

Diseño de software

☰ Tags	UNIDAD 3
📅 Date	@27/04/2023 → 09/05/2023
# Notas	

IRAT

Diseño:

Definiciones:

- **Sistema:** una entidad logica que tiene un conjunto de responsabilidades u objetivos y puede estar compuesto por hw y sw
- **Subsistema:** un: sistema que es parte de un sistema mayor. Tiene una interfaz bien definida
- **Componentes:** Puede ser reutilizado, cualquier pieza de hw o sw. lo saco y lo pongo en cualuqeir lugar y puede seguir funcionando
- **Modulo:** componente que es definido a nivel de programación pakage, conjunto de clases.

TA 1-

UT3-TA1

- 1 - Sistema Operativo → Sistema
- 2 - Browser → Componente^x - Sistema
- 3 - Microservicio de Pago → Subsistema
- 4 - Compilador → Módulo^x - Componente
- 5 - Biblioteca de Criptografía → Componente
- 6 - Controlador de Dispositivo → Subsistema^x - Componente
- 7 - Servidor Web → Subsistema
- 8 - Base de Datos SQL → Componente^x - Sistema
- 9 - Middleware → Componente
- 10 - ORM → Módulo
- 11 - IDE → Componente^x Sistema
- 12 - Motor de Renderizado →
- 13 - Procesador de Texto →
- 14 - Gestor de Notificaciones →
- 15 - Gestor de Ventanas → Subsistema

EL PROCESO DE DISEÑO DE SW

Distintos tipos de diseño:

- Diseño de arq. se define la arq. que se va a utilizar (MVC, monolitico, en capas)
- Diseño de interfaz. como se va a comunicar con otros de los subsistemas.
- Diseño de los componentes: Sirve para identificar como se relaciona nuestro sistema

Si estuviéramos haciendo un diagrama de componentes de modulos. tiene una dependencia con el modulo de operaciones que tiene RRHH y contabilidad

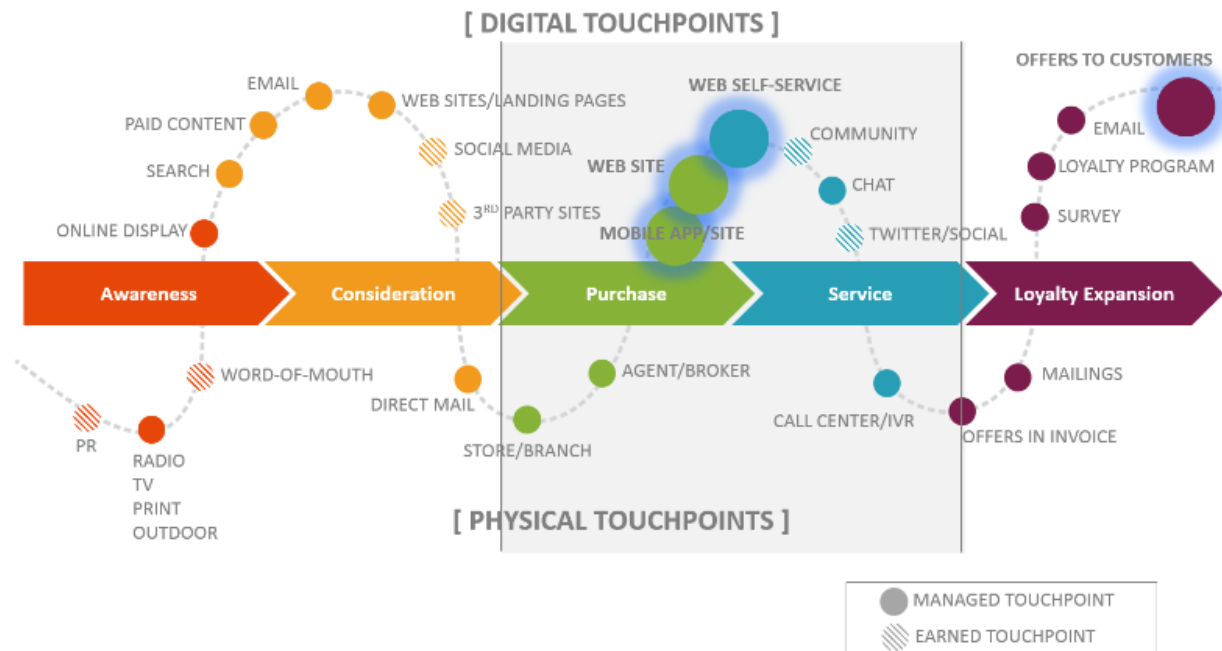
Principios de diseño:

- Dividir y conquistar:
- Incrementa la cohesión: que todo lo que tiene que ver este junto
- Reducir el acoplamiento: que lo que no tiene que ver este lejos. tener las dependencias necesarias
- Mantener el nivel de abstracción tan alto como sea posible: utilizar interfaces o clases no específicas

Cosas a tener en cuenta cuando hacemos diseño

- Incrementar la reusabilidad: por ejemplo el pago, esto lo puedo sacar y definir un producto nuevo.
- Diseñar para ser flexible: es difícil este, que es flexible, no nos podemos adelantar tanto. Pero si estamos diseñando un sistema desde cero, utiliza código para definir la moneda. Peso argentino que paz descanse
- Anticipar la obsolescencia: 80% del producto es el mantenimiento del producto. Y en el medio entre producción y el tiro

Journey Map



- El cliente descubre que hay un producto
- El cliente considera comprarlo
- El cliente decide comprarlo
- POr venta, como hago que el usuario siga utilizando mi app
- Como hago que más usuarios la usen

TA 2