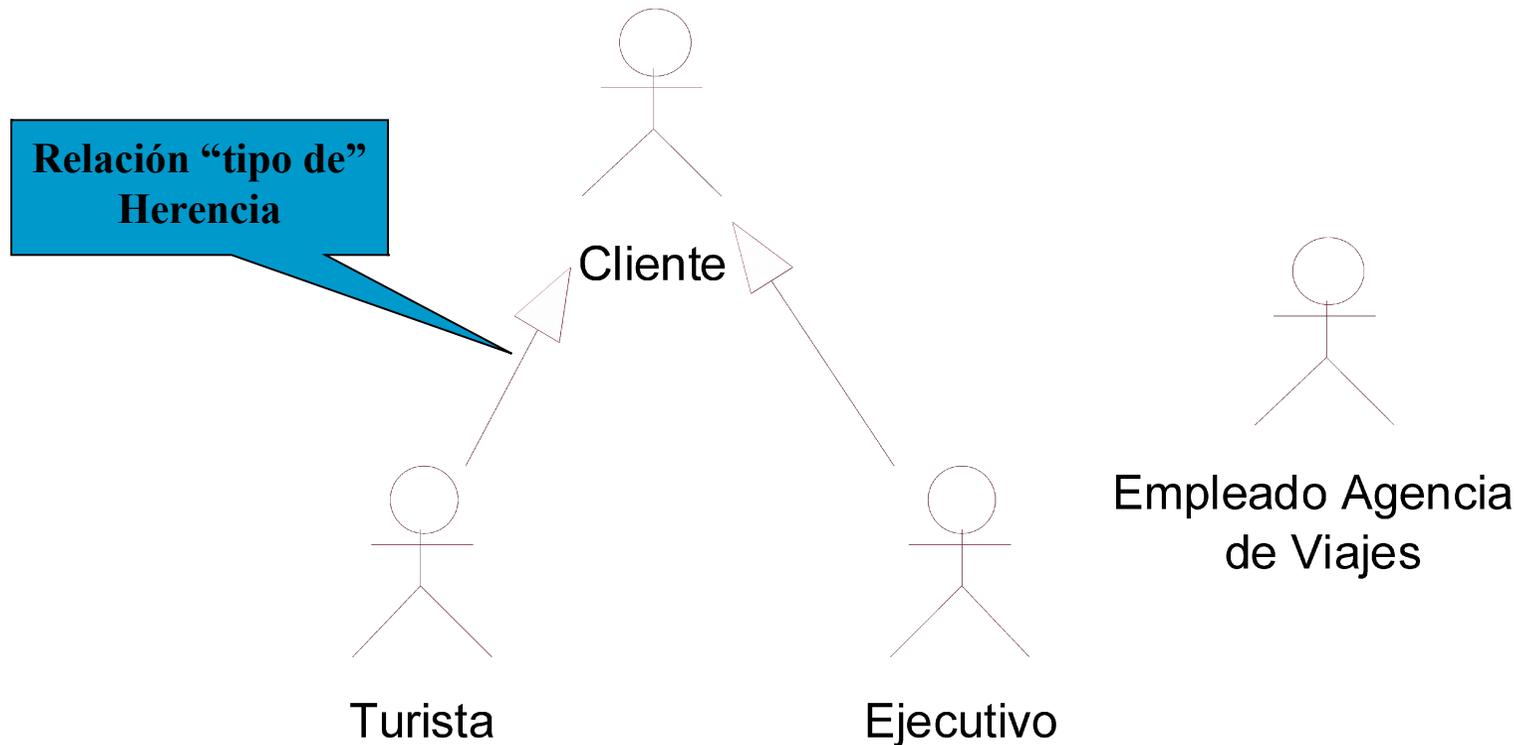
Ejecutivo



Empleado Agencia
de Viajes

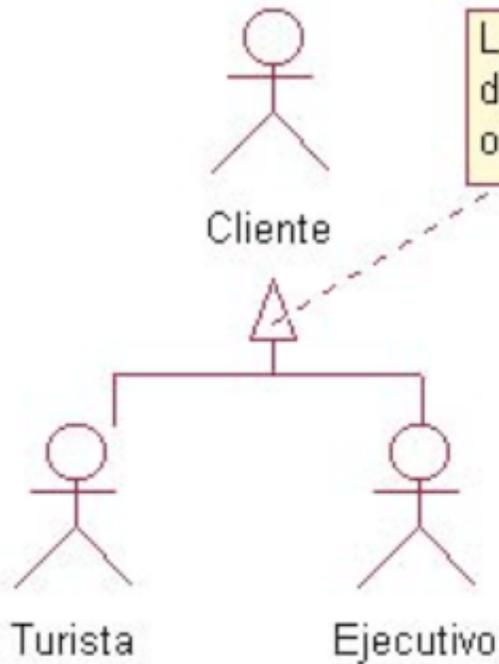
Diagramas de Casos de Uso (4)

- Al obtener información adicional, podemos encontrar actores que tengan algunas características en común y podemos relacionarlos a través de una generalización/especialización.

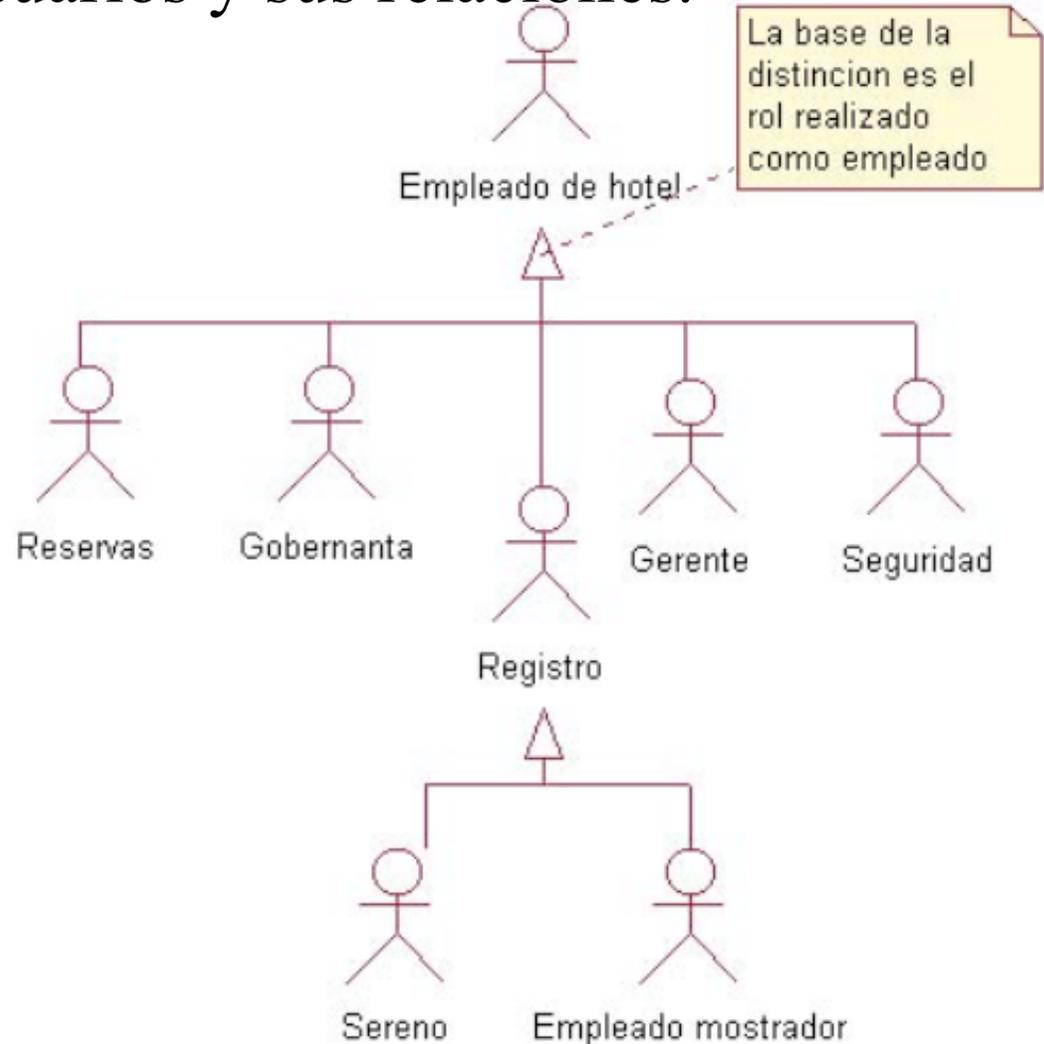


Diagramas de Casos de Uso (5)

- Puede haber otro tipo de usuarios y sus relaciones.



La base de la distinción es el objetivo del viaje



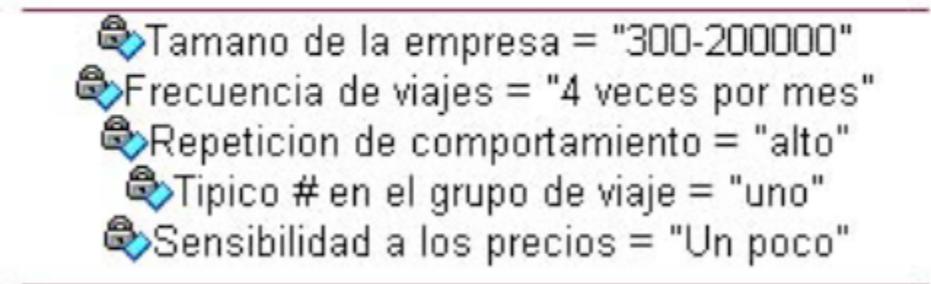
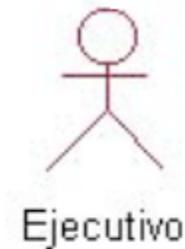
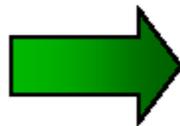
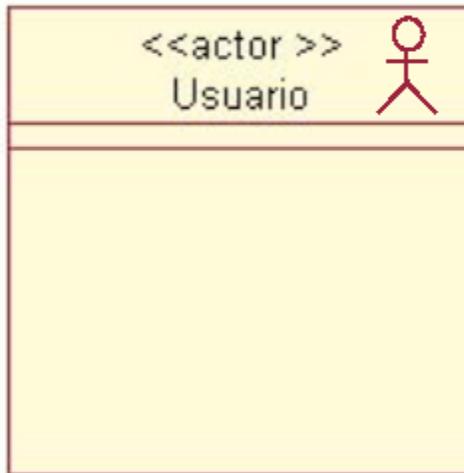
La base de la distinción es el rol realizado como empleado



Diagramas de Casos de Uso (6)

- Diagrama de clase de usuarios: Descripción detallada de cada clase, puesto que un actor no es sino una clase “estereotipada”.
- Un estereotipo es un mecanismo de extensión de UML que permite definir un nuevo tipo de elemento de modelo basado en otro existente.

