

Ingeniería del Software

Práctica 1.2

Especificación de requisitos

2º Grado en Ingeniería Informática
Universidad de Córdoba
Curso 2024/25

1. Organización de la práctica

2. Problema a resolver

3. Extracción de requisitos

4. Historias de usuario

5. Casos de uso

1. Organización de la práctica

Sesiones

- Sesión 1. Extracción y análisis de requisitos.
 - Se realizará una simulación de entrevista con el cliente.
 - Los equipos extraerán y formularán los requisitos.
- Sesión 2. Historias de usuario.
 - Cada equipo deberá crear la lista de producto y las tareas de planificación en YouTrack.
- Sesión 3. Casos de uso.
 - Completar especificación de requisitos y validación con el cliente.
 - Diagramas en Visual Paradigm.

1. Organización de la práctica

Entrega

- Documento formal con:
 - Especificación de requisitos
 - Historias de usuario
 - Diagramas de casos de uso
- YouTrack/Trello:
 - Historias de usuario
 - Planificación de los *sprints*
 - Reuniones

Grupo	GM2	GM4	GM3	GM1
Fecha entrega	28/10	29/10	29/10	30/10

1. Organización de la práctica

2. Problema a resolver

3. Extracción de requisitos

4. Historias de usuario

5. Casos de uso

2. Problema a resolver

Breve descripción del problema

En clase se expondrá el problema a resolver. El alumnado utilizando técnicas de elicitación de requisitos deberá obtener una lista preliminar de estos requisitos.

1. Organización de la práctica

2. Problema a resolver

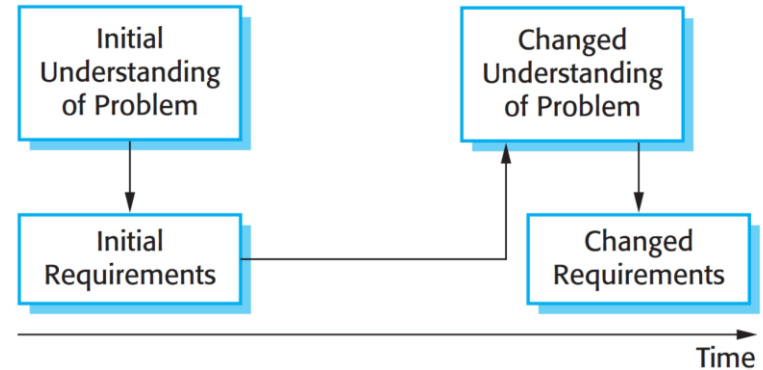
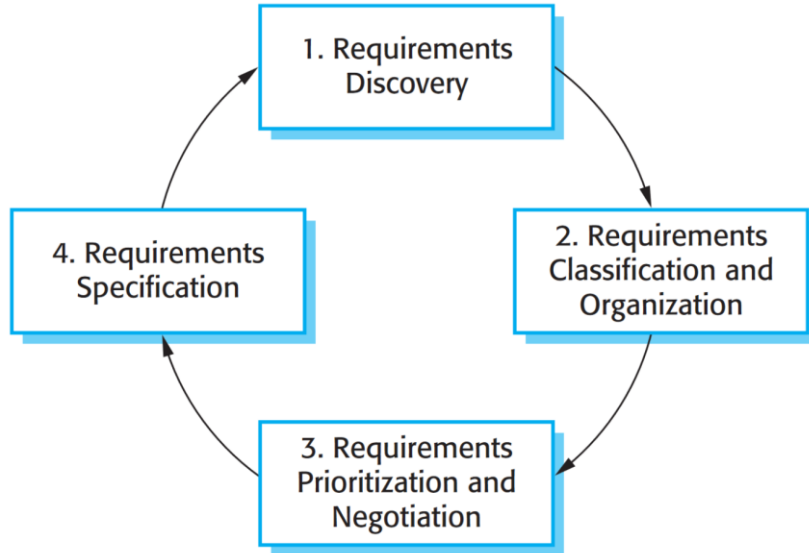
3. Extracción de requisitos

4. Historias de usuario

5. Casos de uso

3. Extracción de requisitos

Obtención, análisis, y evolución de los requisitos



3. Extracción de requisitos

Tipos de requisitos

- **Funcionales**

- Conjunto de funcionalidades o servicios que el sistema debe ofrecer.
- Expresan qué debe hacer el sistema; cómo debe reaccionar ante ciertas entradas.

