

## Respuesta al problema2, tienda en línea de artesanías.

Para resolver este problema se considera contestar las preguntas específicas que se realizan, para lo cual se entrega el presente documento.

En forma general intenté tocar un poco de todos los elementos necesarios en la generación de los proyectos.

Cabe recalcar que no este documento dista mucho de ser un documento completo de proyecto e implementación, debido a que aún faltan más especificaciones de los casos de uso, la historia del usuario, diseño de interface, generación de reportes, cálculo de PKI, administración del almacén o almacenes , etc.

Además falta el documento de la implementación que debe definir las características de la implementación del sistema

Al final del documento se encuentran los glosarios de servicios y actores.

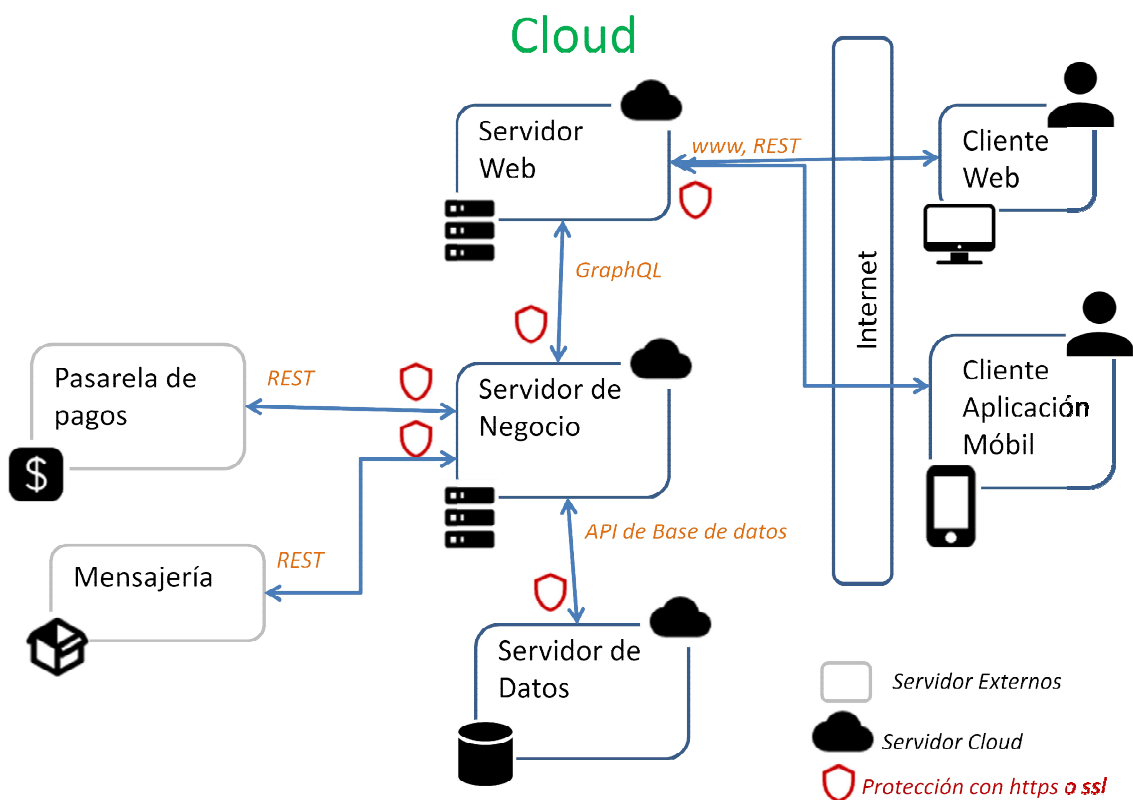
# Diseño General

Se plantean dos soluciones, una con servidores *Cloud* y otro *On premise*, la elección dependerá de las características como: inversión del cliente, número de usuarios, tamaño de datos a usar, etc.

Si yo tuviera que hacer la elección, pensaría en servidores *Cloud* ya que si el tráfico es muy grande pueden agregarse varios "servidores de Negocio" para atender la carga.