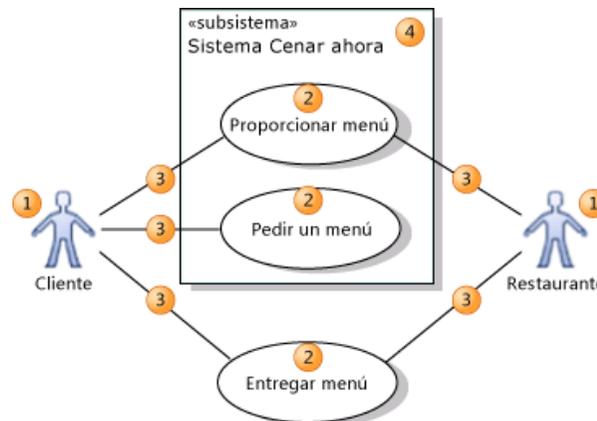
Un actor es una clase.
Las operaciones no
tienen sentido en actores.



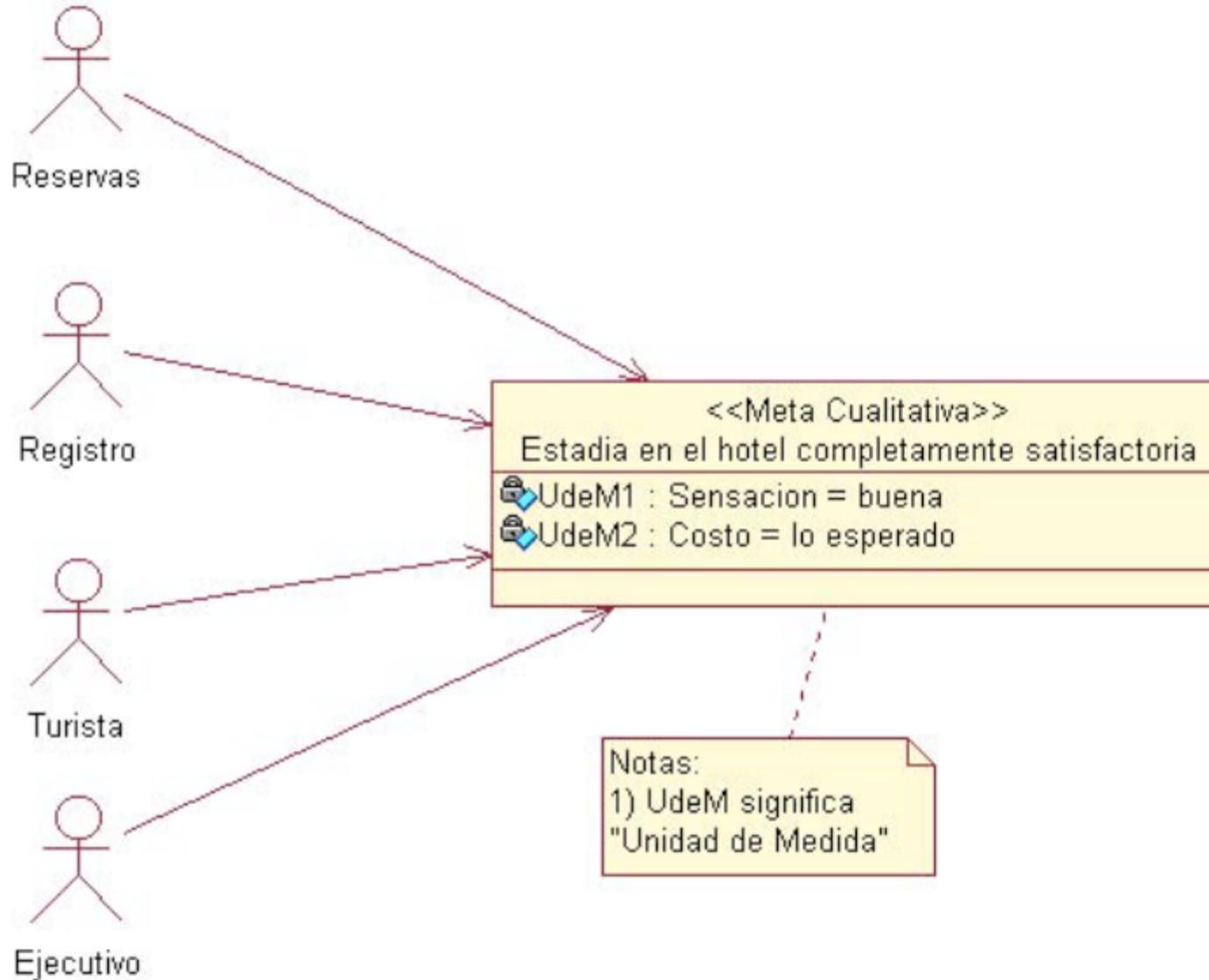
Diagramas de Casos de Uso (7)

Paso 2: Modelar las metas de los usuarios.

- Definir quién tiene qué metas a efectos de validar las distinciones entre grupos de usuarios y ubicar las tareas en el contexto de las metas.
- **Meta:** un fin que debe lograrse independientemente de cómo se obtenga.
- Criterio de éxito de la meta: es importante no sólo definir las metas sino los criterios de éxito para lograr cada meta.

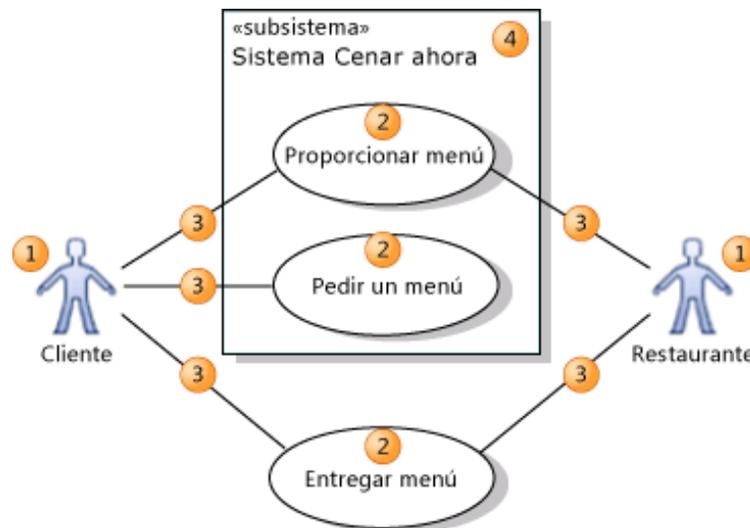


Diagramas de Casos de Uso (8)



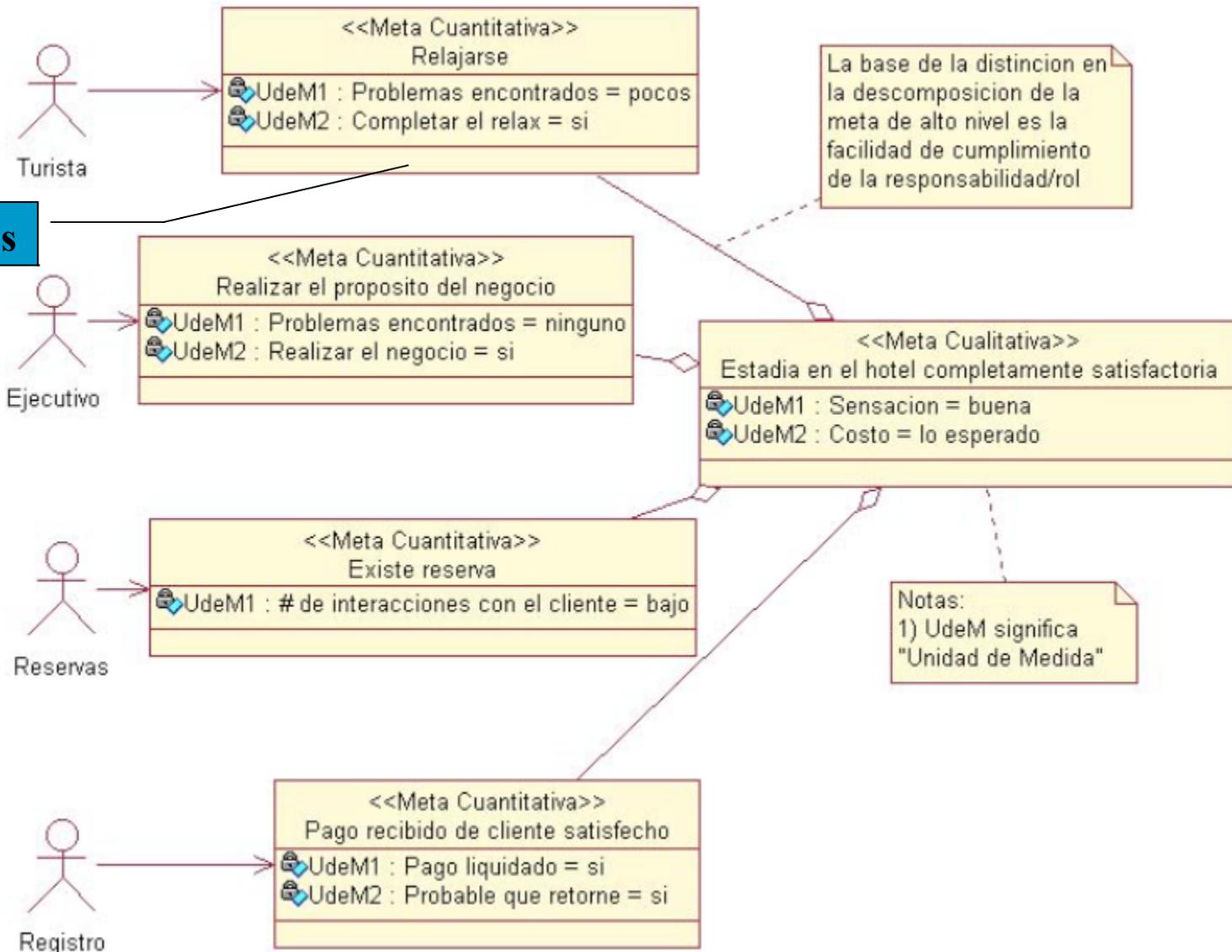
Diagramas de Casos de Uso (9)

- En el diagrama anterior tenemos un problema: no resulta de mucha ayuda para el diseño el hecho de que todos los actores tengan la misma meta.
- Por lo tanto debemos fijarnos si hay sub-metas que sean una base de distinción entre los grupos de usuarios y que definan porqué los usuarios querrán usar el sistema para realizar determinadas tareas.



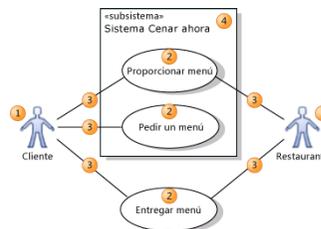
Diagramas de Casos de Uso (10)

Estereotipos



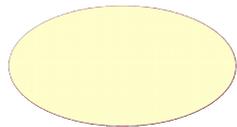
Diagramas de Casos de Uso (11)

- Un caso de uso es una secuencia de acciones realizadas por un actor y el sistema que deriva en un resultado (o conjunto de resultados) de valor para uno o varios actores.
- El texto de un caso de uso describe caminos posibles a través del caso de uso. Se asocian dos clases de flujos de eventos dentro de los casos de uso:
 - El flujo principal de eventos (a veces llamado el curso básico de acción), que es el camino de principio a fin que el actor y el sistema siguen en condiciones normales (escenario principal).
 - Un flujo de eventos de excepción (o curso de acción alternativo), que es el camino a través del caso de uso que representa una condición de error o camino tomado menos frecuentemente (escenario secundario).

