

# Parcial 1 - Ing. Software

## **SOFTWARE:**

Programas de computo y su documentación asociada.

- **Características:**

- Es un elemento lógico y no físico.
- No se desgasta pero se deteriora.
- No se manufactura en el sentido clásico.

- **Tipos de Productos de Software:**

- *Genéricos:*

Sistema independientes que se producen por un empresa de desarrollo y se venden en el mercado abierto.

- *Personalizados:*

Diseños para un cliente en particular.

- **Actividades Fundamentales del Proceso de Software:**

- *Especificación:*

Donde clientes e ingenieros definen el software a producir y sus restricciones

- *Desarrollo:*

Donde se diseña e implementa el software.

- *Validación:*

Donde se verifica el software para asegurar que sea lo que el cliente requiere y con la funcionalidad correcta.

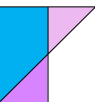
- *Evolución:*

Donde se modifica el software para reflejar los requisitos cambiantes del cliente y el mercado.

REQUISITO	ESPECIFICACIÓN
Condición necesaria para la obtención de cierto objetivo, o para el cumplimiento de cierto objetivo.	Descripción minuciosa de las características de un material que un servicio deberá presentar.
Están conectados a lo que se pretende del software.	Busca detallar sobre la óptica computacional esas pretensiones de los clientes.

## **INGENIERÍA DE SOFTWARE:**

Es la aplicación de un enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento de software.



# Ingeniería de Requisitos.

*Es la manera de sistematizar el proceso de definición de requisitos.*

Surge como respuesta a los problemas derivados de la mala definición de requisitos.  
Propone técnicas que ayudan a producir requisitos confiables.

- **Objetivo:**

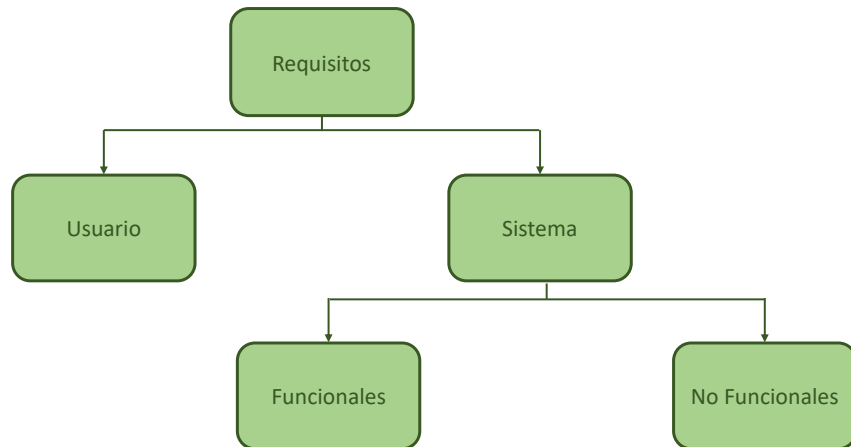
- Aplicar un conjunto de tareas de la Ingeniería a los Requisitos.

## Manejo Sistemático de los Requisitos

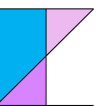
La complejidad de los sistemas exige que se preste más atención al por correcto entendimiento del problema antes del comprometimiento de la solución.

## Niveles de Requisitos

SISTEMA	USUARIO
Descripciones detalladas	Son enunciados
Especificación Funcional	Lenguaje Natural



FUNCIONALES	NO FUNCIONALES
Refieren lo que debe hacer	No se relacionan directamente con los específicos
Dependen del software y enfoque que adopte la organización	Pueden definir restricciones
Se describen de forma abstracta	Suelen ser mas significativos que los funcionales individuales



## Elicitar

Descubrir, hacer explícito, obtener el máximo de información para el conocimiento del objeto en cuestión.

- **Partes:**
  - Identificación de fuentes de información.
  - Recolectar hechos.