- **De información**

- Detallan qué información necesita manejar el sistema y cómo se organiza.

- **No funcionales**

- Indican restricciones al sistema que pueden afectar a la calidad del servicio.
- Expresan cómo debe comportarse el sistema.
- Pueden hacer referencia a consideraciones sobre el desarrollo del sistema.

3. Extracción de requisitos

Ejemplos de requisitos funcionales

- **RF1.** El paciente debe ser capaz de buscar sus citas médicas para todas las clínicas dado su número de identidad (DNI o pasaporte).
- **RF2.** El sistema debe generar el listado de pacientes con cita para un día y clínica determinados.
- **RF3.** El sistema debe devolver al paciente las franjas horarias disponibles para la especialidad solicitada antes de confirmar la cita.

3. Extracción de requisitos

Ejemplos de requisitos de información

- **RI1.** El sistema debe almacenar los datos de contacto del paciente que recibe tratamiento, así como su historial médico completo.
- **RI2.** El sistema debe almacenar el tratamiento aplicado al paciente, incluyendo medicación, tiempos de toma, y plazos de revisión.
- **RI3.** El sistema debe almacenar la dirección, localidad, código postal, y teléfono de contacto de cada clínica.

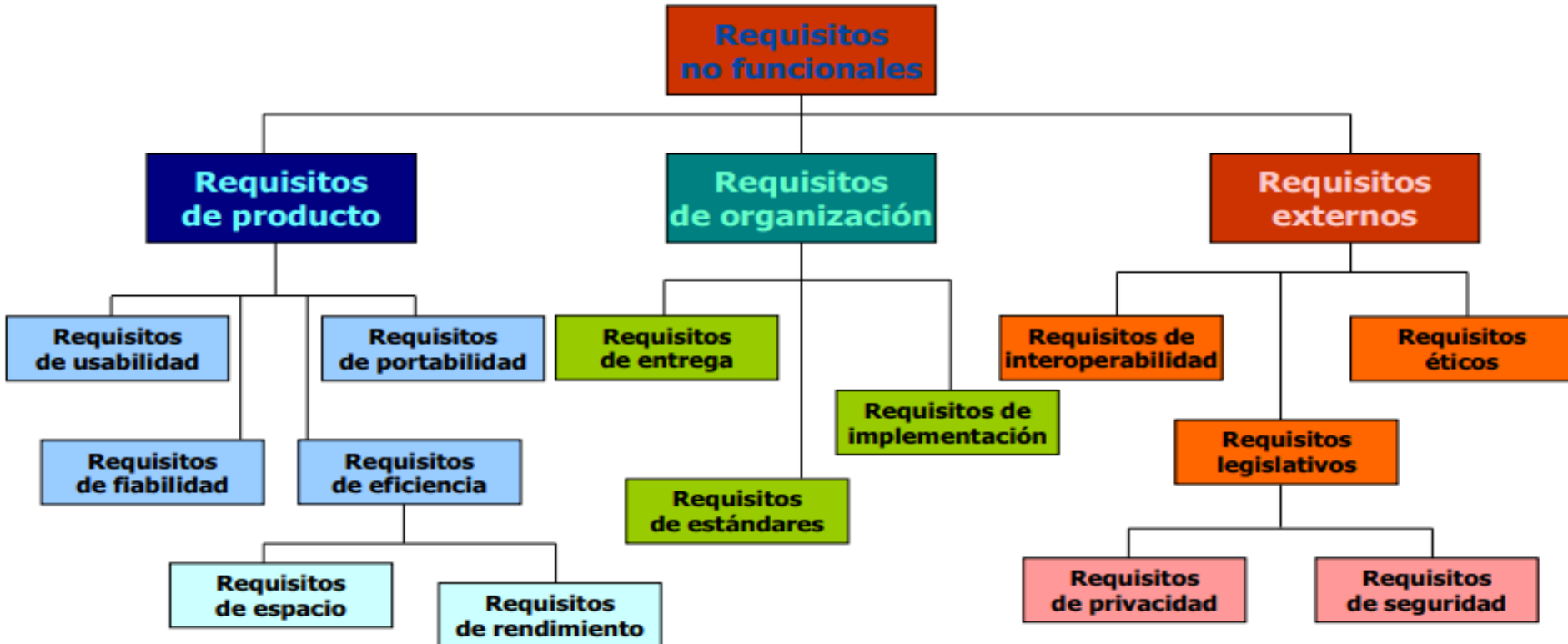
3. Extracción de requisitos

Ejemplos de requisitos no funcionales

- **RNF1.** El sistema debe estar operativo para todas las clínicas en el horario de trabajo (L-V, 8:30-17:30). Cualquier caída del sistema durante este horario no debe exceder los 5 segundos en un día.
- **RNF2.** Los empleados de la clínica deben identificarse a sí mismos utilizando su tarjeta identificativa autorizada.
- **RNF3.** Los datos de los pacientes deben ser tratados de acuerdo al reglamento general de protección de datos europeo (RGDP).

3. Extracción de requisitos

Requisitos no funcionales



3. Extracción de requisitos

Características



3. Extracción de requisitos

Entrevista

Información sobre el cliente	<ul style="list-style-type: none">* Nombre* Compañía / departamento* Cargo / rol