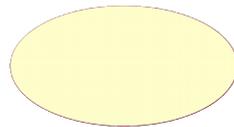


Diagramas de Casos de Uso (12)

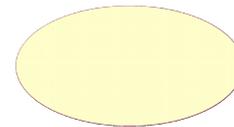
- Las siguientes orientaciones son útiles para producir casos de uso sólidos y fáciles de comprender en una gran variedad de contextos:
 - Usar una voz activa y hablar desde la perspectiva del actor:** evitar (aunque es menos frecuente en castellano) expresiones del tipo “la opción es seleccionada por el usuario”. Mejor es “el usuario selecciona la opción”.
 - Usar el presente:** es común oír expresiones en futuro, del tipo “el sistema hará esto o aquello”. El caso de uso estará expresado en presente: “el usuario selecciona el ítem”, “el sistema efectúa la conexión”.



Hacer una reserva



Entrada



Salida

Diagramas de Casos de Uso (13)

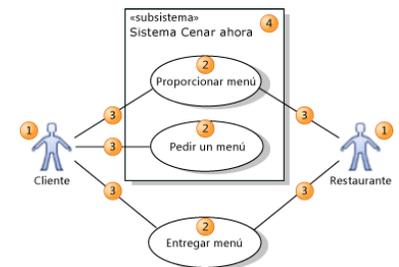
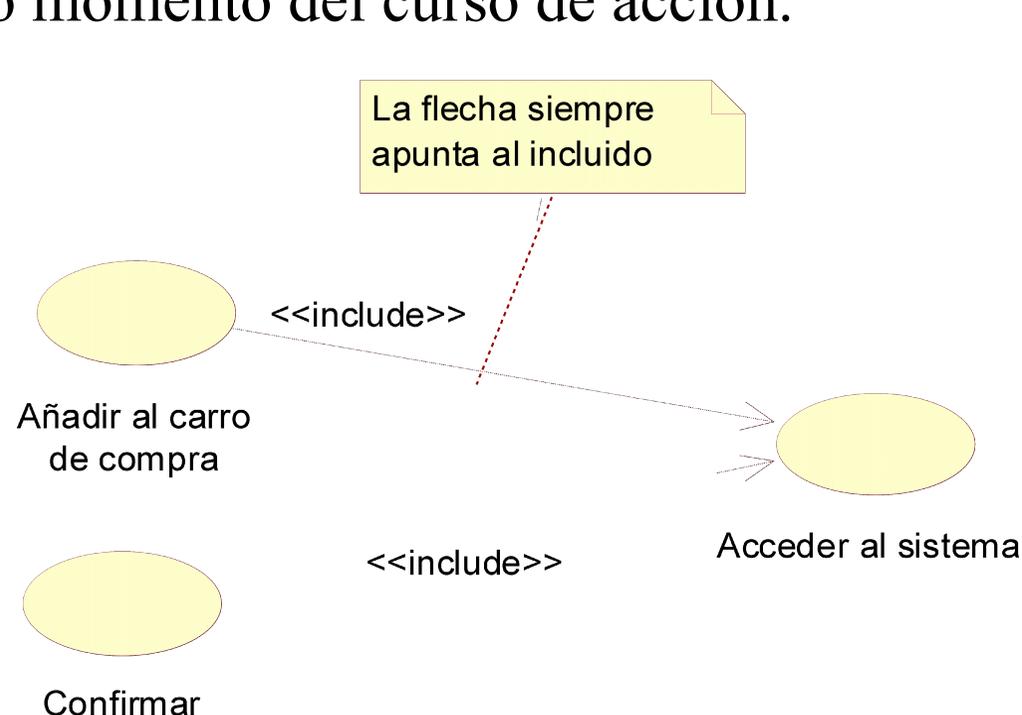
- ***Expresar el texto en la forma de “llamada y respuesta”***: el esquema básico será del tipo: “el actor hace esto”, “el sistema hace lo otro”. Tanto el actor como el sistema podrán hacer más de una cosa consecutivamente, pero el texto deberá reflejar el hecho de que el actor realiza una acción y el sistema responde de acuerdo a ello.
- ***Escribir el texto en no más de tres párrafos***: de la misma manera que uno de los principios de la OO debe hacer (bien) un pequeño número de cosas, los casos de uso también deberían ajustarse a este principio.
- ***Nombrar las clases***: hay dos tipos básicos de clases que se prestan a ser incluidas en el texto de los casos de uso:
 1. Las del modelo del dominio.
 2. Las clases límite (o frontera) que abarcan las ventanas, páginas HTML, etc. que los usuarios utilizan cuando interactúan con el sistema.
 - Será más fácil diseñar con textos del tipo “el cliente cambia una o más cantidades en Editar Contenidos de la página Carro de Compra” o “el sistema crea una cuenta para el Cliente”.

Diagramas de Casos de Uso (14)

- ***Establecer el contexto inicial:*** tenemos que especificar dónde está el actor y lo que pretende al comienzo del caso de uso. Hay dos maneras de hacerlo:
 - La primera implica especificar el contexto como parte de la primera oración: “El contable introduce su ID de usuario y palabra clave en la ventana de acceso al sistema”.
 - La segunda manera implica definir una **precondición**. “El contable ha accedido al sistema”. Muchas veces se exige que se haya ejecutado como CU.
- ***Asegurarse de que cada caso de uso produce al menos un resultado de valor para uno o más actores, incluso si el resultado es negativo:*** normalmente ocurre algo positivo: “El actor accedió al sistema”, “el sistema completa la tarea del usuario actualizando la base de datos”, “el sistema genera un informe”. Pero puede ocurrir que ejecute un curso alternativo de acción: “el sistema bloqueó al usuario y envió un mail al administrador”.
- ***Ser exhaustivo en describir cursos alternativos de acción:*** una manera efectiva es cuestionar cada oración del curso básico con preguntas de tipo “¿cómo puede actuar el usuario de forma diferente o errónea?”.

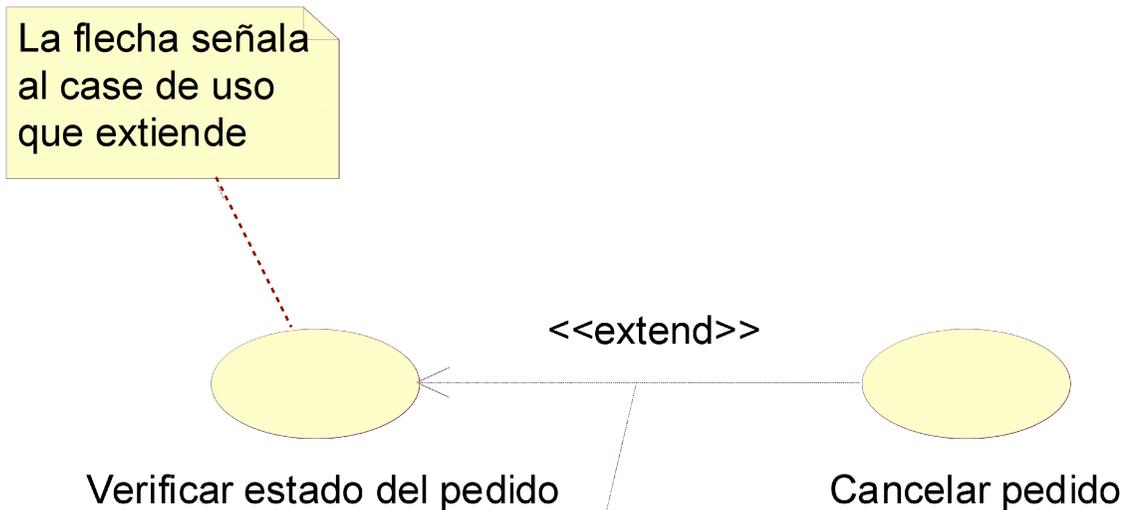
Diagramas de Casos de Uso (15)

- UML ofrece tres estructuras para construir el comportamiento común y los caminos alternativos de los casos de uso:
- Incluye (Include):** es una relación por la cual un caso de uso explícitamente incluye el comportamiento de otro en un determinado momento del curso de acción.

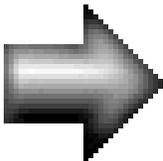
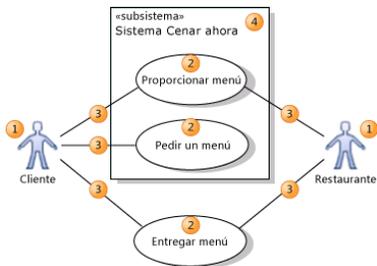


Diagramas de Casos de Uso (16)

- **Extiende (Extend):** es una relación por la cual un caso de uso base implícitamente incluye el comportamiento de otro en uno o más puntos especificados. Estos puntos se denominan puntos de extensión.

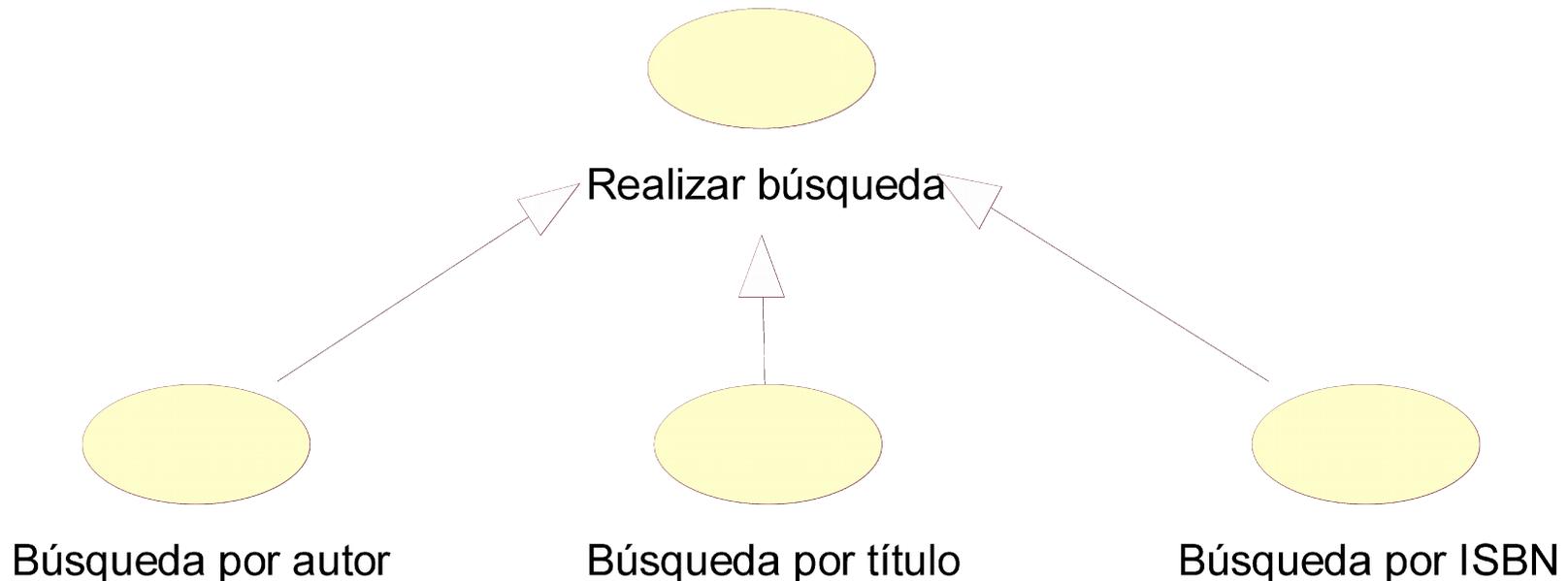


Cuando el sistema muestra el ID del pedido, el cliente tiene varias opciones, incluida la posibilidad de cancelarlo