### TÉCNICAS DE RECOLECCIÓN DE HECHOS

Lectura de Documentos	Observación	Entrevistas	Cuestionario	Análisis de Protocolos	Enfoque Antropologico
-----------------------	-------------	-------------	--------------	------------------------	-----------------------

Estructuradas, tutoriales e informales

## Conocimiento Tacito

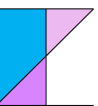
Aquel conocimiento que es trivial para el entrevistado y no para el entrevistador.

### TÉCNICAS PARA LA COMUNICACIÓN

Presentación	Entendimiento	Lenguajes	Retroalimentación
--------------	---------------	-----------	-------------------

## Modelar

Representación abstracta, gráfica, escrita, conceptual, visual y física de un problema. Permite simplificar el objeto del mundo real, simplificación que es suficientemente realista para dar una idea de lo que ocurre en el mundo real



# UML UNIFIED MODELING LANGUAGE

Conjunto de herramientas que permite modelar (analizar y diseñar ) sistemas orientados a objetos.

## PERO NO ES...

- Un metodo de desarrollo.
- Serie de pasos que llevan a producir código.
- Dependiente del ciclo de desarrollo.

## Tipos de Diagramas:

- *Estructurales o Estáticos*: muestran la organización de un sistema en términos de componentes.

Paquetes	Clases	Componentes	Instalación
----------	--------	-------------	-------------

- *Comportamiento o Dinámicos*: utilizados para visualizar, especificar, construir y documentar aspectos dinámicos de un sistema.

Actividades	Casos de Uso	Interacción	Estado
-------------	--------------	-------------	--------

## Casos de Uso

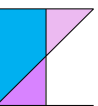
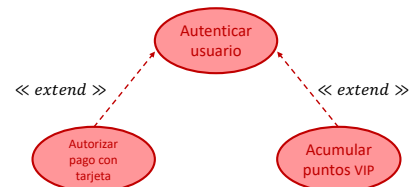
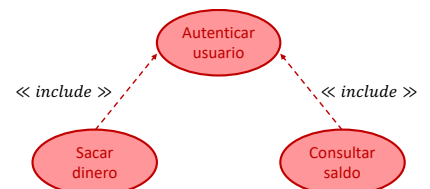
*Técnica de descubrimiento de requisitos que se introdujo en 1993.*

Describe el comportamiento del sistema en distintas condiciones.

Se define desde el punto de vista de un actor ( persona o dispositivo ).

## ORGANIZACIÓN

- `<< INCLUDE >>`
  - Denota obligatoriedad.
  - Se usa para evitar describir el mismo flujo de eventos repetidas veces.
  - Simplifica la comprensión del caso de uso base.
- `<< EXTENDS >>`
  - Denota opcionalidad.
  - Se usa cuando se tiene un caso similar a otro.
  - Flujos distintos que pueden ejecutarse en base a la selección del actor.