Identificación de problemas	<ul style="list-style-type: none">* ¿Cuál es el problema?* ¿Por qué es un problema?* ¿Cómo se resuelve actualmente?* ¿Cómo le gustaría que se resolviese en realidad?
Entorno de usuario	<ul style="list-style-type: none">* ¿Quién utilizará el sistema?* ¿Qué nivel de experiencia tienen los usuarios?* ¿Qué plataformas tecnológicas utilizan actualmente?* ¿Existen otros sistemas con los que se vaya a interactuar?
Identificar soluciones	<ul style="list-style-type: none">* Plantear soluciones alternativas* Establecer prioridades
Identificar requisitos no funcionales	<ul style="list-style-type: none">* Conocer expectativas* Preguntar sobre la instalación, configuración y mantenimiento* Recopilar información sobre aspectos de regulación o legales que puedan influir
Resumen del problema	<ul style="list-style-type: none">* Describir las necesidades extraídas* Describir los problemas de la solución actual, si la hay

1. Organización de la práctica

2. Problema a resolver

3. Extracción de requisitos

4. Historias de usuario

5. Casos de uso

4. Historias de usuario

Historias de usuario en Scrum

- Sirven para entender la funcionalidad desde el punto de vista del usuario.
- En forma de tarjeta, con estructura determinada.
- Ayudan a plasmar claramente las funcionalidades y a validar todas las necesidades del cliente.
- Responsabilidad del dueño de producto, pero pueden ser refinadas en los *sprints*.

4. Historias de usuario

Principios

- *Independent*. No depende de ninguna otra historia.
- *Negotiable*. Es flexible al diálogo entre cliente y equipo.
- *Valuable*. Añade valor al producto final.
- *Estimable*. El esfuerzo requerido debe ser calculable.
- *Small*. Abordable en un único *sprint*.
- *Testable*. Debe poder ser validada.

4. Historias de usuario

Estructura

“Como [rol de usuario], **quiero** [objetivo], **para** [beneficio/valor].”

- **Como**. Pensar en el usuario concreto, capacidades, y necesidades.
- **Quiero**. Expresar la intención que se persigue.
- **Para**. Visualizar el valor que aporta al usuario en el contexto del proyecto.

