



Mi Universidad

Mapa Conceptual

*Sofia Pereyra Orantes
Ingeniería de Requisitos
Unidad 2
Ingeniería en software
Reyes Molina Andrés Alejandro
Ingeniería en Sistemas Computacionales
Cuatrimestre 8*

Comitan de Dominguez, Chiapas a 15 de Marzo de 2025

INGENIERIA DE REQUISITOS

ANALISIS DE REQUERIMIENTOS

es
Proceso que identifica las necesidades del cliente para un proyecto de software

TIPOS DE REQUERIMIENTOS

1. Ambiente físico

2. Interfaces

3. Usuarios

4. Funcionalidad

5. Documentación

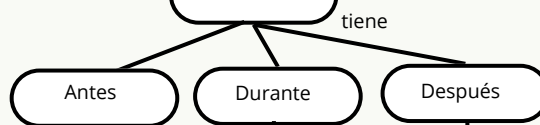
6. Datos

7. Recursos

8. Seguridad

IDENTIFICACION, ANALISIS, IDENTIFICACION

Entrevista



1. Enumerar y dar prioridad a los clientes que se entrevistarán.
2. Programar una entrevista con tiempos de inicio y terminación fijos.

1. No ser pasivo, investigar y animar, persistir en entender deseos y explorar necesidades. Examinar casos de uso, flujos de datos y/o diagramas de estado.
3. Tomar notas exhaustivas.
4. Programar una reunión de seguimiento.

1. Bosquejar la especificación de los requerimientos. Enviar correos electrónicos a los clientes para obtener sus comentarios.

VALIDACIÓN Y GESTIÓN DE REQUISITOS

Proceso de 4 pasos

son
1. Estudio de Viabilidad

2. Recogida de requisitos

3. Requisitos de Software

4. Validación de requisitos de Software

MODELADO DE ANALISIS, CASOS DE USO

Descripción de las acciones de un sistema desde el punto de vista del usuario

es una
Herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema

VALIDACION DE LOS REQUISITOS DE SOFTWARE

El usuario puede que pida soluciones ilegales y poco prácticas, y los expertos puede que interpreten los requisitos de forma incorrecta

evaluar

1. Si pueden ser implementados de manera práctica.
2. Si son válidos a nivel de funcionalidad y dominio del software
3. Si hay alguna ambigüedad
4. Si se han completado
5. Si se pueden demostrar

CONCEPTOS BASICOS DE LA ORIENTACIÓN A OBJETOS

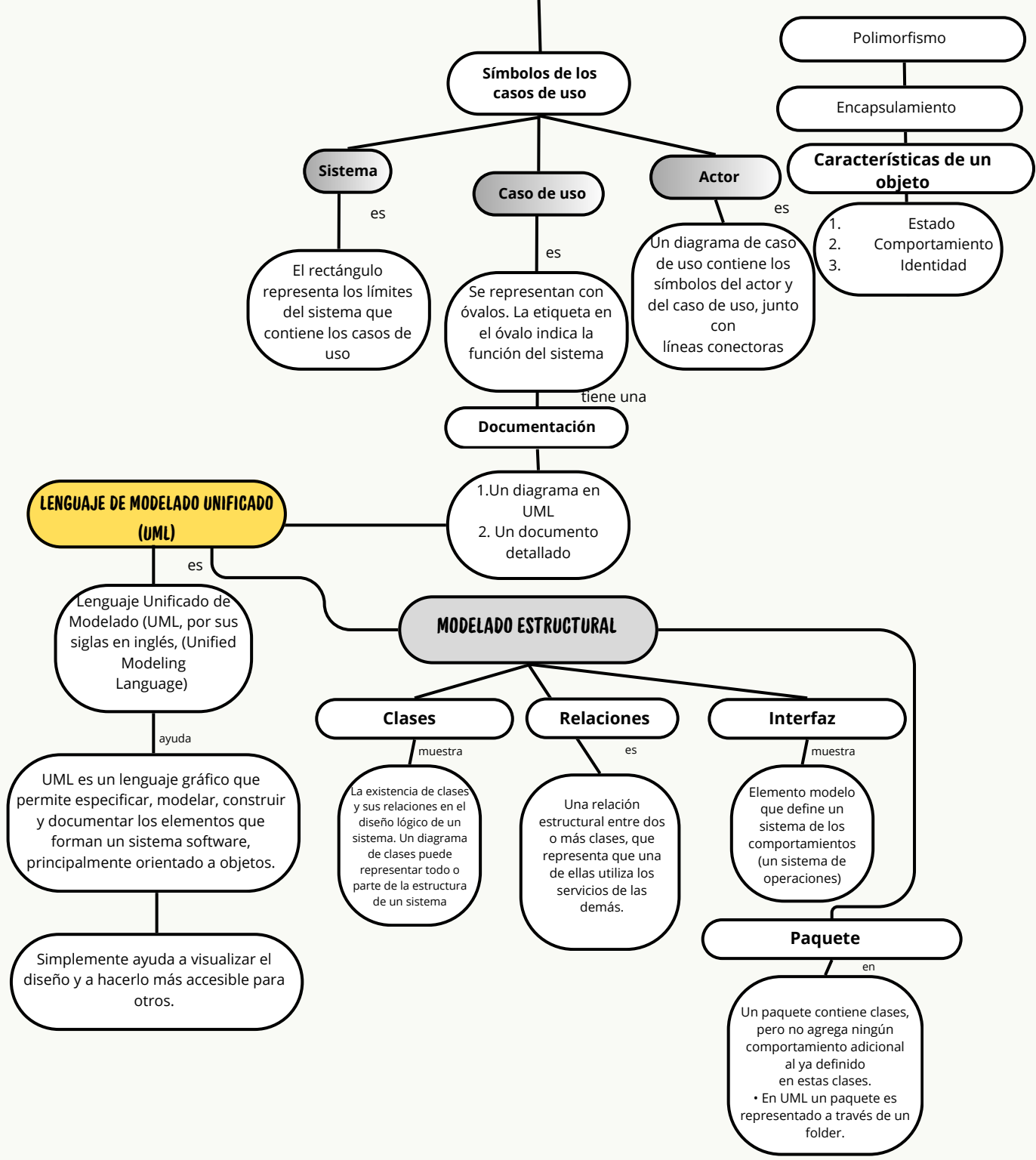
Conceptos elementales

Clase

Atributo

Método

Herencia



GRUPOS DE DISEÑO DE PAQUETES

Se deben

agrupar en los paquetes para los propósitos de organización y configuración

Se pueden utilizar los paquetes y los subsistemas como orden, configuración o unidades de la entrega cuando termina un sistema

Visibilidad de los paquetes

Public

Las clases públicas se pueden alcanzar afuera del paquete que posee. Visibility symbol:

+ Protected

Las clases protegidas se pueden alcanzar solamente por el paquete que posee, y cualquier paquete que herede del paquete que posee. Visibility symbol: #

Private

Las clases privadas se pueden alcanzar solamente por las clases dentro del paquete que posee. Visibility symbol: -

Principios Generales

Los paquetes no deben cruzar los pares (es decir codependiente)

Los paquetes en capas más bajas no deben depender de los paquetes en capas superiores.