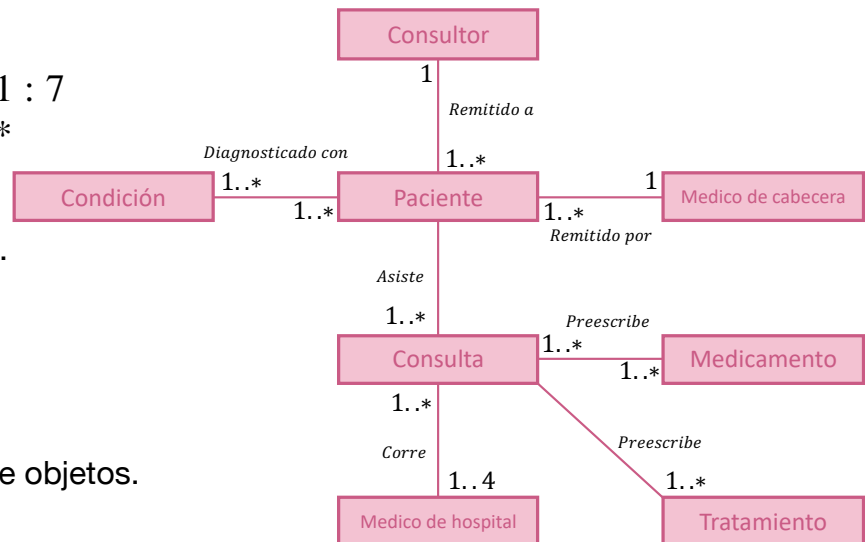
# Modelo de Dominio

*Muestra el vocabulario del sistema de forma gráfica y relacionada.*

Es útil al desarrollar un modelo de sistema orientado a objetos para mostrar las clases en un sistema y las relaciones entre dichas clases.

Es uno de los diagramas más utilizados en los primeros procesos de diseño del software.

- Tiene relaciones que se indican:
  - Con el numero exacto 1 : 1 ó 1 : 7
  - Con un numero indefinido 1 : \*
- Entidades:
  - Clases de objeto simplificadas.
- Atributos:
  - Atributos de clases de objeto.
- Relaciones:
  - Nombres de asociaciones entre objetos.

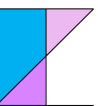


## Diagrama de Clases

Modelo estructural

LAS CLASES		
Nombre de la clase	Atributos ( opcional sus tipos )	Operaciones ( metodos en JAVA )

DIFERENTES FORMAS DE UNA CLASE			
<div>Nombre Clase</div> <div>Atributo Atributo</div> <div>Operación Operación</div>	<div>Nombre Clase</div> <div>Atributo Atributo</div>	<div>Nombre Clase</div> <div>Operación Operación</div>	<div>Nombre Clase</div>



← Mas debil relación

Mas fuerte relación →

RELACIONES				
Dependencia	Asociación	Agregación	Composición	Generalización
← - - - -	_____	◇ _____	◆ _____	◁ _____
Cuando los objetos de una clase trabajan brevemente con los objetos de otra clase.	Cuando los objetos de una clase trabajan con los objetos de otra clase durante un tiempo prolongado.	Cuando una clase posee pero comparte una referencia a objetos de otra clase.	Cuando una clase contiene objetos de otra clase.	Cuando una clase es un tipo de otra clase.

← Mas accesibles

Menos accesibles →

VISIBILIDAD			
Publica	Protegido	Paquete	Privado
+	#	~	-

## Modelo de Contexto

*La especificación de un sistema, decidir sobre las fronteras del sistema.*

Que funcionalidades ofrece el sistema y cuál es el entorno del sistema.

Cuales funcionalidades se automatizan o cuales son desarrollados por otros sistemas.

## Modelo de Comportamiento

*Son modelos Dinámicos que muestran lo que sucede cuando un sistema responde ante un estimulo en su entorno.*

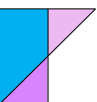
### LOS ESTÍMULOS PUEDEN SER...

Datos: Algunos se procesan por el sistema

Eventos: Algunos activan el procesamiento del sistema

### MODELO DIRIGIDO POR EVENTOS

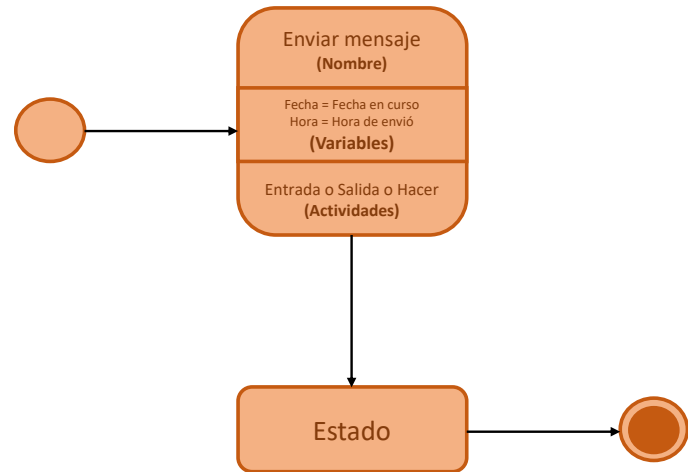
Muestra cómo responde un sistema a eventos externos o internos.