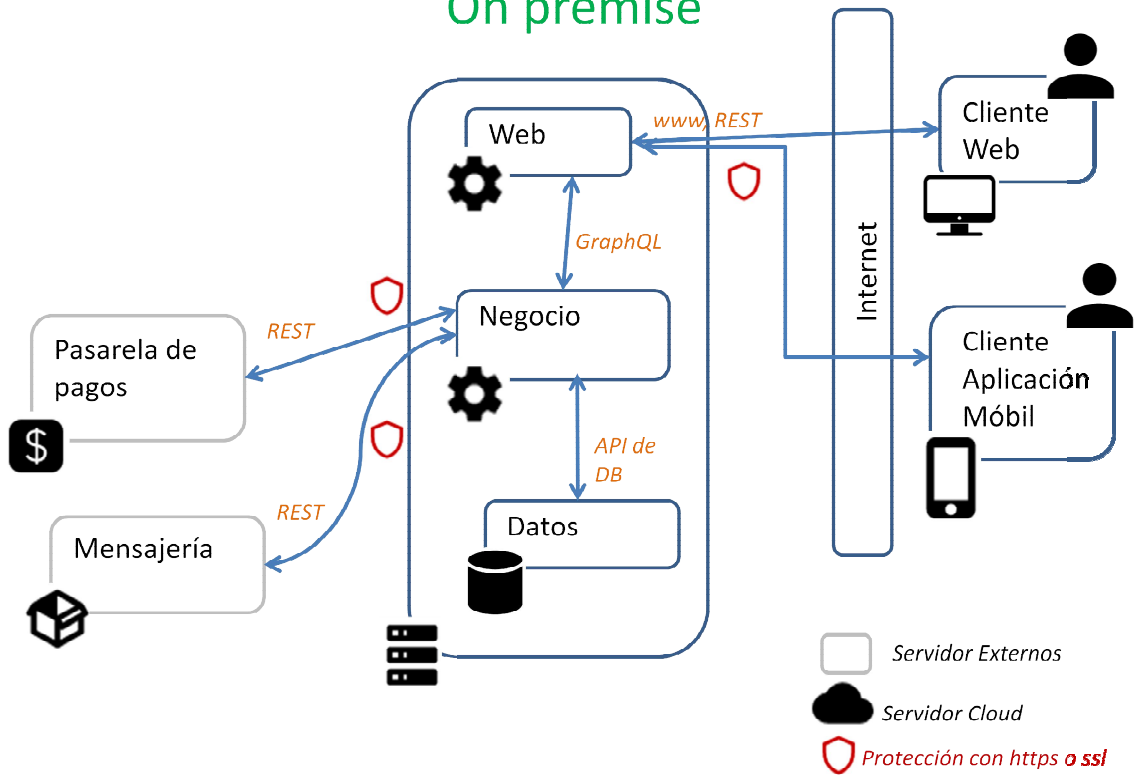
En ambos casos todas las conexiones de los usuarios deben pasar por el "servidor Web" para hacer un proxy, prevenir ataques DOS y distribuir la carga, en caso de que el volumen de solicitudes sea muy grande.

## Diagrama general de tienda en línea de artesanías



# Diagrama general de tienda en línea de artesanías

On premise



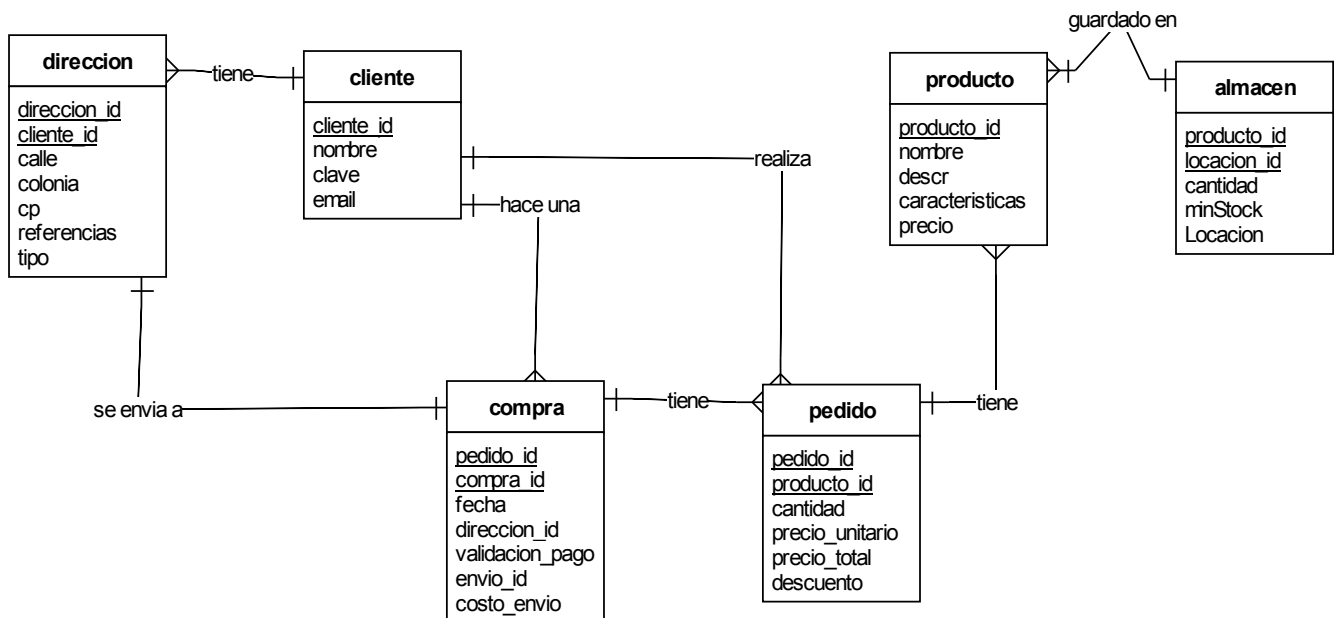
# Diseño de la base de datos

Se considera el uso de una base de datos relacional, pero puede ser implementada en cualquier base de datos NoSql, como MongoDB.