*Como enfermero/a, **quiero** disponer de una lista de pacientes **para** gestionar la llegada de pacientes a la sala de espera*

4. Historias de usuario

Estructura

- **Título.** Breve frase que expresa quién hace una acción y con qué necesidad.
- **Identificador.** Código único.
- **Descripción.** Explicación más detallada que aporta contexto. Puede enlazar a otros elementos e incluir limitaciones.
- **Responsable.** Persona encargada de completar esta historia de usuario.

4. Historias de usuario

Estructura

- **Criterios de validación.** Condiciones para dar por finalizada la implementación. Han de ser atómicos, no ambiguos, verificables y completos.
- **Discusión.** Conversaciones entre cliente, equipo y dueño del producto. Añaden detalles o nuevas percepciones.
- **Prioridad.** Indica la importancia con que debe realizarse dicha tarea.
- **Estimación en días.**
- **Sprint.** Indicador del *sprint* en que se encuentra.

4. Historias de usuario

Ejemplo de historia de usuario (I)

H1. COMO enfermero/a QUIERO disponer de una lista de pacientes PARA gestionar la llegada de pacientes a la sala de espera

El enfermero/a debe introducir la fecha, que por defecto debe ser el día actual. El sistema comprueba si hay citas ese día y muestra las especialidades con pacientes. El enfermero/a puede imprimir esta lista pulsando un botón *Imprimir*.

Para consultar una especialidad, el sistema debe permitir seleccionar una de una lista desplegable. Una vez filtradas, debe habilitarse de nuevo la opción *Imprimir*, imprimiendo solo las consultas de esa especialidad. El sistema permite cancelar el proceso de impresión y seleccionar otra especialidad.

En el listado debe figurar el nombre de los pacientes, la especialidad de consulta, y la hora de la cita (sea impresa o no).

4. Historias de usuario

Ejemplo de historia de usuario (II)

H1. COMO enfermero/a QUIERO disponer de una lista de pacientes PARA gestionar la llegada de pacientes a la sala de espera

Responsable: Jose Moyano.

Criterios de validación:

- El enfermero/a debe tener un usuario de acceso permitido a la consulta de pacientes únicamente de su clínica de trabajo.
- La fecha introducida debe ser igual o posterior a la fecha actual.
- Si no existen pacientes con cita para ese día o especialidad, el sistema debe mostrar un mensaje de error informando de la causa.

Conversaciones:

- 06/10/2021: El cliente expresa que quiere que el listado pueda ser por especialidad.
- 14/10/2021: Se considera añadir un campo de búsqueda que filtre por especialidad dadas las primeras letras del nombre.

4. Historias de usuario

Ejemplo de historia de usuario (III)

H1. COMO enfermero/a QUIERO disponer de una lista de pacientes PARA gestionar la llegada de pacientes a la sala de espera
<u>Prioridad</u> : menor
<u>Estimación en días</u> : 4
<u>Sprint</u> : Sprint1

4. Historias de usuario

Historias de usuario en YouTrack

The screenshot shows a YouTrack user story interface. At the top, there are icons for navigation and actions. Below them, the story ID 'P1-5' and creator 'aramirez' are visible. The title 'Cambiar dirección de envío' is highlighted with a red box. The description 'Como cliente quiero cambiar la dirección de envío de un pedido para que me pueda llegar a casa o a la oficina' is also highlighted with a red box. A validation section, marked with a red box, contains a bullet point: 'El cliente puede cambiar la dirección de entrega de cualquiera de los pedidos que tiene pendiente de envío'. Below this is a placeholder for an image. On the right, a sidebar shows the story's metadata, which is highlighted with a red box. At the bottom, there is a comment section with a text input, a file attachment button, and a markdown editor. The visibility is set to 'Igual que la incidencia'.

