
Universidad
Tecmilenio®

Universidad Tecmilenio

# Ingeniería de Software

## Tipos de requerimientos

Semana 2

# Bienestar-mindfulness

Te invito a realizar la siguiente actividad de bienestar-mindfulness antes de comenzar a revisar el tema:

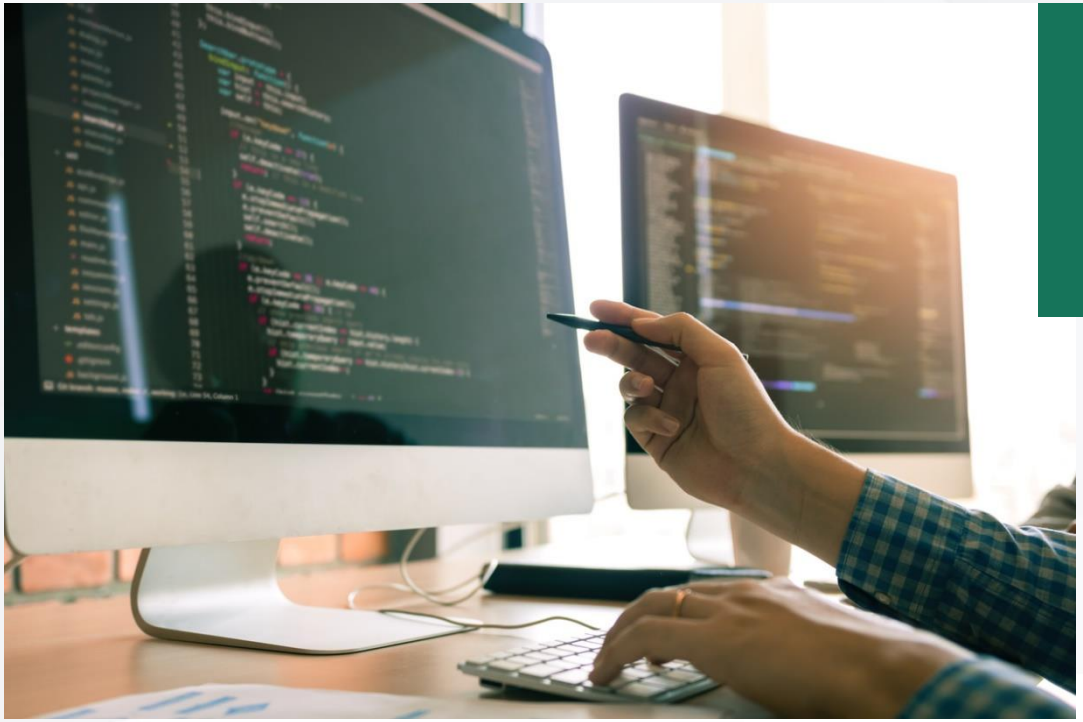
https://youtu.be/oq-klVxvm5g

# Introducción

Los requerimiento son aquellas características que los usuarios buscan en un sistema. Existen dos tipos de requerimientos.