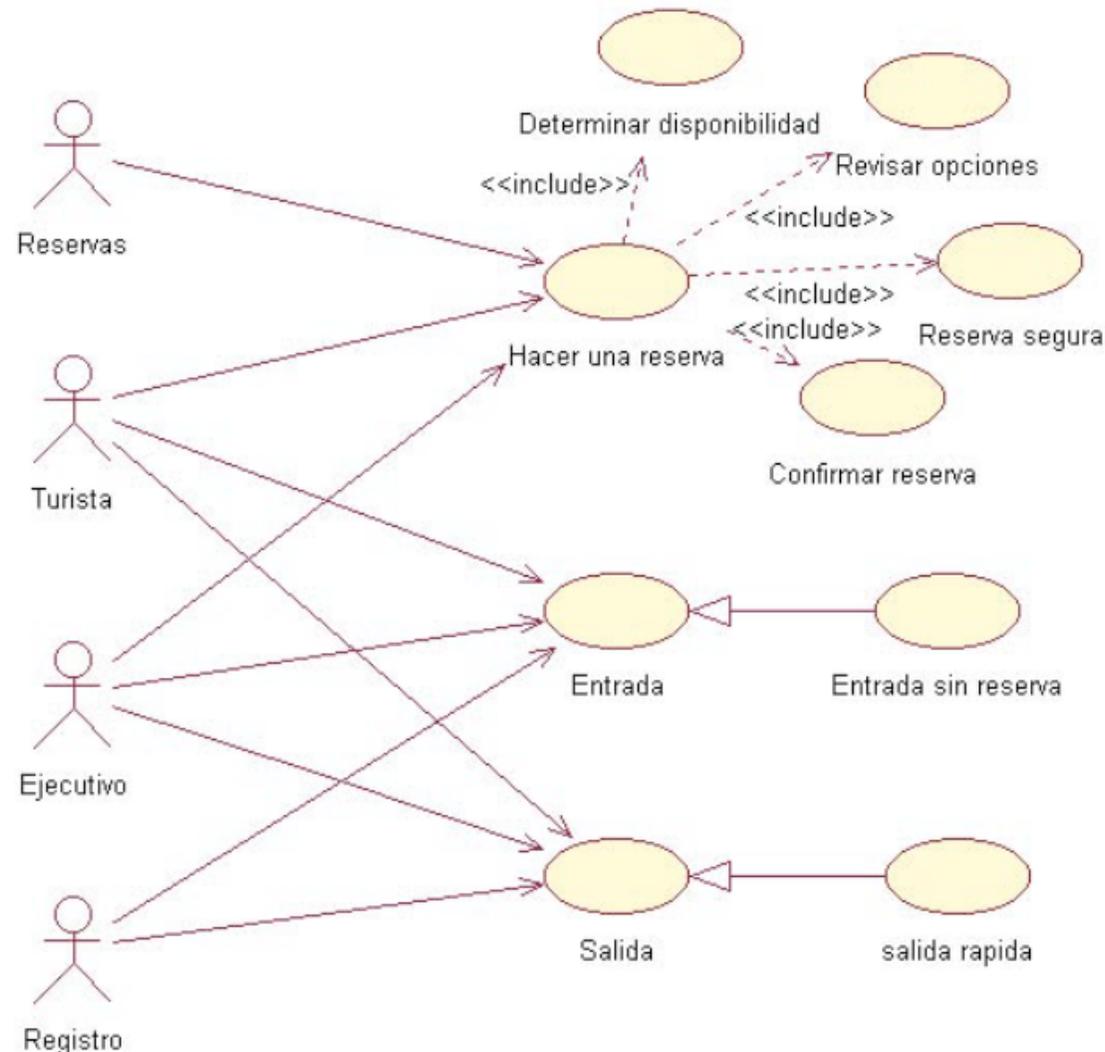
Diagramas de Casos de Uso (17)

- **Generalización:** funciona de la misma manera para casos de uso que para clases en general: un caso de uso padre define un comportamiento que los hijos pueden heredar, añadiendo o ignorando este comportamiento.



Diagramas de Casos de Uso (18)

Los casos de uso para nuestro ejemplo del hotel serán:



Diagramas de Casos de Uso (19)

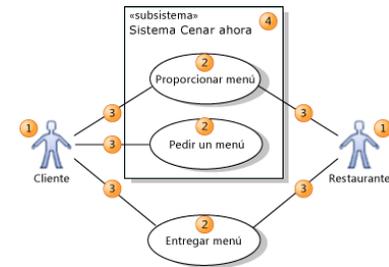
Ejemplo

Se trata de un sistema simplificado de Cajero Automático (CA) que ofrece los siguientes servicios:

- Distribución de dinero a cada poseedor de una tarjeta inteligente a través de un lector de tarjetas y un distribuidor de efectivo.
- Consulta del saldo de cuenta, facilidades para depósito de efectivo y cheques para los clientes del banco poseedor de una tarjeta del mismo.

Tener en cuenta además:

- Todas las transacciones deben ser seguras.
- A veces es necesario rellenar el distribuidor de billetes, etc.

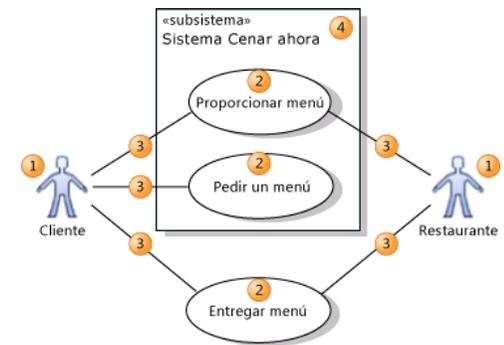


Diagramas de Casos de Uso (20)

Ejemplo

A partir de esta descripción, habrá que realizar las siguientes tareas:

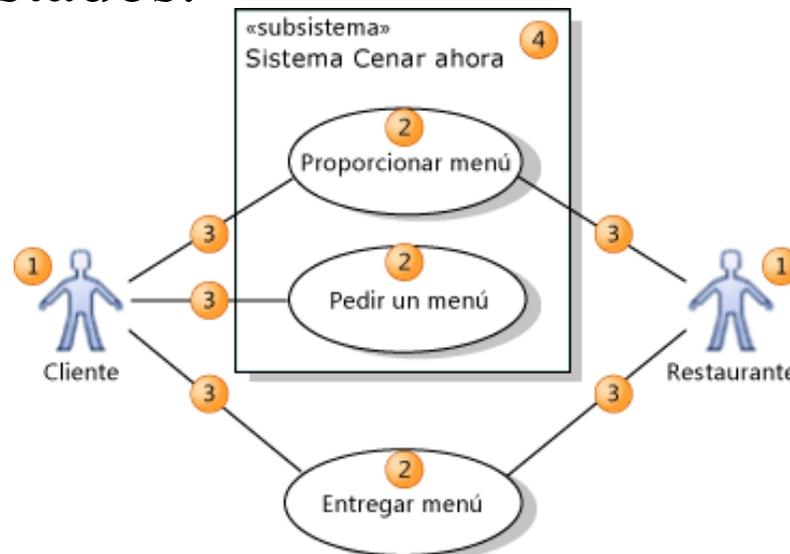
- Identificar los actores
- Identificar los casos de uso
- Construir un diagrama de casos de uso
- Escribir una descripción textual de casos de uso
- Organizar y estructurar los casos de uso



La descripción anterior es (de forma deliberada) incompleta e imprecisa, tal como aparecen los casos reales.

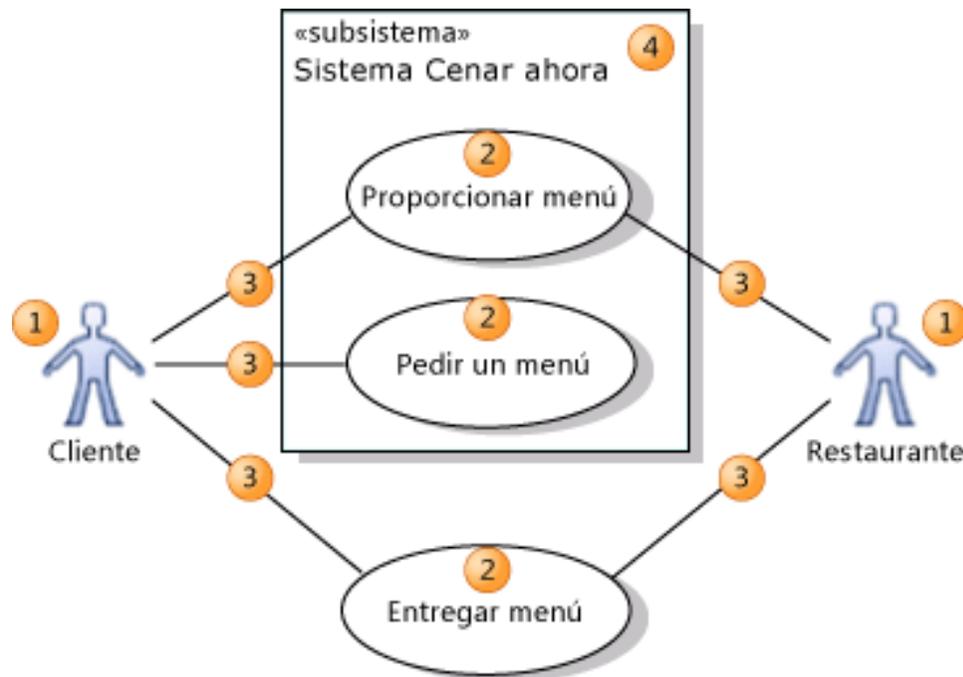
Diagramas de Casos de Uso (21)

- Hemos modelado lo que hemos descubierto sobre:
 - Usuarios
 - Metas
 - Tareas
- Hemos declarado explícitamente los casos de uso.
- Ahora debemos ver las clases involucradas con sus atributos, operaciones y estados.



Modelado de Casos de Uso

- Diagramas de Casos de Uso
- **Descripción Textual de los Casos de Uso**



Descripción Textual de CU (1)

- La descripción textual de los casos de uso no está estandarizada por UML. Sólo existen orientaciones y recomendaciones de algunos actores (Larman, <https://use-cases.org/>)
- Resumen de identificación (obligatorio): incluye el título, un resumen, fechas de creación y modificación, versión, persona a cargo, actores, etc.
- Flujo de eventos (obligatorio): describe el escenario principal, las secuencias alternativas y de error además de las pre-condiciones y post-condiciones.
- Requisitos de IU (interfaz de usuario): se pueden añadir restricciones de IU (la apariencia y sensación) resultando muy útil las copias de pantallas o una maqueta desechable.

Descripción Textual de CU (2)

Resumen de Identificación:

Título: Retirar dinero utilizando tarjeta Visa.