P1-5 Creada por aramirez ahora mismo Visible para Todos los usuarios ▼

☆ Cambiar dirección de envío

Como cliente quiero cambiar la dirección de envío de un pedido para que me pueda llegar a casa o a la oficina

Validación:

- El cliente puede cambiar la dirección de entrega de cualquiera de los pedidos que tiene pendiente de envío

Haga clic para adjuntar, arrastre archivos o pulse Ctrl+V para pegar una imagen

Configuración de actividad ▼

Escribir un comentario, @mencionar personas

Adjuntar archivo... ![] (attachment.png) **Negrita** # Título `código` ```bloque de código``` Markdown ▼ ⓘ

Añadir comentario Visible para Igual que la incidencia ▼

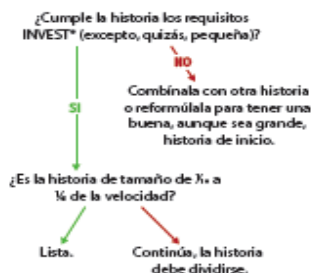
Proyecto	Proyecto1
Priority	Minor
Type	Task
State	Open
Assignee	aramirez
Sprints	Sprint1
Ideal days	2

Observadores 0 > ☆ Observar incidencia

Paneles > + Agregar al panel

COMO DIVIDIR UNA HISTORIA DE USUARIO

1 PREPARA LA HISTORIA



PASOS DE UN WORKFLOW

¿Puedes dividir la historia de manera que hagas el inicio y el final del workflow y luego la completes con las historias del medio?

¿Puedes dividir la historia haciendo una pequeña loncha de todo el workflow, y luego mejorarlo con más historias?

RETRASA RENDIMIENTO

¿Podrías dividir la historia para hacerla funcionar primero y luego mejorarla cumpliendo los requisitos no funcionales?

¿Debe la historia mucha de su complejidad a requerimientos no funcionales, como el rendimiento?

2 APLICA LOS PATRONES DE DIVISION

¿Podrías dividir la historia para hacer primero el núcleo sencillo y mejorar posteriormente con el resto?

SIMPLE / COMPLEJA

¿Tiene la historia un núcleo simple que provee la mayoría del valor o aprendizaje?

Cuando realizas la división obvia, ¿es la que decides hacer primero la más difícil de todas?

¿Podrías agrupar las últimas historias y diferir la decisión sobre cual viene primera?

ESFUERZO IMPORTANTE

OPERACIONES

¿Puedes dividir cada operación en historias separadas?

¿Esta historia incluye múltiples operaciones? (p. ej. ¿es sobre "gestionar" o "configurar" algo?)

VARIACIONES EN LAS REGLAS DE NEGOCIO

¿Puedes dividir la historia para hacer primero un subconjunto de las reglas, y mejorar con reglas adicionales posteriormente?

¿Tiene la historia variedad en las reglas de negocio? (p. ej. ¿hay términos del dominio en la historia como "fechas flexibles" que sugieren diferentes variaciones?)

¿Hace lo mismo la historia sobre diferentes tipos de datos?

VARIACIONES EN DATOS

¿Puedes dividir la historia para procesar primero un tipo de dato, y mejorar posteriormente con más tipos?

¿La historia tiene una interfaz complicada?

¿Hay una versión más simple que podrías realizar antes?

VARIACIONES EN LA INTERFAZ

¿Gestiona la historia los mismos tipos de datos mediante interfaces múltiples?

¿Puedes dividir la historia para gestionar datos de un tipo de interfaz, y mejorar posteriormente con el resto?

CONSTRUYE UN "SPIKE"

¿Todavía estás confuso sobre cómo dividir la historia?

¿Puedes encontrar una pequeña parte que entiendas bien para empezar?

Escribe primero esa historia, constrúyela, y empieza el proceso de nuevo

¿Puedes definir las 2 o 3 preguntas que más te inquietan?

Escribe un spike para responder a esas preguntas, construye lo mínimo para responderlas, y empieza el proceso de nuevo.

3 EVALUA LA DIVISION

¿Son las nuevas historias parecidas en tamaño?

¿Es cada historia un 1/4 a 1/2 de tu velocidad?

Prueba otro patrón en la historia original o las historias divididas resultantes.

¿Cumple cada una INVEST?

Prueba otro patrón.

¿Hay historias que puedas repriorizar o borrar?

Prueba otro patrón. Probablemente tienes desperdicio en cada una de tus historias.

¿Hay una historia obvia por la que empezar que proporcione valor temprano, aprendizaje o mitigación del riesgo?