# Diagrama de Estados

Manera para caracterizar un cambio en el sistema mostrando las condiciones de un objeto y haciendo uso de transiciones que retornan *V* o *F*.

REPRESENTACIÓN	
Nombre del Estado	
Variables de Estado	
Las actividades que constan de sucesos y acciones	Entrada: Sucedec cuando el sistema entra al Estado.
	Salida: Sucedec cuando el sistema sale del Estado.
	Hacer: Sucedec cuando el sistema esta en el Estado.

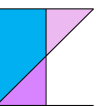
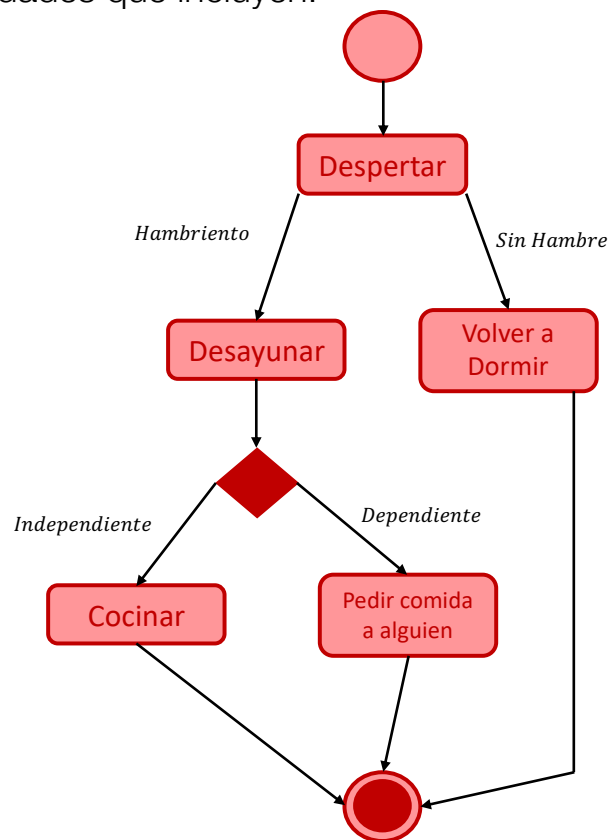


EXTENSION DE DIAGRAMA DE ESTADOS

# Diagrama de Actividades

Es una visión simplificada de lo que ocurre durante una operación o proceso mostrando las actividades que incluyen.

REPRESENTACIÓN
<b>Actividad</b> se representa por un rectángulo de esquinas redondeadas.
<b>Procesamiento</b> se realiza en una actividad y continua con otra.
<b>Flecha</b> representa la transición de una actividad a otra.
Cuenta con un <b>Punto Inicial</b> y un <b>Punto Final</b> .
<b>Puntos de Decisión</b> se representan entre actividades.



# Modelos de Interacción

Interacción entre el sistema a desarrollar y otros sistemas, implicando la entrada y salida del usuario.

## Diagramas de Secuencia

Se usa para modelar las interacciones entre actores y los objetos del sistema permitiéndonos diseñar el comportamiento del software.

### REPRESENTACIÓN

**Objetos** se colocan en la parte superior cómo rectángulos con los nombres subrayados.

**Línea de Vida** es la extensión debajo de cada objeto como una línea discontinua.

**Actividades** van junto a la línea de vida como un pequeño rectángulo que representa la ejecución de una operación.

### TIPOS DE MENSAJES

**Simple** Transferencia del control de un objeto a otro.



**Simétrico** Espera la respuesta de un mensaje para continuar el proceso.



**Asimétrico** no espera respuesta del mensaje antes de continuar.



### PRINCIPIOS BÁSICOS

- Identificar a los objetos involucrados.
- Definir la responsabilidad de cada objeto.
- Mostrar el ciclo de vida de cada objeto.