Resumen: este caso de uso permite al poseedor de una tarjeta Visa, que no es el cliente de un banco, retirar dinero si su límite diario se lo permite.

Actores: PoseedorTarjetaVisa (primario), Sistema de Autorización de Visa (secundario)

Fecha de Creación: 29/08/2004 **Fecha de Actualización:** 13/10/2004 **Versión:** 2.2

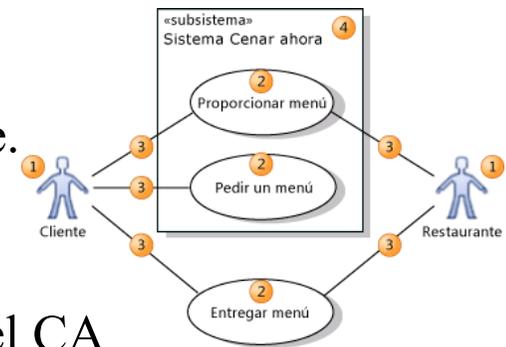
Flujo de Eventos:

Precondiciones:

- El alimentador de billetes del CA tiene una cantidad suficiente.
- No hay una tarjeta en el lector

Escenario principal:

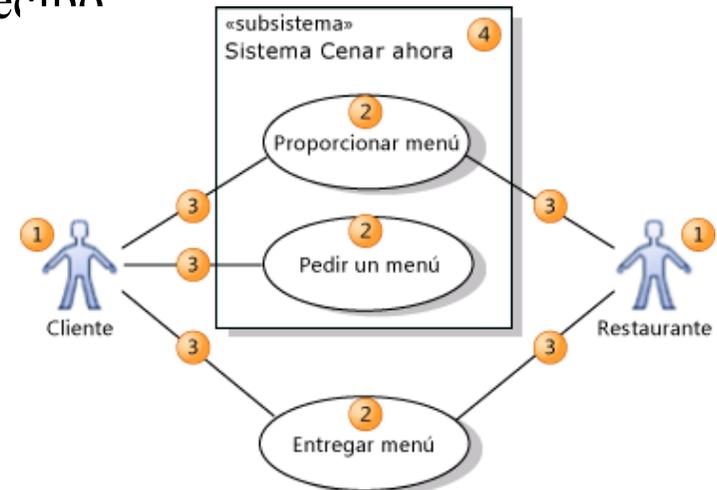
1. El PoseedorTarjetaVisa inserta su tarjeta en el lector de tarjetas del CA
2. El CA verifica que la tarjeta es de un tipo reconocible.
3. El CA pide al PoseedorTarjetaVisa que introduzca su código personal.
4. El PoseedorTarjetaVisa introduce su código personal.
5. El CA compara el código personal con el que está codificado en el chip de la tarjeta.



Descripción Textual de CU (3)

Escenario principal:

6. El CA solicita una autorización al Sistema de Autorización de Visa
7. El Sistema de Autorización de Visa confirma la autorización y comunica el límite diario de extracción.
8. El CA pide al PoseedorTarjetaVisa que indique la cantidad deseada.
9. El PoseedorTarjetaVisa introduce la cantidad deseada.
10. El CA compara la cantidad deseada respecto del límite diario disponible.
11. El CA pregunta al PoseedorTarjetaVisa si desea un recibo
12. El PoseedorTarjetaVisa pide un recibo.
13. El CA devuelve la tarjeta al PoseedorTarjetaVisa.
14. El PoseedorTarjetaVisa retira su tarjeta.
15. El CA emite los billetes e imprime un recibo.
16. El PoseedorTarjetaVisa recoge los billetes y el recibo.



Descripción Textual de CU (4)

Secuencias Alternativas:

A1: Código personal temporalmente incorrecto

La secuencia A1 comienza en el punto 5 del escenario principal

6. El CA informa al PoseedorTarjetaVisa que el código personal es incorrecto durante la primera o segunda vez.

7. El CA graba el fallo en la tarjeta.

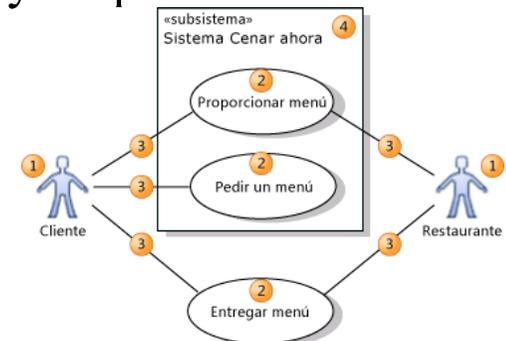
El escenario vuelve al punto 3.

A2. La cantidad solicitada es mayor que el límite diario disponible.

La secuencia A2 comienza en el punto 10 del escenario principal

11. El CA informa al PoseedorTarjetaVisa que la cantidad es mayor que el límite diario disponible.

El escenario vuelve al punto 8.



Descripción Textual de CU (5)

Secuencias Alternativas:

A3: El PoseedorTarjetaVisa no quiere un recibo

La secuencia A3 comienza en el punto 11 del escenario principal

12. El PoseedorTarjetaVisa indica que no quiere un recibo.

13. El CA devuelve la tarjeta al PoseedorTarjetaVisa

14. El PoseedorTarjetaVisa recoge su tarjeta.

15. El CA emite los billetes.

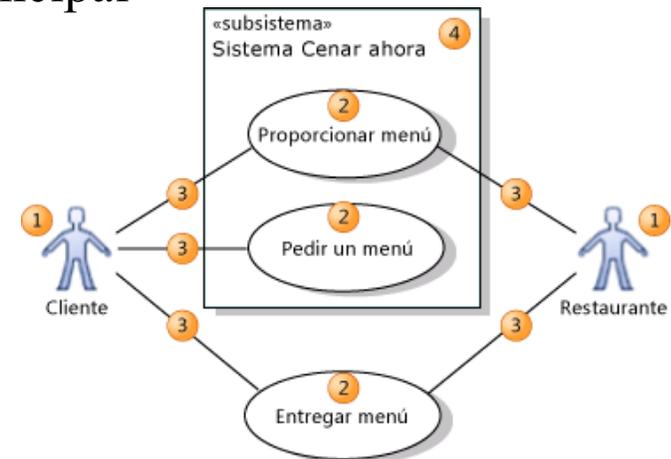
16. El PoseedorTarjetaVisa recoge los billetes.

Secuencias de error:

E1: tarjeta inválida

La secuencia E1 comienza en el punto 2 del escenario principal.

3. El CA informa al PoseedorTarjetaVisa que la tarjeta no es válida (ilegible, expirada, etc.) y la retiene; el caso de uso falla.



Descripción Textual de CU (6)

Secuencias de error:

E2: código personal incorrecto definitivamente

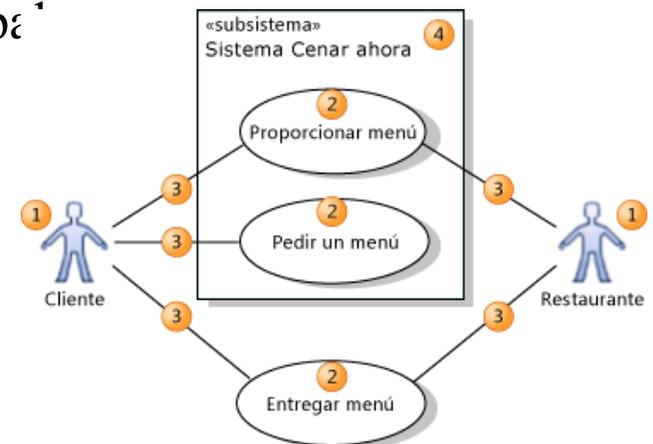
La secuencia E2 comienza en el punto 5 del escenario principal.

6. El CA informa al PoseedorTarjetaVisa que el código personal es incorrecto por tercera vez.
7. El CA retiene la tarjeta.
8. Se notifica al Sistema de Autorización de Visa; el caso de uso falla.

E3: extracción no autorizada

La secuencia E3 comienza en el punto 6 del escenario principal.

7. El SA Visa desautoriza cualquier extracción.
8. El CA devuelve la tarjeta; el caso de uso falla.

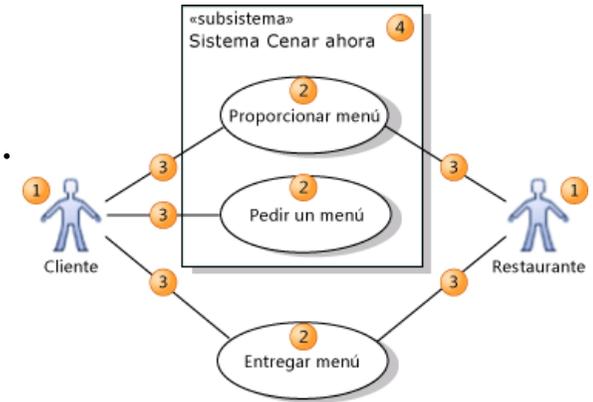


Descripción Textual de CU (7)

Secuencias de error:

E4: la tarjeta no es recogida por el usuario

La secuencia E4 comienza en el punto 13 del escenario principal.
14. Después de 15 segundos, el CA retiene la tarjeta.
15. Se informa al SA Visa; el caso de uso falla.



E5: el usuario no recoge los billetes emitidos

La secuencia E5 comienza en el punto 15 del escenario principal
16. Después de 30 segundos, el CA vuelve a recoger los billetes.
17. Se informa al SA Visa; el caso de uso falla.

Post-Condiciones:

El alimentador de billetes del CA contiene menos billetes que al comienzo del caso de uso
(el número de billetes entregados depende de las cantidades solicitadas)

Descripción Textual de CU (8)

Requisitos de interfaz de usuario:

Los mecanismos de entrada/salida disponibles al PoseedorTarjetaVisa serán:

- Un lector de tarjetas.
- Un teclado numérico (para introducir el código personal u otras cantidades) con las teclas “introducir”, “correcto” y “cancelar”.
- Una pantalla para mostrar cualquier mensaje del CA.
- Teclas alrededor de la pantalla de tal manera que el poseedor de la tarjeta pueda seleccionar una cantidad deseada de las ofrecidas en pantalla.
- Un distribuidor de billetes.
- Un emisor de recibos.