Acabado, aunque podrías probar otro patrón para ver si funciona mejor.

Prueba otro patrón para ver si lo puedes obtener.

* INVEST - Las historias deberían ser:
Independiente
Negociable
Valiosa
Estimable
(S)Pequeña
Testable

4. Historias de usuario

Ejemplo de historia de usuario

Prescribing medication

Kate is a doctor who wishes to prescribe medication for a patient attending a clinic. The patient record is already displayed on her computer so she clicks on the medication field and can select 'current medication', 'new medication' or 'formulary'.

If she selects 'current medication', the system asks her to check the dose; if she wants to change the dose, she enters the new dose then confirms the prescription.

If she chooses 'new medication', the system assumes that she knows which medication to prescribe. She types the first few letters of the drug name. The system displays a list of possible drugs starting with these letters. She chooses the required medication and the system responds by asking her to check that the medication selected is correct. She enters the dose then confirms the prescription.

If she chooses 'formulary', the system displays a search box for the approved formulary. She can then search for the drug required. She selects a drug and is asked to check that the medication is correct. She enters the dose then confirms the prescription.

The system always checks that the dose is within the approved range. If it isn't, Kate is asked to change the dose.

After Kate has confirmed the prescription, it will be displayed for checking. She either clicks 'OK' or 'Change'. If she clicks 'OK', the prescription is recorded on the audit database. If she clicks on 'Change', she reenters the 'Prescribing medication' process.



Task 1: Change dose of prescribed drug

Task 2: Formulary selection

Task 3: Dose checking

Dose checking is a safety precaution to check that the doctor has not prescribed a dangerously small or large dose.

Using the formulary id for the generic drug name, look up the formulary and retrieve the recommended maximum and minimum dose.

Check the prescribed dose against the minimum and maximum. If outside the range, issue an error message saying that the dose is too high or too low. If within the range, enable the 'Confirm' button.

Historia de usuario

Descomposición en tareas

1. Organización de la práctica

2. Problema a resolver

3. Extracción de requisitos

4. Historias de usuario

5. Casos de uso

5. Casos de uso

Introducción

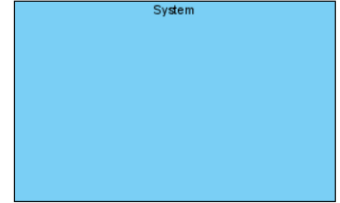
- Los casos de uso (CU) son un medio para capturar los requisitos del sistema, especificando su comportamiento.
- Un caso de uso es una descripción simple de una tarea que implica **interacción** entre el **usuario** y el **sistema**.
- Los casos de uso dan una visión general de la interacción, que debe ser luego descrita en detalle (texto, tabla, u otro tipo de diagrama).
- UML (**U**nified **M**odeling **L**anguage) proporciona notación (gráfica) para la creación de los diagramas de casos de uso.

5. Casos de uso

Elementos

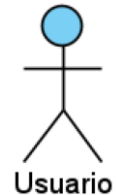
- **Sistema**

- Abstracción del programa o sistema que se va a construir.
- Se representa con una caja que marca el límite entre el sistema y el exterior.



- **Actor**

- Abstracción de una persona o sistema externo que interacciona con el sistema que estamos modelando.
- Se representa con un muñeco de palo y se ubica fuera del sistema.



5. Casos de uso

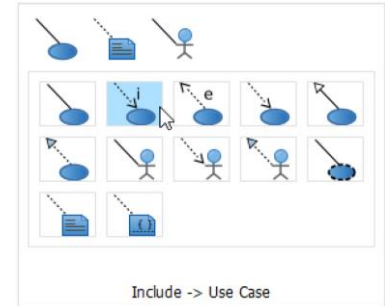
Elementos

- **Caso de uso**

- Tarea que realiza el sistema ante un estímulo (externo o interno).
- Se representa con una elipse dentro del sistema.

- **Relación**

- Conexión entre un actor y un caso de uso, entre casos de uso, o entre actores.
- Asociación, generalización, inclusión, o extensión.
- Se representa con una línea (continua o discontinua).

