

# Universidad Autónoma Metropolitana

## Unidad Azcapotzalco

### Tarea 1. Comparativa de Arquitecturas.

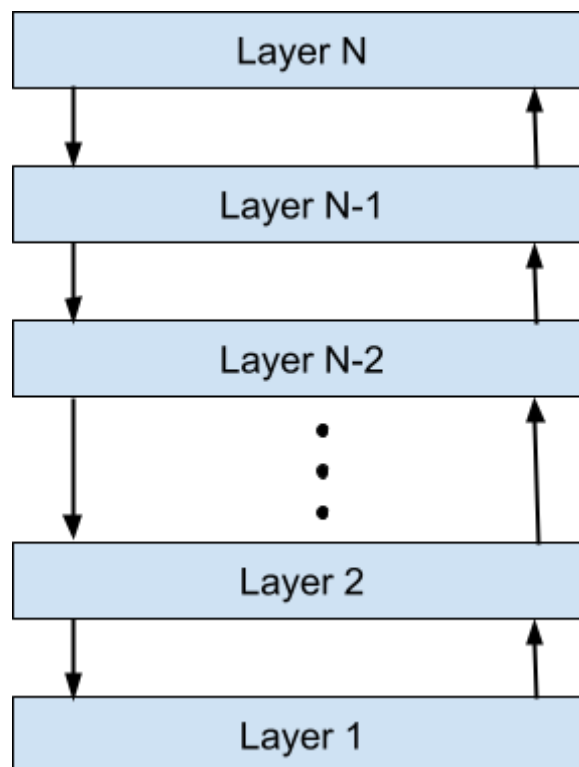
Nombre del alumno: García Godoy Víctor Saúl

Matrícula: 2183042802

#### Arquitectura en capas.

##### Características y descripción.

Es una de las arquitecturas que predominan hoy en día en las aplicaciones empresariales. Esta arquitectura considera al sistema como un conjunto de capas interconectadas. Para acceder a estas capas, es necesario realizar peticiones entre ellas. Ninguna capa puede pasar a la capa inmediata inferior. A continuación se muestra una imagen de esta arquitectura.



#### Ventajas y desventajas de esta arquitectura.

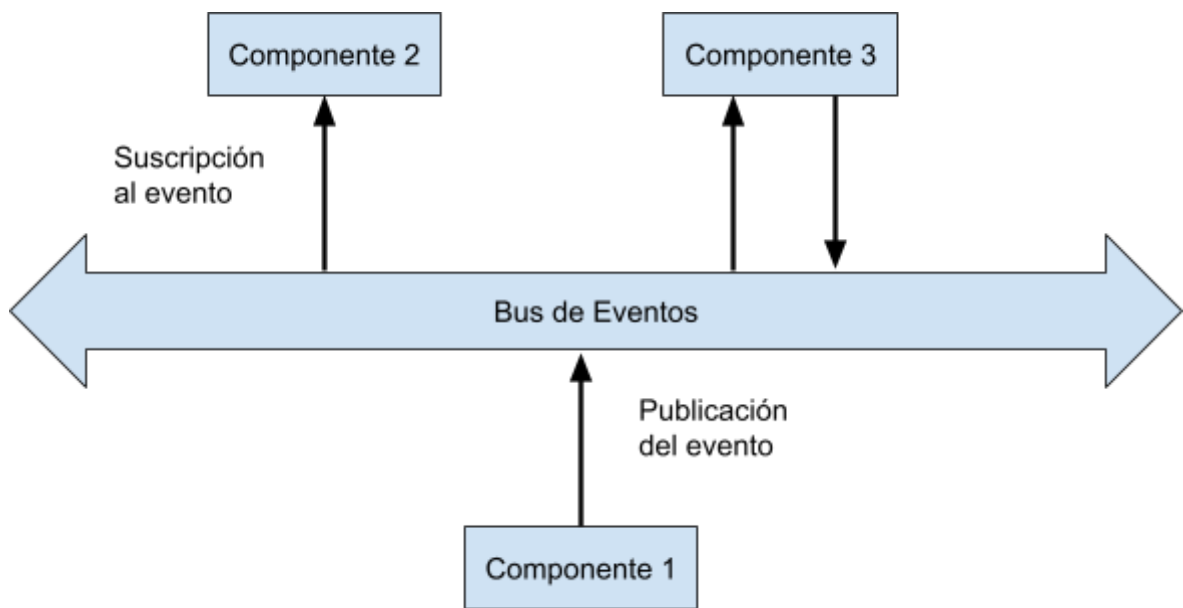
Ventajas	Desventajas
<ul style="list-style-type: none"><li>• Es una arquitectura fácil de mantener, ya que cada capa puede actualizarse sin alterar a las demás.</li><li>• Las capas pueden ser puestas a prueba de forma independiente.</li><li>• La arquitectura es modular ya que cada capa tiene una función específica.</li></ul>	<ul style="list-style-type: none"><li>• Si el sistema es complejo, puede llegar a ser complicado el acceso a una determinada capa, pues habrá que pasar una gran cantidad de capas previas para acceder a la información.</li><li>• Dependencia entre capas. Las capas pueden llegar a tener un alto grado de dependencia y</li></ul>