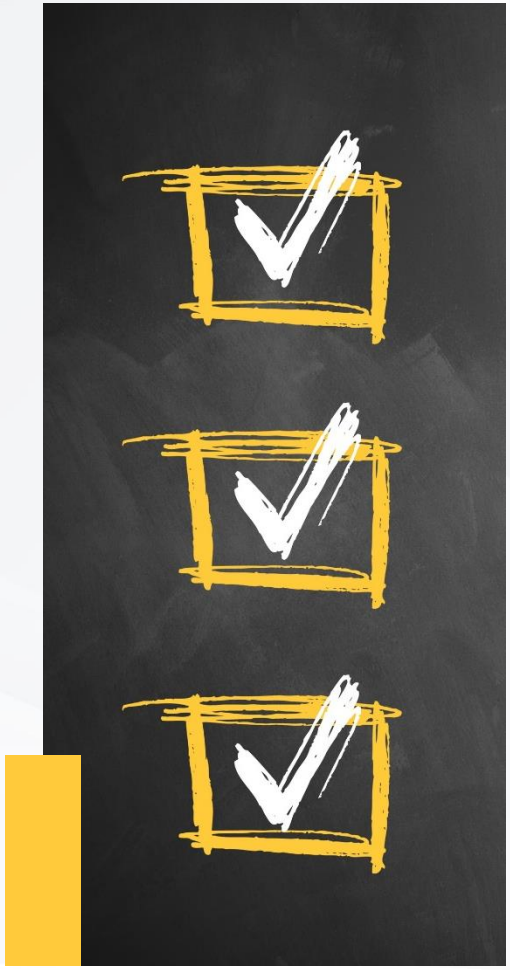
# Explicación

1. Los requerimientos funcionales establecen lo que debe hacer el sistema o las acciones que debe llevar a cabo para satisfacer la razón fundamental de su existencia.



# Explicación

2. Los requerimientos no funcionales establecen las características del desempeño que deberá demostrar el sistema en operación. Son tan importantes como los requerimientos funcionales, porque pueden ser un factor decisivo en la aceptación del usuario final.



# Explicación

Los requerimientos no funcionales son propiedades que debe tener la funcionalidad del producto de software, por ejemplo:

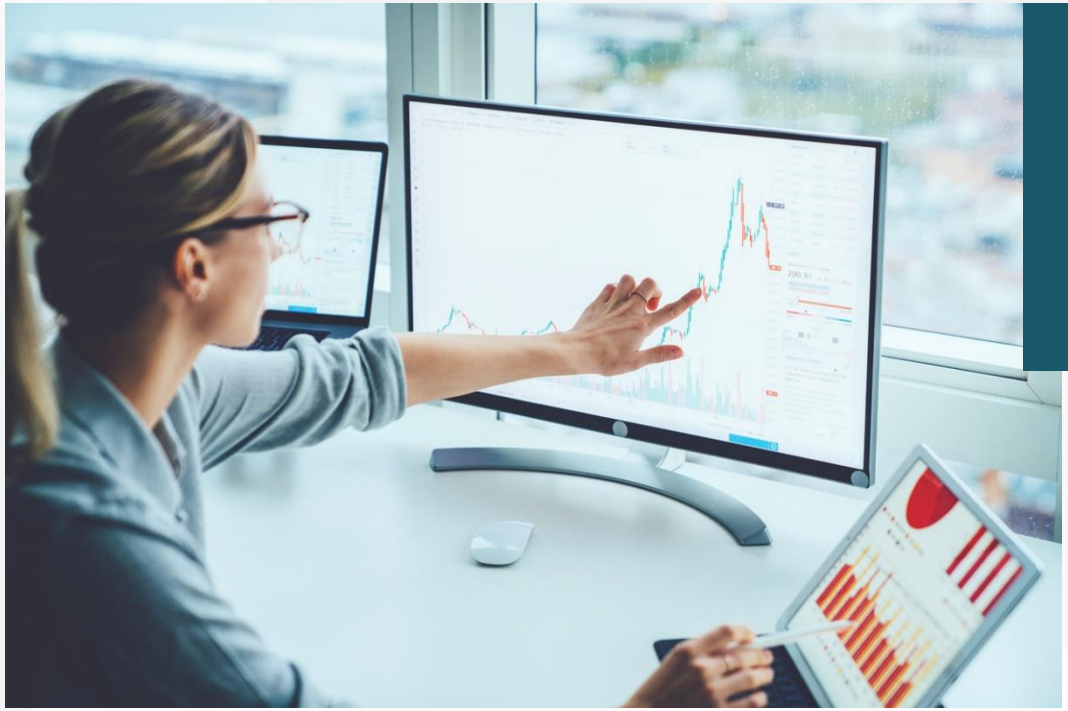
- *Look and feel*
- Usabilidad
- Rendimiento
- Entorno operacional
- Mantenibilidad y soporte
- Seguridad
- Legales

# Ejercicio

Enlista cinco requerimientos funcionales y cinco no funcionales para desarrollar un software para la gestión de una farmacia.