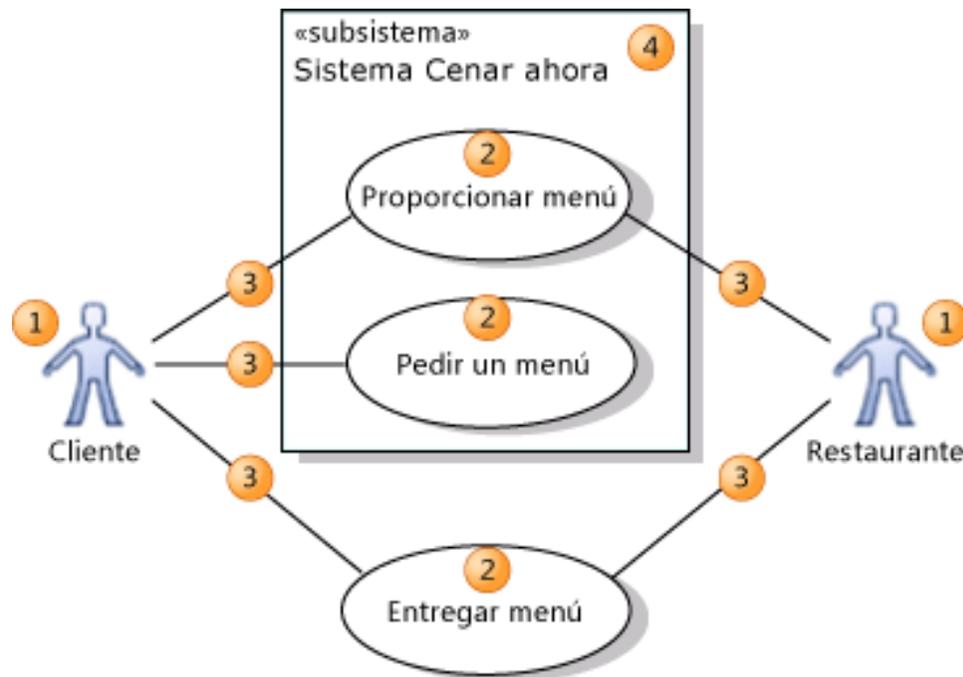
Descripción Textual de CU (9)

Restricciones no funcionales:

Restricción	Especificaciones
Tiempo de Respuesta	La interfaz del CA debe responder dentro de un tiempo máximo de 2 segundos. Una transacción de extracción debe requerir menos de 2 minutos.
Concurrencia	No se aplica (usuario único)
Disponibilidad	El CA debe poder accederse 24/7. La falta de papel para imprimir recibos no debe impedir que el poseedor de la tarjeta pueda realizar su extracción.
Integridad	Las interfases del CA deben ser extremadamente robustas como para evitar vandalismos.
Confidencialidad	El procedimiento de comparación entre el código personal introducido por teclado y el de la tarjeta debe tener un máximo de fallos de 10^{-6} .

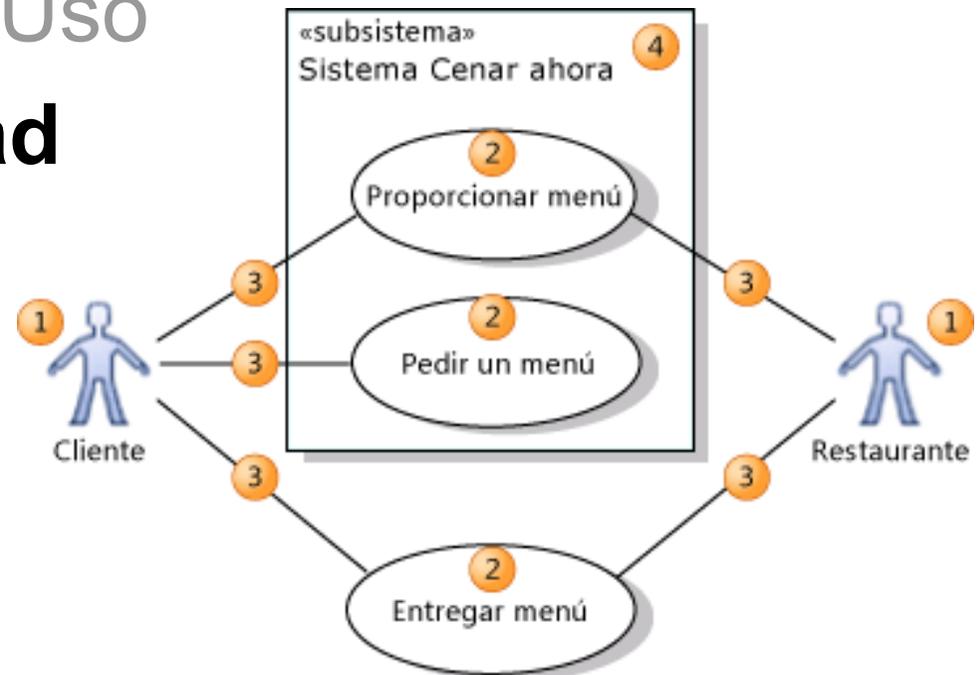
Modelado de Casos de Uso

- Diagramas de Casos de Uso
- Descripción Textual de los Casos de Uso



Modelado de Casos de Uso

- La Disciplina de Requisitos
- Vista de Casos de Uso
- Modelado de Casos de Uso
- **Diagramas de Actividad**



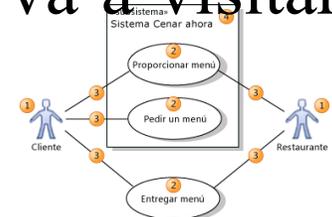
Diagramas de Actividad (1)

- Es similar a un diagrama de estados pero con un propósito diferente, que es el de capturar acciones y sus resultados en términos de cambios de estado de objetos.
- Se entiende como acciones a los trabajos y actividades que deben realizarse, modelados como nodos de actividad ejecutable. Pueden usarse con distintos propósitos:
 - Capturar las acciones que deben realizarse cuando una operación se ejecuta.
 - Capturar el trabajo interno de un objeto.
 - Mostrar cómo se puede realizar un conjunto de acciones relacionadas y cómo afectan a los objetos que están a su alrededor.
 - Mostrar cómo una instancia de un caso de uso puede realizarse en términos de acciones y cambios de estado de objetos.
 - Mostrar cómo funciona una empresa en términos de empleados (actores), flujos de trabajo (workflows), organización y objetos (factores físicos e intelectuales utilizados en el negocio).

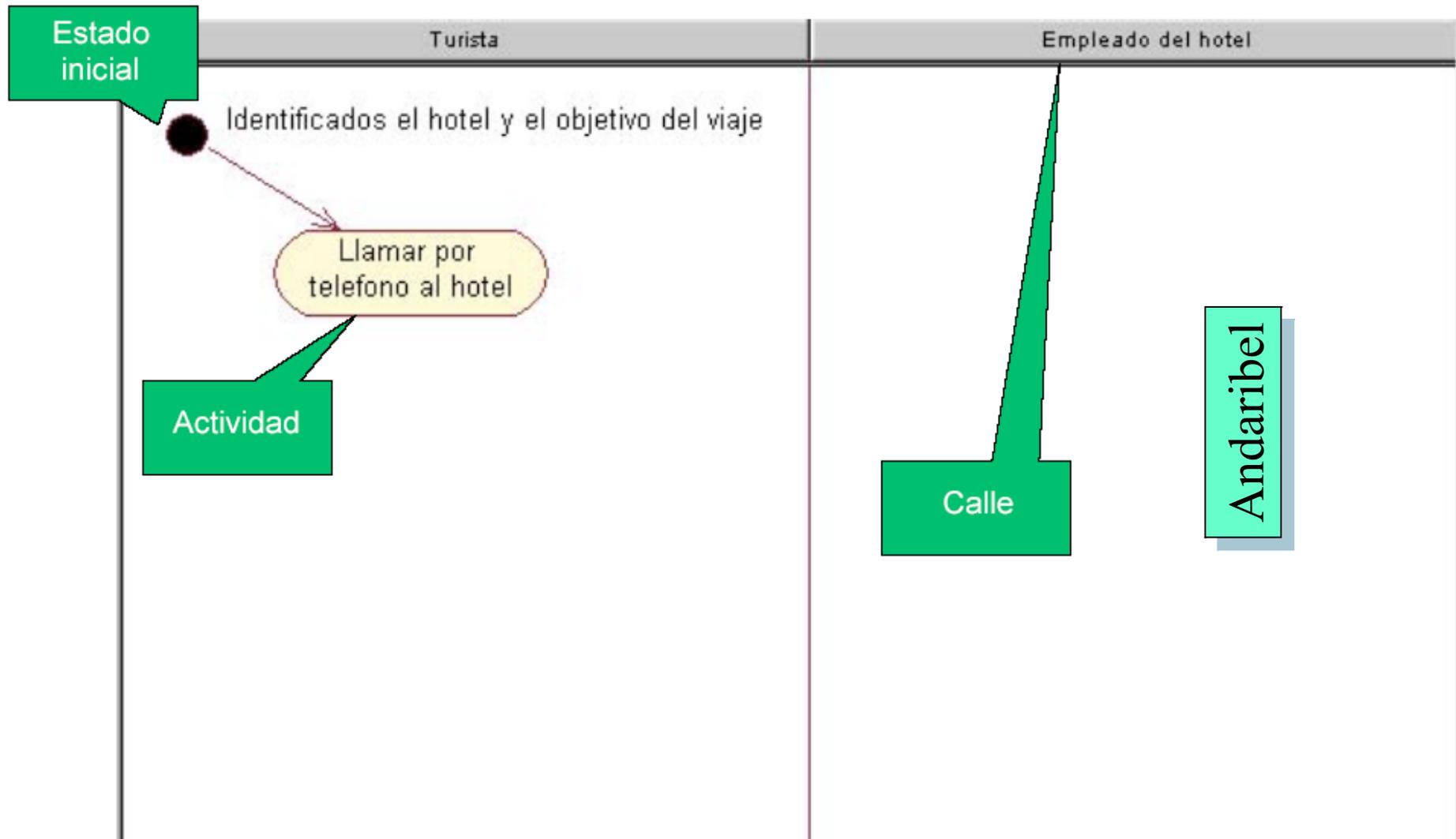
Diagramas de Actividad (2)