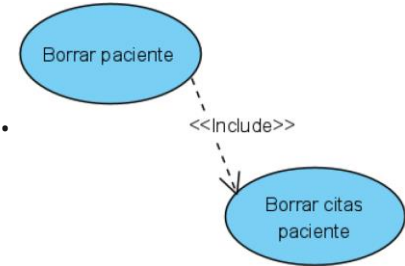


5. Casos de uso

Relaciones

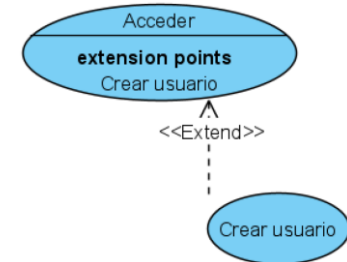
- **<< include >>**

- Un CU **incorpora** el comportamiento definido en otro CU.
- El CU necesita la parte incluida para ser descrito completamente.
- El CU incluido tiene partes comunes a 2 o más CU (reutilización).
- La flecha se dirige del CU “principal”, al que es incluido.



- **<< extend >>**

- Expresa comportamiento complementario bajo ciertas condiciones.
- El CU extendido es independiente y completo.
- La extensión sucede en un paso concreto del CU (alternativa).
- La flecha se dirige de la extensión al CU que es extendido.



5. Casos de uso

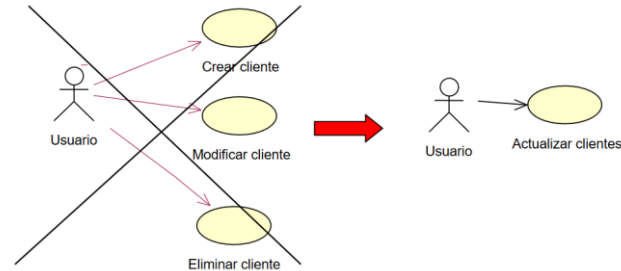
Descripción

Caso de uso	Nombre del caso de uso
Identificador	Código único que identifica al caso de uso
Objetivo	Frase corta que describe el caso de uso
Contexto	Frase más larga para detallar condiciones, contexto, etc.
Actor principal	Actor involucrado en el caso de uso
Escenario principal	Pasos numerados que detallan el proceso esperado
Extensiones	Acciones alternativas a algunos de los pasos

5. Casos de uso

Pasos a seguir

1. **Identificar a los actores.** Durante la entrevista se habrán mencionado distintos tipos de usuarios, así como sus responsabilidades y permisos. Esta información es clave para decidir quiénes son los actores que interactúan con el sistema.
2. **Identificar los principales casos.** Para cada actor identificado, deben recopilarse las acciones de las que son responsables a un nivel de abstracción elevado.



5. Casos de uso

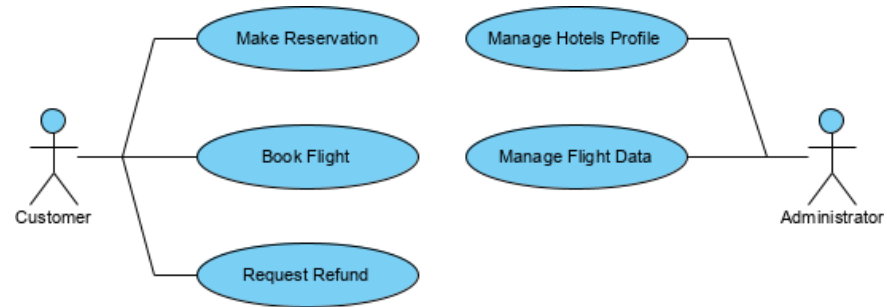
Pasos a seguir

3. **Identificar nuevos casos de uso.** Refinar la lista de casos de uso para reflejar el resto de funcionalidades, establecer relaciones, etc.
4. **Describir los casos de uso.** Crear una descripción detallada del caso de uso, incluyendo su escenario principal y los alternativos (extensiones).

