

# Taller 2

API REST

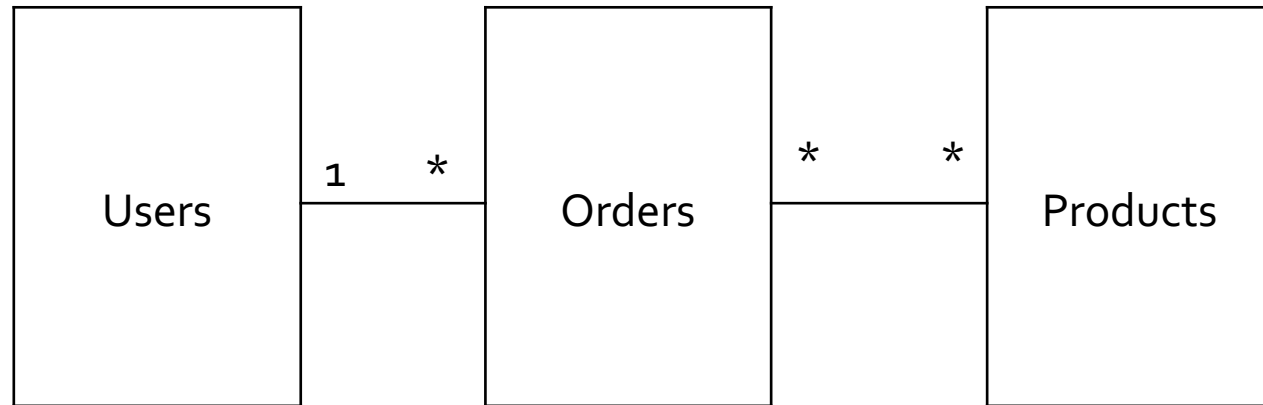
# Objetivo

- ▮ Desarrollar un API Rest y desplegar el servicio en la Web
- ▮ Generar diversos endpoints para implementar una funcionalidad a partir de un modelo
- ▮ Usar una base de datos para almacenar los datos necesarios para garantizar el funcionamiento del servicio

# Consigna

- Almacenes la 14 es una empresa naciente que busca implementar una pasarela de pagos. Lo contactan para que usted construya un modelo mínimo viable que le permita implementar la funcionalidad de ***Carrito de Compras***.
- Esta funcionalidad NO TIENE aspecto gráfico de ningún tipo. Debe generar los servicios necesarios para soportar la funcionalidad, de modo que cualquier front pueda usar estos servicios para implementar la sección gráfica del carrito.

# Modelo



▮ *Este es el modelo sugerido.*

▮ El usuario debe crearse usando al menos su cédula.

▮ El usuario puede generar varias órdenes. Estas órdenes tienen información de la fecha de creación, si está pagado o no y cuál es la fecha de pago (use Unix timestamp para almacenar fechas). Estas órdenes representan el carrito de compras.

▮ Cada orden puede tener varios productos y cada producto puede estar presente en varias ordenes.

▮ Agregue las variables necesarias para soportar los requerimientos.

# Requerimientos

- ▮ Creación de usuarios, órdenes y productos. Para la creación de usuarios y productos no tiene ninguna restricción. Para crear una orden, necesita crear antes el usuario al que pertenece esta orden.
- ▮ Método para agregar un producto a un orden. Este método tiene como parámetro de entrada la cantidad de este producto a introducir al carrito.
- ▮ Método para eliminar un producto de la orden. Este método tiene como parámetro de entrada la cantidad a quitar del producto a eliminar.
- ▮ Método para cambiar el estado de la orden, de no pagado a pagado.
- ▮ Método que permite ver la información de una orden por ID. Debe devolver la información completa de cada producto, la cantidad de cada uno, el número de elementos del carrito y el precio total de la orden.
- ▮ Método para ver el histórico de órdenes de un usuario por cédula.
- ▮ El servicio está desplegado en la Web.

\* Todos los puntos valen lo mismo

# Requerimientos

- ▮ Todas las respuestas devuelven un JSON, incluso los métodos de orden (POST, PUT, DELETE).
- ▮ Use la base de datos de Icesi. Use su código de Icesi como prefijo para cualquier tabla que cree allí.