<ul style="list-style-type: none"> <li>Control de acceso. Para acceder a una capa N es necesario hacerlo desde una capa anterior. Esto brinda seguridad en la arquitectura.</li> </ul>	<p>si una falla la comunicación en el sistema puede llevar a fallas.</p> <ul style="list-style-type: none"> <li>El rendimiento de la arquitectura depende de la cantidad de capas.</li> <li>Al agregar más capas se incrementa el riesgo de cometer un error.</li> </ul>
--	--

### Arquitectura basada en eventos.

#### Características y descripción.

Esta arquitectura consta de un conjunto de componentes interconectados a través de un *pipeline* o bus de eventos. Cada componente del sistema es capaz de publicar un evento. A su vez, cada componente es capaz también de suscribirse a dicho evento emitido.



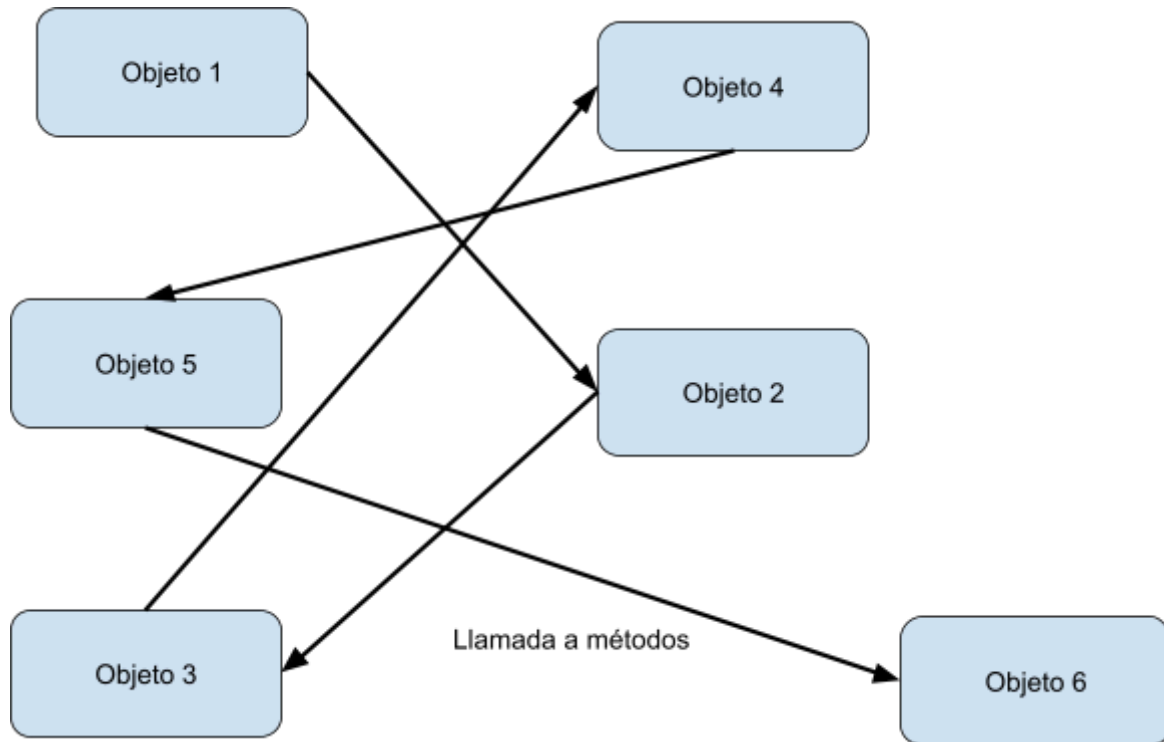
#### Ventajas y desventajas de esta arquitectura.