5. Casos de uso

Ejemplo 1

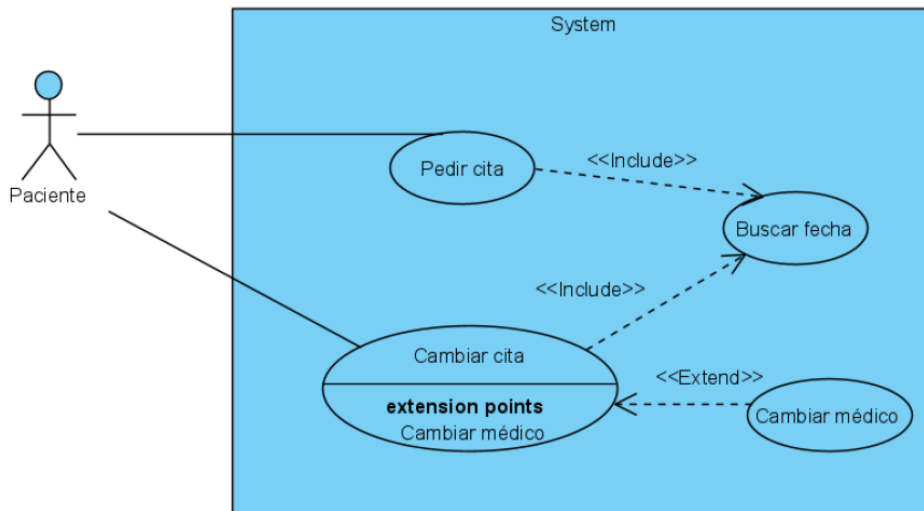
- Administrador
 - Gestionar hoteles
 - Gestionar vuelos
- Cliente
 - Hacer reserva
 - Reservar vuelo
 - Solicitar devolución



5. Casos de uso

Ejemplo 2

- Pedir cita
 - Es necesario buscar una fecha libre
- Cambiar cita
 - El paciente quiere cambiar la fecha (hay que buscar una fecha libre).
 - Puede implicar un cambio de médico según la nueva fecha.



5. Casos de uso

Ejemplo 2

Caso de uso	Cambiar cita
Identificador	CU2
Objetivo	El cliente quiere cambiar una cita médica reservada con anterioridad.
Contexto	La reserva debe existir en el sistema y tener una fecha posterior a la actual.
Actor principal	Paciente
Escenario principal	<ol style="list-style-type: none">1. El paciente se identifica con su número de tarjeta sanitaria.2. El sistema muestra las citas médicas reservadas (fecha y especialidad).3. El paciente selecciona la cita que desea modificar.4. El sistema busca las fechas disponibles (CU3: Buscar fecha).5. El paciente selecciona la fecha que desea de entre las disponibles.6. El sistema confirma la nueva cita sin que se afecte el médico asignado.
Extensiones	<ol style="list-style-type: none">6a. El sistema detecta que debe cambiarse la asignación de médico (CU4: Cambiar médico)

5. Casos de uso

Ejemplo 3

Name	Modify Reservation	
ID	03	
Description	To modify reservation inside the system.	
Main actor	Student	
Precondition	Reservation must exist	
Main flow	1	The student selects a reservation
	2	The student selects modify option
	3	The student fills up the fields
	4	The student confirms the modification
Alternative flow	3a	The data is invalid
		The system returns an error message
	Resume 3	
	4a	The student cancels the process
Postcondition	The reservation has been modified.	

5. Casos de uso

Visual Paradigm

- <https://online.visual-paradigm.com/>

1. Organización de la práctica

Entrega

- Documento formal con:
 - Especificación de requisitos
 - Historias de usuario
 - Diagramas de casos de uso
- YouTrack:
 - Historias de usuario
 - Planificación de los *sprints*
 - Reuniones

Grupo	GM2	GM4	GM3	GM1
Fecha entrega	28/10	29/10	29/10	30/10

Ingeniería del Software

Práctica 1.2

Especificación de requisitos

2º Grado en Ingeniería Informática
Universidad de Córdoba
Curso 2024/25