# Cierre

Los requerimientos funcionales expresan las acciones que debe realizar el software, mientras que los requerimientos no funcionales aclaran cómo debe comportarse. Ambos son importantes para definir todo lo que se espera del producto del software.



# Bibliografía


- Roberson, S. y Robertson, J. (2012) Mastering the Requirements Process: Getting Requirements Right (3a ed.). EE.UU: Pearson.

Universidad Tecmilenio

# Ingeniería de Software

## Modelos basados en escenarios

Semana 2

# Introducción

En un proyecto de desarrollo de sistemas es necesario que se aclare el alcance y los objetivos del software desde antes de comenzar el diseño y construcción del sistema.

Esto se logra a través de una exhaustiva documentación de las necesidades, requerimientos y expectativas que tiene el cliente, a través de las especificaciones de requerimientos.

# Explicación

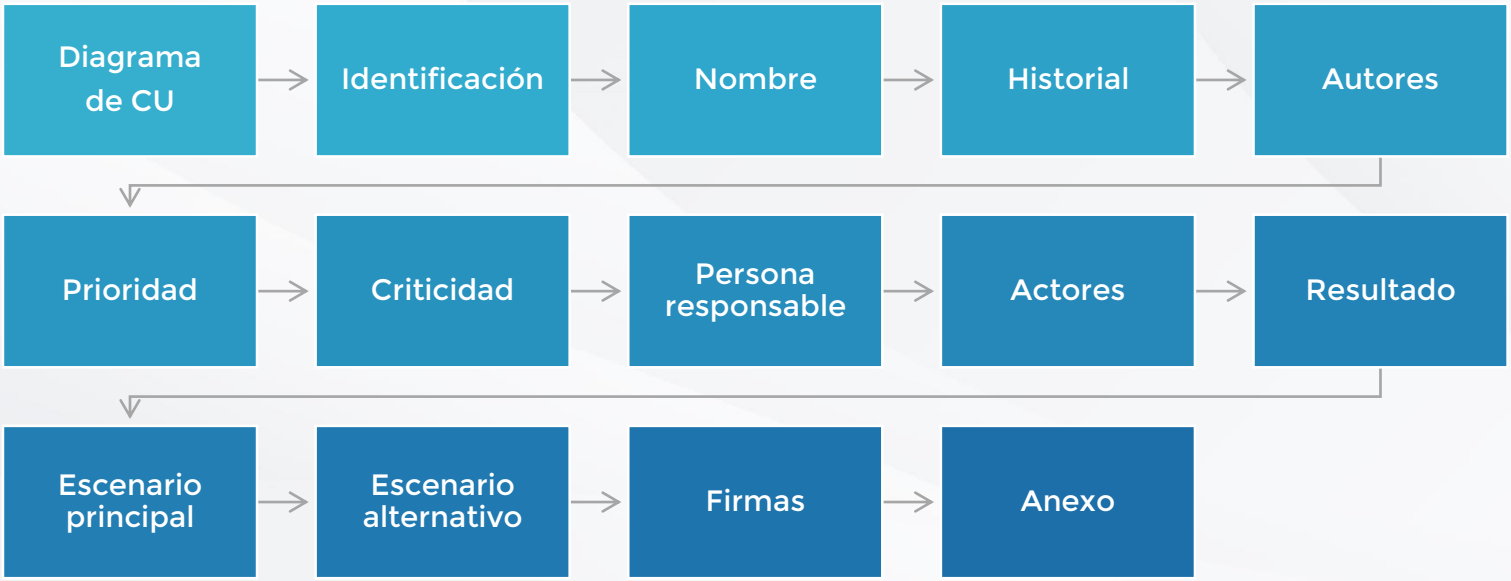
Los casos de uso permiten establecer con claridad la forma en la que interactúan diferentes entidades o actores en un sistema, ya sea ingresando datos o bien recuperando información que servirá para otro proceso del sistema.

# Explicación

Los diagramas de casos de uso permiten establecer: Escenarios en los que existe una interacción entre las personas, organizaciones o sistemas externos con el sistema.