Ejemplo: un turista desea hacer una reserva de hotel

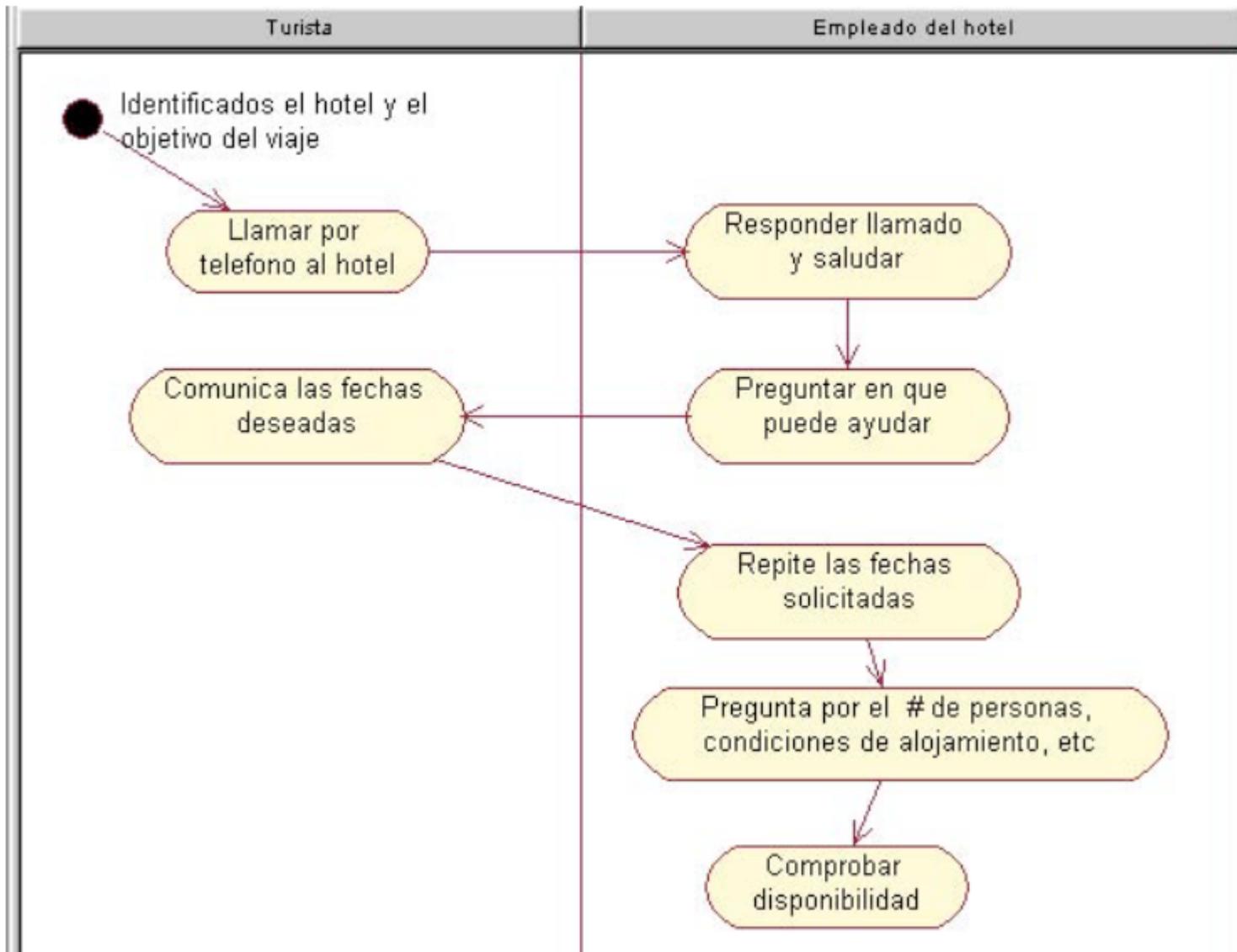
- Julia quiere salir de vacaciones de la ciudad a un lugar que no conozca.
- Le gustaría que la trataran muy bien durante su estadía de dos noches.
- Normalmente se aloja en hoteles de grandes cadenas, pero le gustaría considerar un hotel independiente con buenas referencias.
- Le preocupan el acceso desde el aeropuerto, la proximidad a las principales atracciones y la seguridad.
- Ha encontrado algunos hoteles a tener en cuenta y va a visitar los sitios Web de todos ellos.



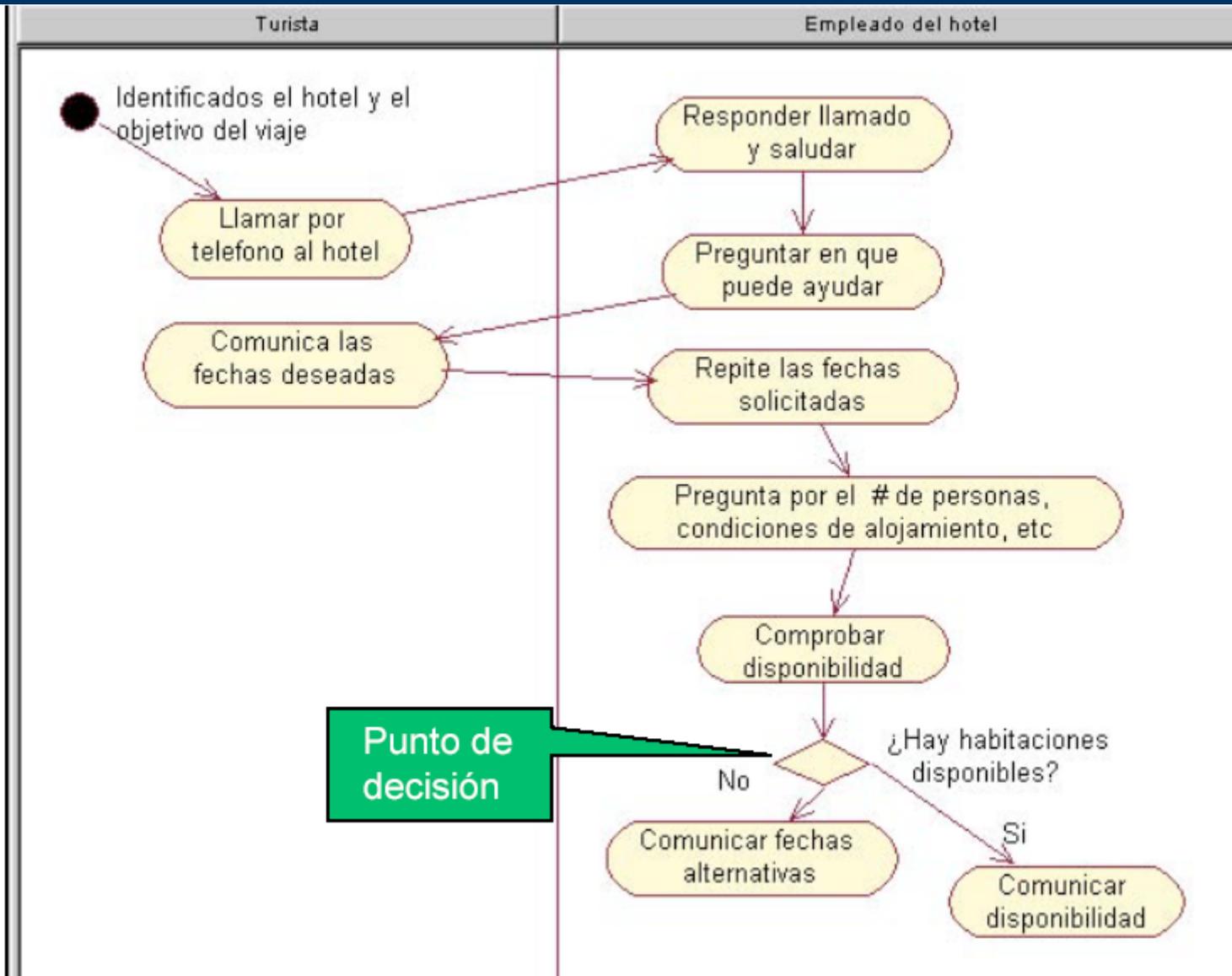
Diagramas de Actividad (3)



Diagramas de Actividad (4)

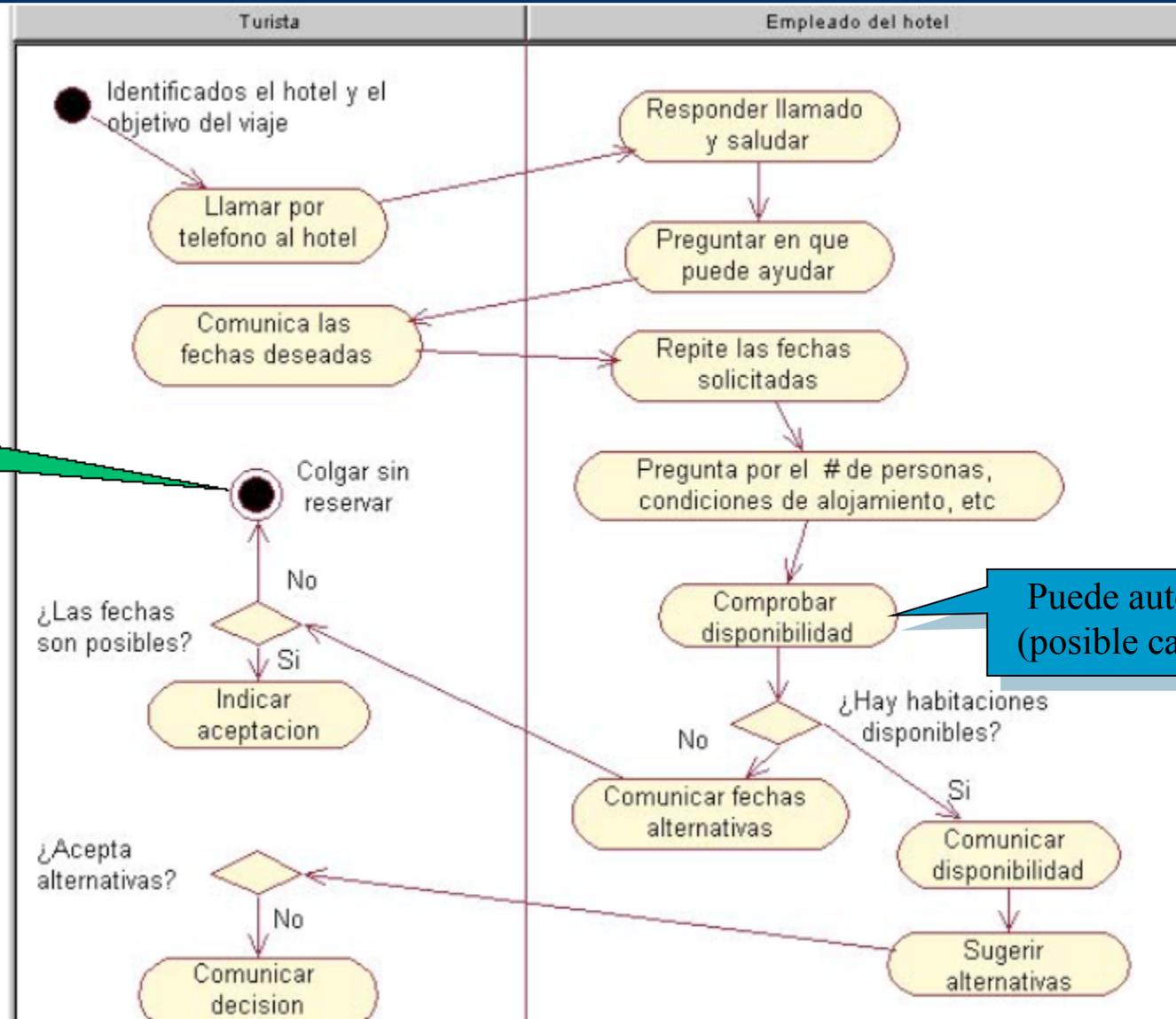


Diagramas de Actividad (5)



Diagramas de Actividad (6)

Estado final

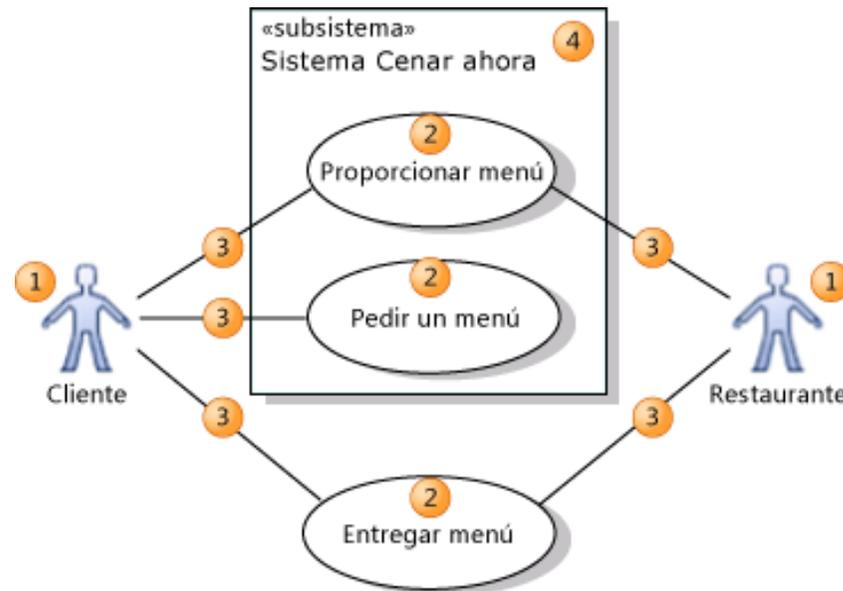


Puede automatizarse (posible caso de uso)