Ventajas	Desventajas
<ul style="list-style-type: none"> <li>Los componentes no dependen directamente unos de otros.</li> <li>Alta escalabilidad. Se pueden añadir nuevos receptores sin afectar la arquitectura.</li> <li>Permite el desacoplamiento de servicios.</li> <li>La comunicación puede ser hecha a través de los eventos en lugar de utilizar la referencia al componente u objeto.</li> </ul>	<ul style="list-style-type: none"> <li>Puede ser complicada la gestión de eventos si existen demasiados componentes.</li> <li>El flujo de eventos puede ser complicado de seguir.</li> <li>Tiene una mayor complejidad en la gestión de errores.</li> <li>No es ideal si la cantidad de eventos es mayor que la cantidad de suscriptores.</li> </ul>

## Arquitectura basada en objetos.

### Características y descripción.

En la arquitectura basada en objetos, cada componente del sistema es visto como un objeto, con atributos y métodos. La transmisión de mensajes entre estos componentes es a través de la invocación de los métodos de cada objeto. Los componentes son independientes y ofrecen diferentes funcionalidades.



### Ventajas y desventajas de esta arquitectura.

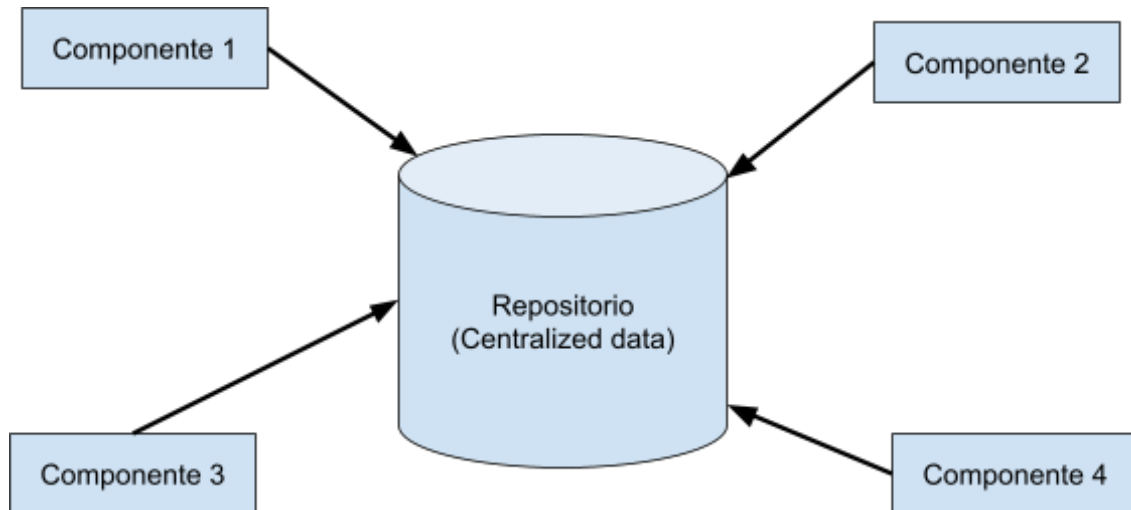
Ventajas	Desventajas
<ul style="list-style-type: none"><li>• Alta modularidad dada la utilización de objetos.</li><li>• Tiene una escalabilidad alta. Es posible agregar nuevos objetos sin alterar la arquitectura existente.</li><li>• Abstracción. Los objetos pueden representar entidades complejas de manera simple.</li><li>• Distribución de tareas y asignación de trabajo a cada objeto.</li></ul>	<ul style="list-style-type: none"><li>• Saturación de los objetos si la comunicación entre ellos es muy frecuente o compleja.</li><li>• Riesgo de sobrecarga de memoria. Debido a la creación de múltiples instancias de objetos.</li><li>• Si el sistema es grande, la complejidad de la arquitectura aumenta considerablemente pues aumenta la cantidad de objetos y el llamado a métodos.</li></ul>

## Arquitectura centrada en datos.

### Características y descripción.

La arquitectura centrada en datos consta de un repositorio principal, al cual, los componentes pueden acceder o hacer modificaciones. El principal objetivo de esta arquitectura es la integridad de los datos contenidos en el repositorio, pues estos deben ser consistentes. Los componentes interactúan solamente a través de los

accesos al repositorio. Existen dos componentes principales: El repositorio y el componente que accede al repositorio.



Ventajas y desventajas de esta arquitectura.

Ventajas	Desventajas
<ul style="list-style-type: none"><li>● Persistencia y consistencia de los datos. Los componentes acceden a los datos del repositorio y, si uno cambia, los demás componentes, podrán acceder al mismo dato que ha sido modificado.</li><li>● Fácil integración de nuevos componentes al tener un repositorio común.</li><li>● Es de fácil acceso pues es un sólo repositorio central de datos.</li><li>● Es barata y fácil de mantener.</li></ul>	<ul style="list-style-type: none"><li>● Fuerte dependencia al repositorio central. Si el repositorio central falla, todo el sistema se verá afectado.</li><li>● Complejidad en gestión de concurrencia. Cuando múltiples componentes intentan acceder de forma simultánea a los datos.</li><li>● Flexibilidad baja, debido a la dependencia con el repositorio central.</li></ul>