# Explicación

Las secciones de una especificación de caso de uso son:



# Ejercicio

Desarrolla un flujo principal y un flujo de  excepción para un sistema que gestiona la administración de una farmacia.

# Cierre

Los casos de uso son herramientas indispensables en cualquier proyecto de desarrollo de software, pues facilitan la comunicación entre el cliente y el equipo de desarrollo, establecen con claridad cómo abordar los escenarios de interacción entre actores y el sistema, y son piezas clave para el proceso de validación.

# Bibliografía


- Jacobson, I., Christerson, M., Jonsson, P. y Overgaard, G. (1993). Object-oriented software engineering. Wokingham: Addison-Wesley.
- Pohl, K. y Rupp, C. (2011). Requirements engineering fundamentals. EE. UU: Rocky Nook.
- Sommerville, I. (2011). Ingeniería de software (9ª ed.). México: Pearson.
- Tsui, F., Karam, O. y Bernal, B. (2013). Essentials of Software Engineering (3ª ed.). EE. UU: Jones & Bartlett Learning.

Universidad Tecmilenio

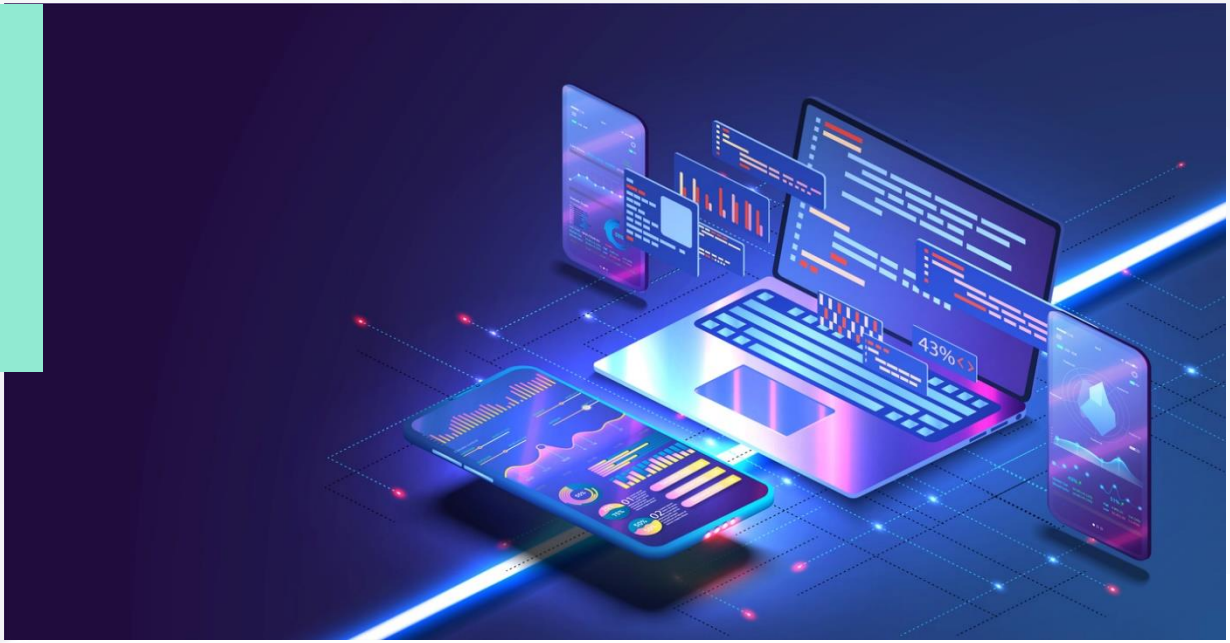
# Ingeniería de Software

## Modelo de clases

Semana 2

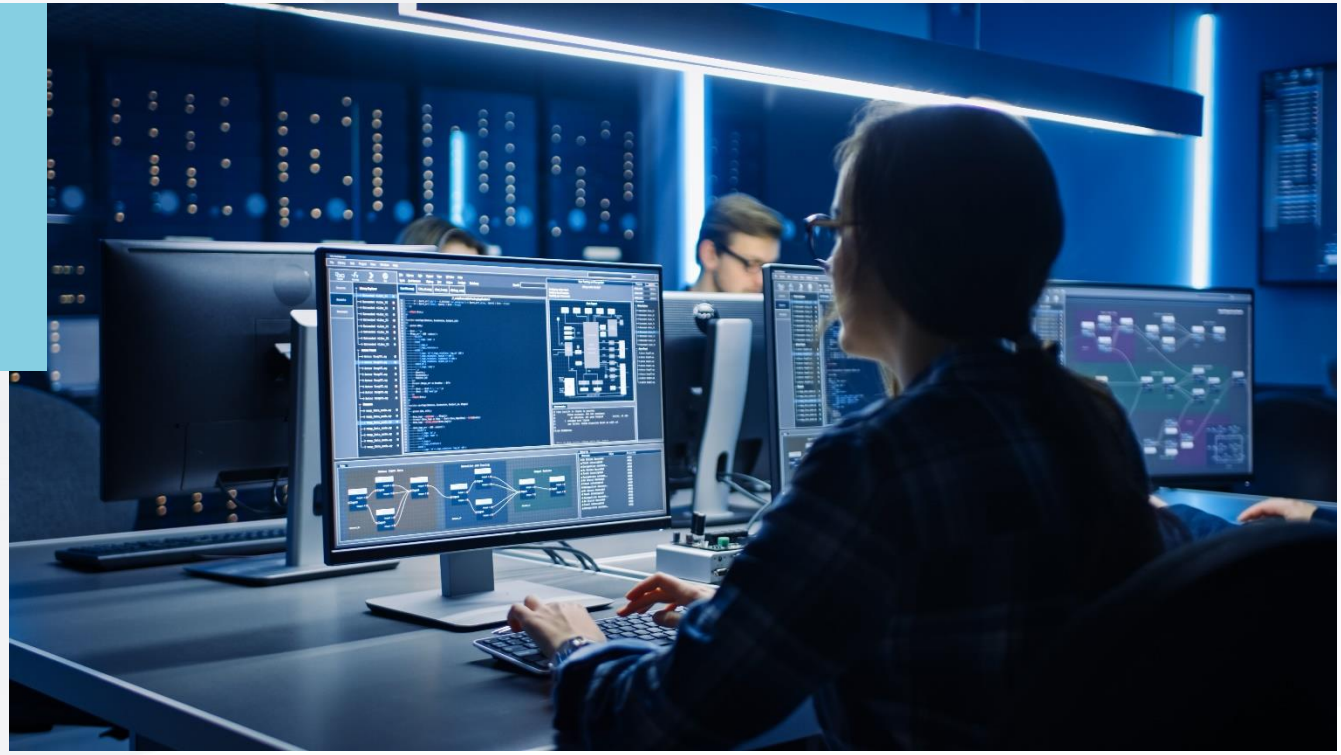
# Introducción

El modelado de clases es un paradigma de la ingeniería de software que permite trabajar con objetos haciendo una analogía del mundo real. Este paradigma puede ser utilizado en la ingeniería de requerimientos para definir las clases que se utilizarán en un contexto específico del negocio y que darán solución a un problema.



# Explicación

Una clase es una colección lógica de atributos de datos que necesita el producto para llevar a cabo sus funciones.



# Explicación

El modelo de clases se inicia con lo que Pressman (2010) denomina como análisis gramatical, es decir, a partir de la descripción de los escenarios de los casos de uso, se irán subrayando los sustantivos que llegarán a ser clases.