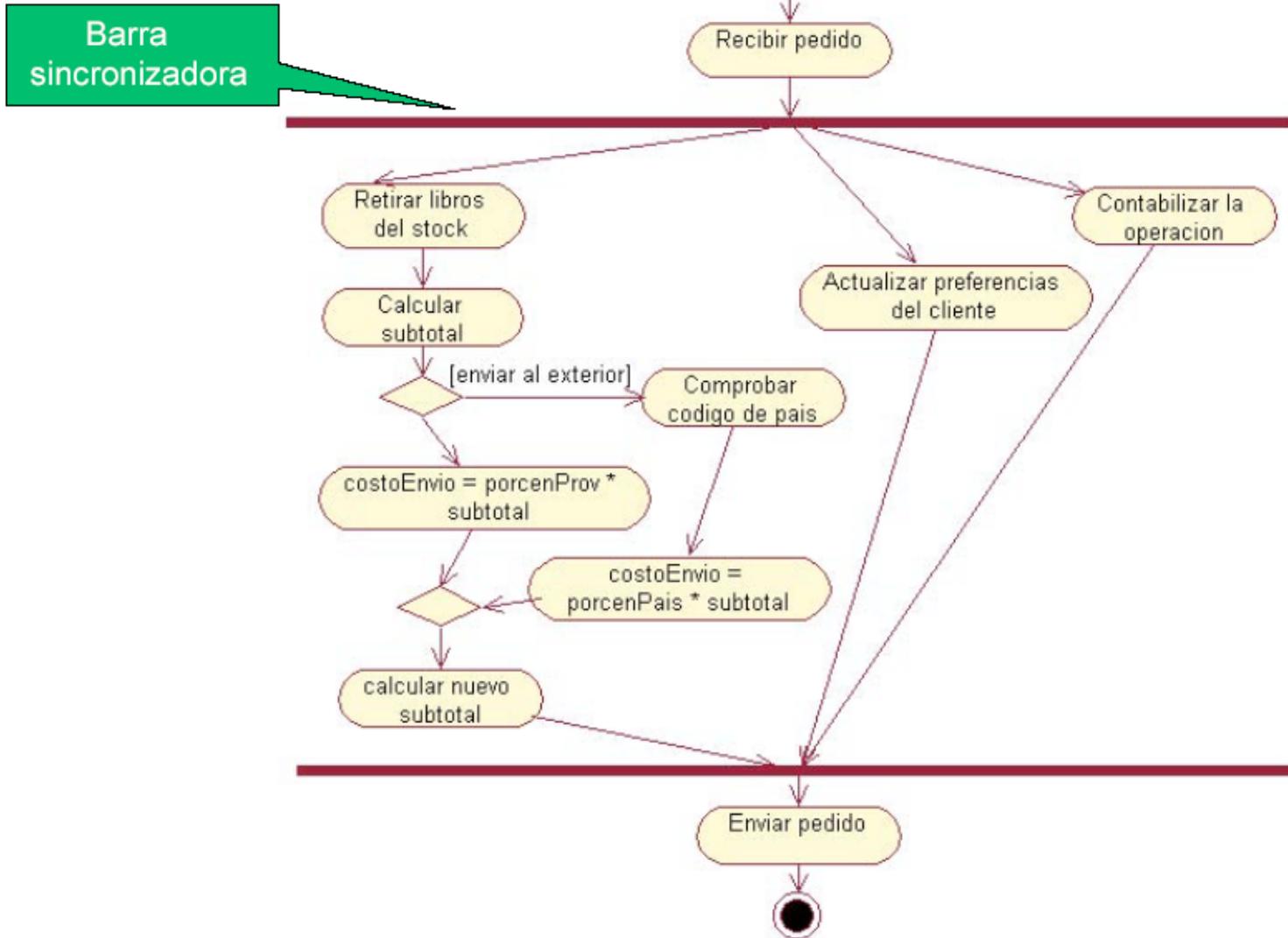
Diagramas de Actividad (7)

Ejemplo: recibimos pedidos de los clientes para nuestra librería.

- Una cliente realiza un pedido en nuestra librería por Internet.
- Cuando llega el pedido tenemos que ocuparnos del envío, de actualizar las preferencias del cliente y de contabilizar el pedido.
- Una vez realizadas todas las actividades, enviamos el paquete con los libros.

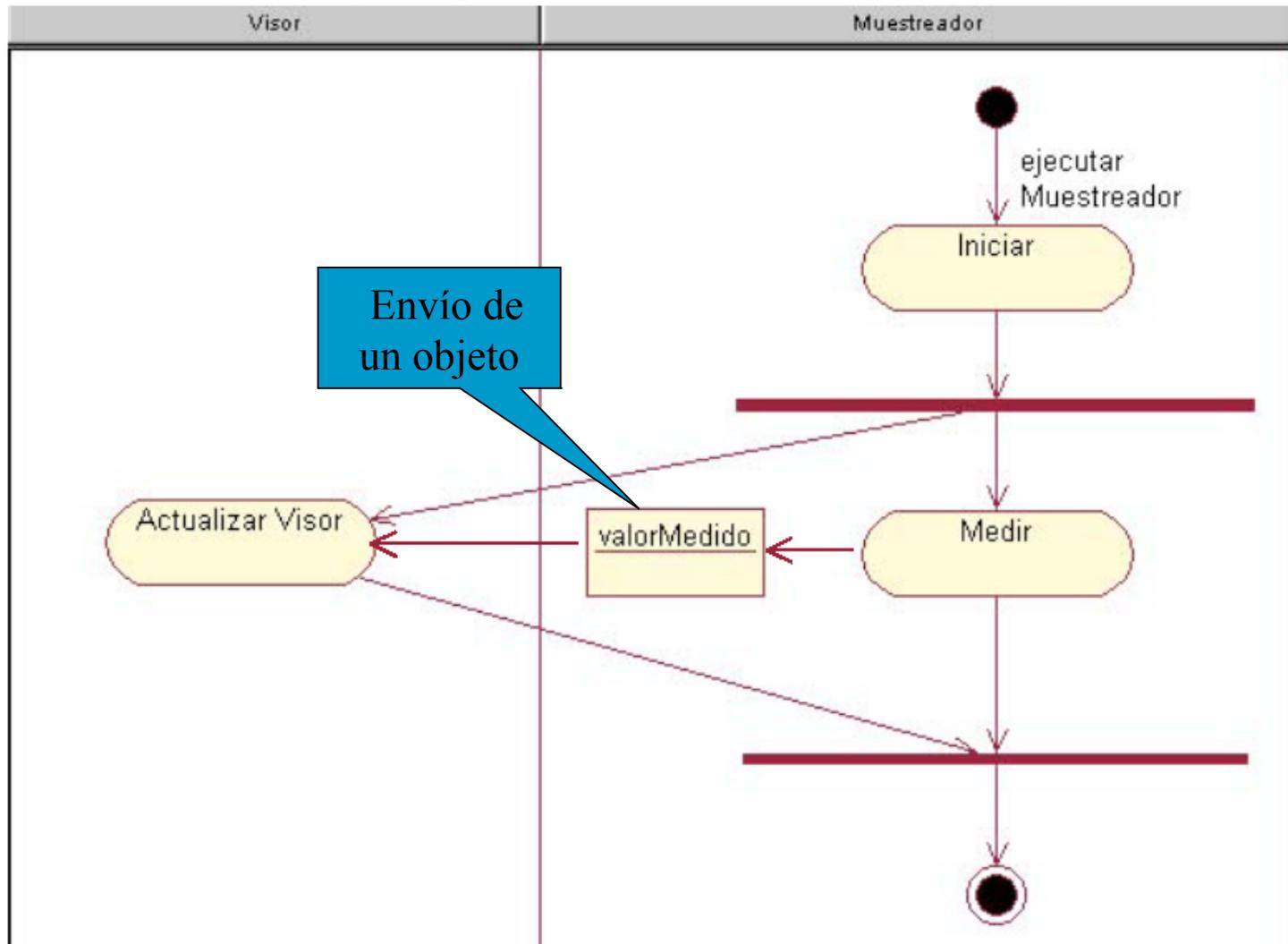


Diagramas de Actividad (8)



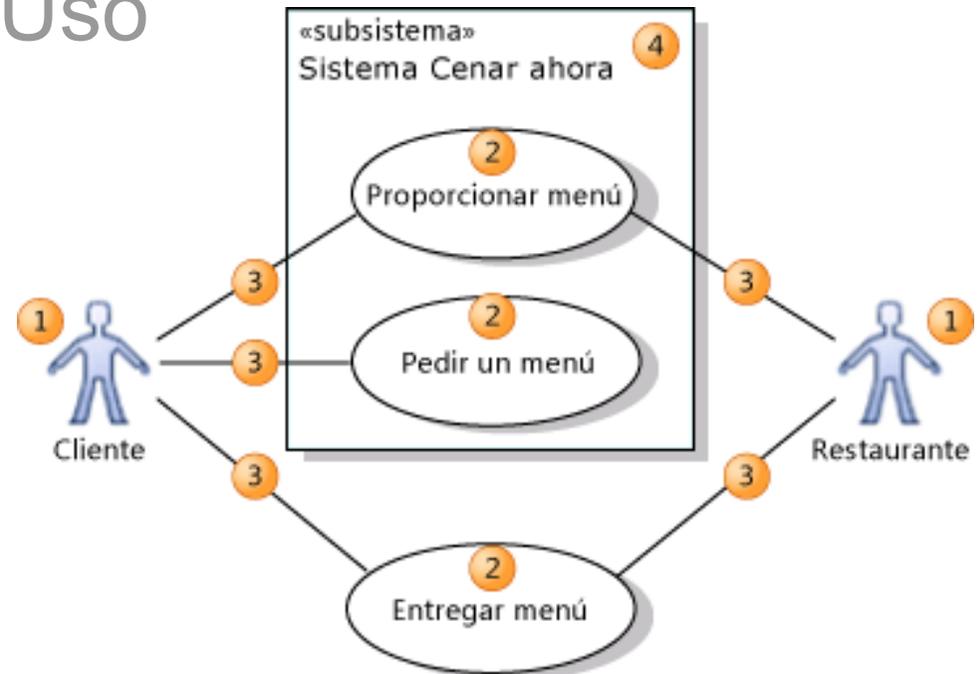
Diagramas de Actividad (9)

A veces puede haber algún objeto que está involucrado en un diagrama de actividad.



Modelado de Casos de Uso

- La Disciplina de Requisitos
- Vista de Casos de Uso
- Modelado de Casos de Uso
- Diagramas de Actividad



Requerimientos y Casos de Uso

- Contenido
 - Introducción
 - Técnicas de Educación de Requerimientos
 - Modelo IEEE/ANSI 830-1993
 - Modelado de Casos de Uso

