

# Modelo Vista Controlador (MVC)

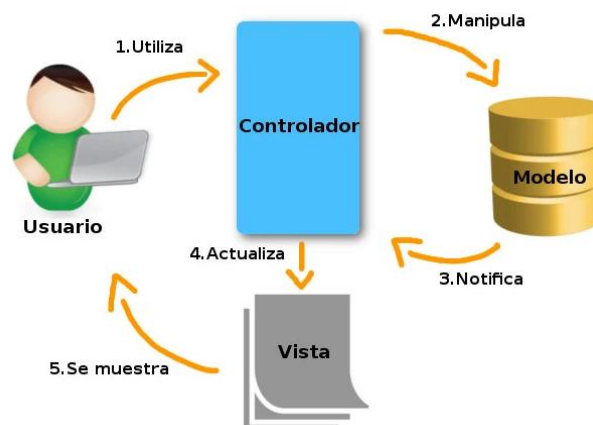
## *(model view controller)*

El patrón de modelo de vista controlador es un patrón de arquitectura de software que tiene como función el organizar y estructurar el código de una aplicación, este patrón separa los datos del negocio de una aplicación, la interfaz de usuario y la lógica de control de negocio en tres componentes distintos. Es una arquitectura importante ya que se utiliza tanto en componentes gráficos básicos hasta sistemas empresariales; la mayoría de los frameworks modernos utilizan MVC o alguna adaptación del mismo para la arquitectura.

El **Modelo** que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia de datos. Generalmente se encarga de acceder a la capa de almacenamiento de datos para realizar modificaciones en ella, como por ejemplo actualizaciones, consultas, búsquedas, etc. Se encarga de definir las reglas del negocio (la funcionalidad del sistema) y también se encargará de llevar un registro de vistas y controladores del sistema.

La **Vista** o interfaz de usuario es responsable de mostrar los datos al usuario y de proporcionar una forma para que el usuario interactúe con la aplicación. Es la representación visual de los datos, es decir, en este componente va todo lo que tenga que ver con el diseño gráfico de la aplicación, su única función es determinar cómo se verán los datos de cara al usuario.

El **Controlador** que actúa como intermediario entre el modelo y la vista, gestionando el flujo de información entre ellos, el controlador recibe los eventos de entrada, es decir, las órdenes del usuario por medio de la vista, se lo comunica al modelo y devuelve la información a la vista para que actualice su interfaz gráfica con los nuevos datos.



Ejemplo de cómo funciona el mvc.

1. El usuario interactúa con la interfaz de usuario de alguna forma, por ejemplo, una persona quiere pedir comida a domicilio en su celular por medio de una aplicación como uber eats, entonces la persona elige su comida y pone su método de pago y realiza la compra.
2. El controlador recibe la notificación de la acción solicitada por el usuario, en este caso de una compra de un alimento. El controlador gestiona el evento que llega.
3. El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario
4. 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.
5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.