



**EDUCACIÓN  
EN LÍNEA**



# Computación Paralela y Distribuida

---

Modelo vista controlador (MVC)

Integrantes:

- Anderson Lucero
- Diego Vaca
- Cristian Quilumbaquin

# Modelo vista controlador (MVC)

---

## Modelo vista controlador (MVC)

- Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.
- Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

# Modelo vista controlador (MVC)

---

## Modelo vista controlador (MVC)

- El **Modelo** que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
- La **Vista**, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.
- El **Controlador**, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.

# Modelo vista controlador (MVC)

---

El modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Define las reglas de negocio .Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".

# Modelo vista controlador (MVC)

---

El modelo es el responsable de:

- Lleva un registro de las vistas y controladores del sistema.
- Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo.

# Modelo vista controlador (MVC)

---

El controlador es responsable de:

- Recibe los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contiene reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar()". Una petición al modelo puede ser "Obtener\_tiempo\_de\_entrega (nueva\_orden\_de\_venta)".

# Modelo vista controlador (MVC)

---

Las vistas son responsables de:

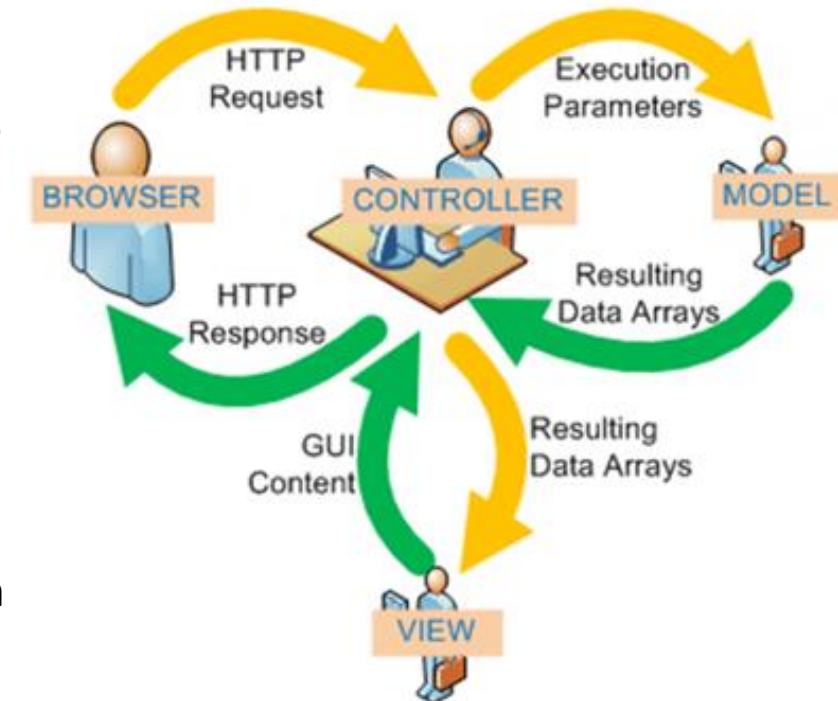
- Recibir datos del modelo y los muestra al usuario.
- Tienen un registro de su controlador asociado (normalmente porque además lo instancia).
- Pueden dar el servicio de "Actualización()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).



# Modelo vista controlador (MVC)

## Las vistas son responsables de:

- El flujo que sigue el control generalmente es el siguiente:
- El usuario interactúa con la interfaz de usuario de alguna forma.
- El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.
- El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario (por ejemplo, el controlador actualiza el carro de la compra del usuario).
- El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se refleja los cambios en el modelo (por ejemplo, produce un listado del contenido del carro de la compra).
- La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.



# Modelo vista controlador (MVC)

---

## MVC en aplicaciones web

- Vista:
  - ❖ La página HTML
- Controlador:
  - ❖ Código que obtiene datos dinámicamente y genera el contenido HTML
- Modelo:
  - ❖ La información almacenada en una base de datos o en XML
  - ❖ Junto con las reglas de negocio que transforman esa información (teniendo en cuenta las acciones de los usuarios)

# Modelo vista controlador (MVC)

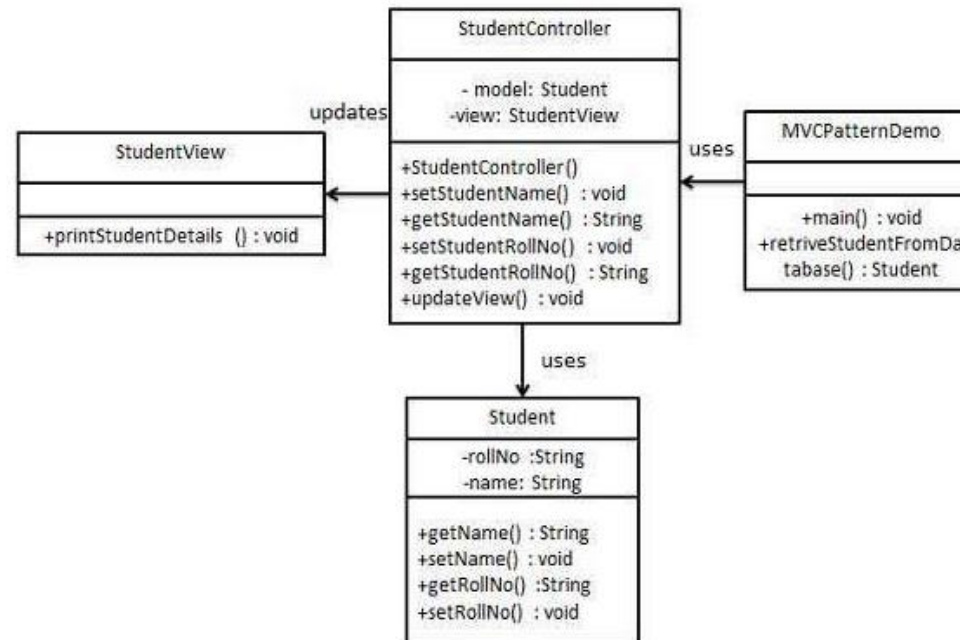
---

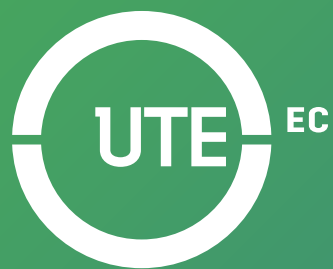
## MVC en Java Swing

- Modelo:
  - ❖ El modelo lo realiza el desarrollador
- Vista:
  - ❖ Conjunto de objetos de clases que heredan de `java.awt.Component`
- Controlador:
  - ❖ El controlador es el thread de tratamiento de eventos, que captura y propaga los eventos a la vista y al modelo
  - ❖ Clases de tratamiento de los eventos (a veces como clases anónimas) que implementan interfaces de tipo `EventListener` (`ActionListener`, `MouseListener`, `WindowListener`, etc.)

# Modelo vista controlador (MVC)

## Ejemplo Estudiantes:





**¡GRACIAS!**

**TRAS  
CENDE  
MOS**

A white curved line graphic, resembling a stylized 'C' or a partial arc, positioned to the right of the text.