## Taxonomía de clases

- Entidades
- Cosas
- Ocurrencias
- Roles
- Unidades organizacionales
- Lugares
- Estructuras

# Explicación

El diagrama de clases se construye utilizando una caja rectangular divida en tres partes o compartimentos

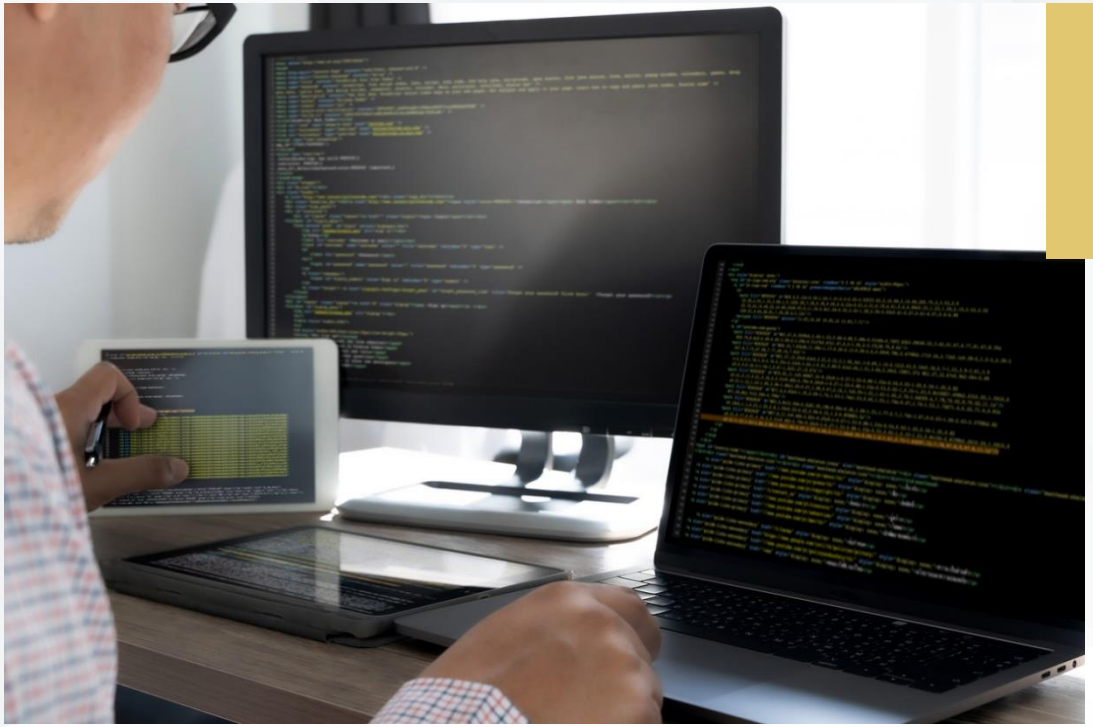
- Clase
- Atributos
- Métodos

# Ejercicio

Desarrollar un diagrama de clases para el flujo de compra de medicamentos.

# Cierre

Los diagramas de clases son herramientas muy útiles como parte de la ingeniería de requerimientos, ya que establecen las partes que abarca el sistema, son sencillos de generar y proveen información que es muy valiosa para el equipo de desarrollo.



# Bibliografía


- Pohl, K. y Rupp, Chris. (2011). Requirements Engineering Fundamentals. EE. UU: Rocky Nook.
- Pressman, R. (2010). Ingeniería de Software. Un enfoque práctico (7ª ed.). México: McGraw-Hill.

Universidad Tecmilenio

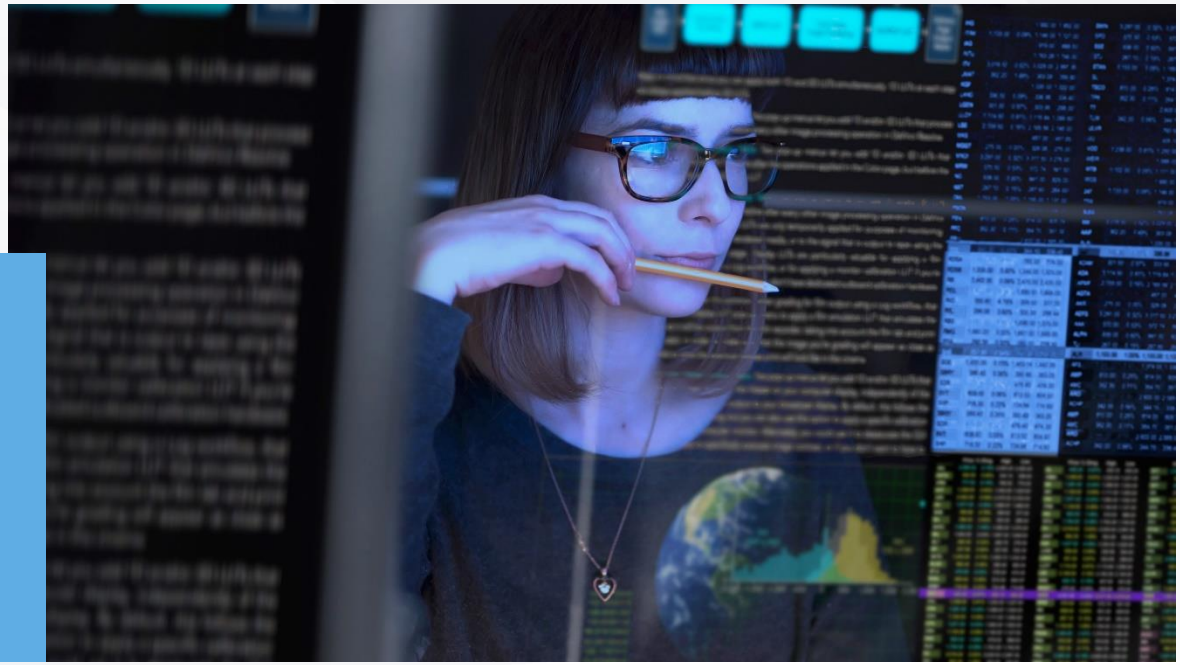
# Ingeniería de Software

## Modelo de flujo de datos

Semana 2

# Introducción

Cualquier aplicación de software que utilice un negocio o una organización que administra información, requiere del ingreso de datos para transformarlos en información.



# Introducción

El modelo de datos y el diagrama de flujo de datos es todo lo que se necesita para obtener una visión significativa de los requerimientos de algunas aplicaciones de software.

# Introducción

Un diccionario de datos funciona de la misma forma como lo hace un diccionario común, es decir, es una colección de palabras ordenadas de forma alfabética y ofrece un significado utilizando sinónimos que puedan ayudar a las personas a entenderlas.

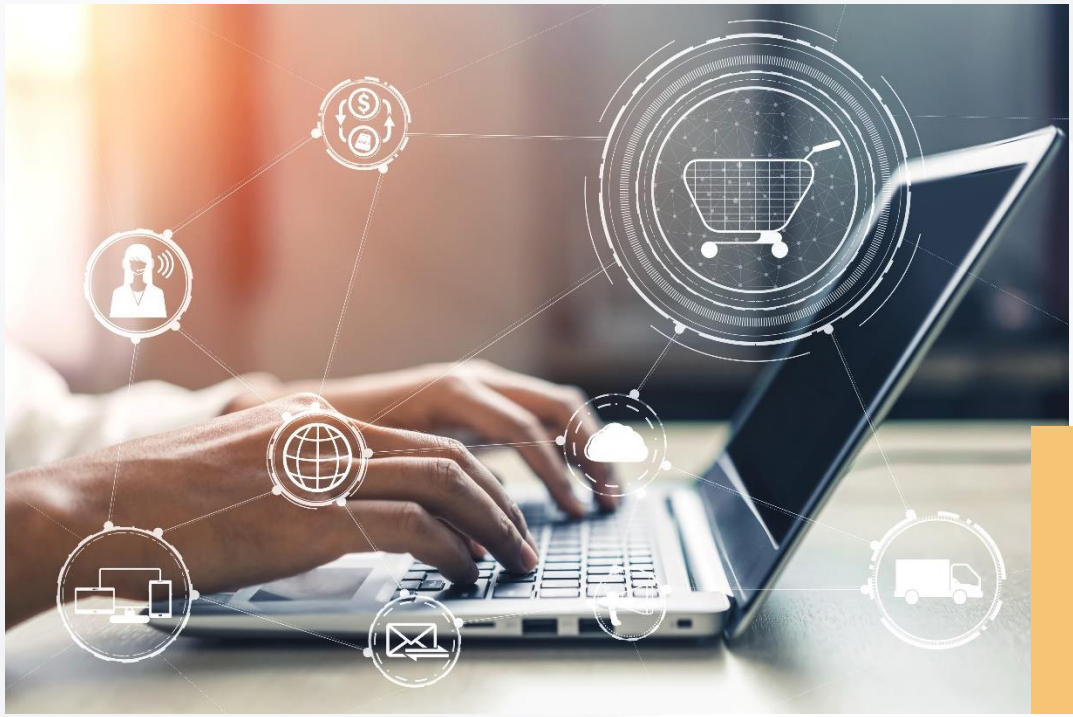
# Introducción

Existen diferentes tipos de elementos según el tipo de datos que se describen a continuación:

| | | |
|---|---|---|
| *Bit o Boolean* | Carácter | Fecha |
| Numérico (entero) | Numérico (decimal) | Moneda |

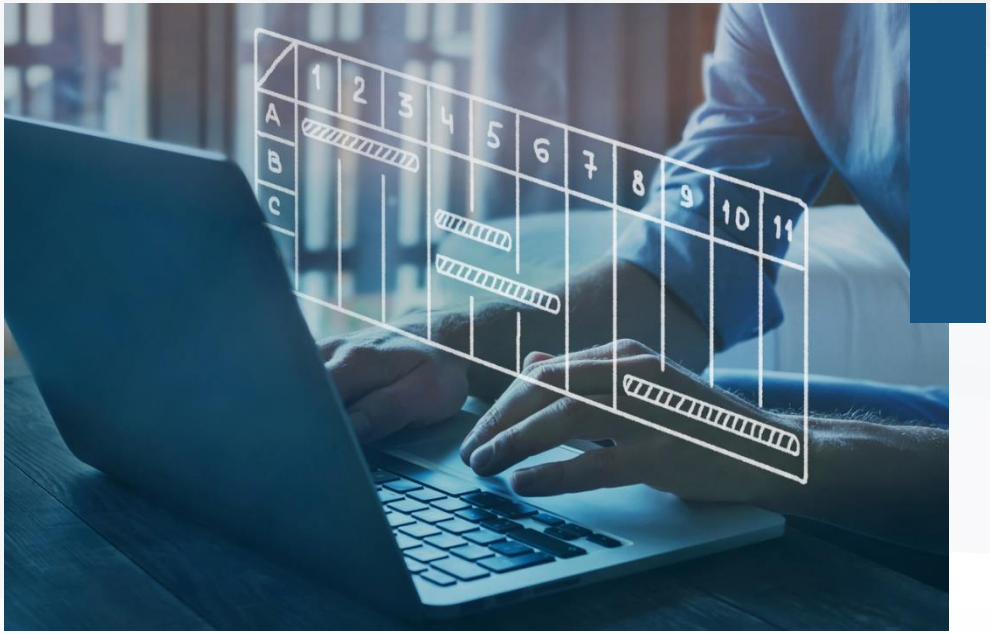
# Ejercicio

Desarrolla un diagrama de flujo de datos del proceso de compra de medicamentos.



# Cierre

El modelado de sistemas a través de diagramas de flujo de datos y diccionario de datos, es una estrategia de la ingeniería de requerimientos para aclarar las necesidades de información del usuario. Al dibujar los diagramas es posible identificar huecos de información que requieren de mayor detalle o de una definición más clara que permita abordar el problema de la mejor manera.



# Bibliografía


- Pohl, K. y Rupp, Chris. (2011). Requirements Engineering Fundamentals. EE. UU: Rocky Nook.
- Pressman, R. (2010). Ingeniería de Software. Un enfoque práctico (7ª ed.). México: McGraw-Hill.