Por simplicidad de tienen las siguientes consideraciones:

- Sólo hay un almacén, el cual está dividido en "locaciones" que pueden utilizarse para cualquier producto.
- Todos los productos de un mismo tipo pueden estar en una sola "locación".
- Todos los envíos se realizan mediante la misma compañía de mensajería.

Diagrama ER de la base de datos relacional.



# Casos de uso

## *Creación de nuevos pedidos*

### **Actores**

Usuario, Servicio Web, Servicio de Negocio, Servicio de Datos, Servicio de Pagos, Servicio de Mensajería.

### **Precondiciones**

El usuario ha accedido a la *Plataforma* y ha agregado los productos a comprar en el "carrito de compras".

### **Desarrollo**

La *Plataforma* recibe los pedidos, se realiza el pago de los productos y se contacta a mensajería para que recoja el pedido.

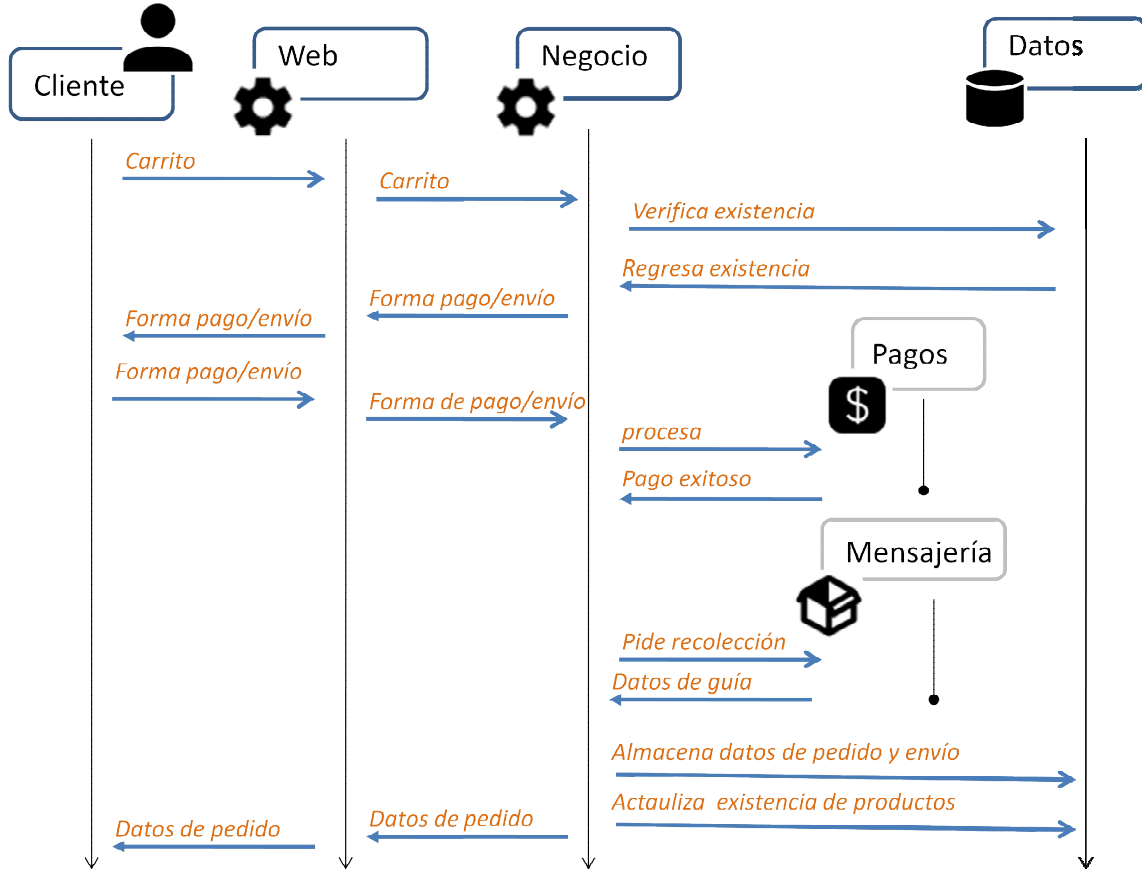
### **Post condición**

El usuario recibe los datos del pedido procesado, así como datos de envío.

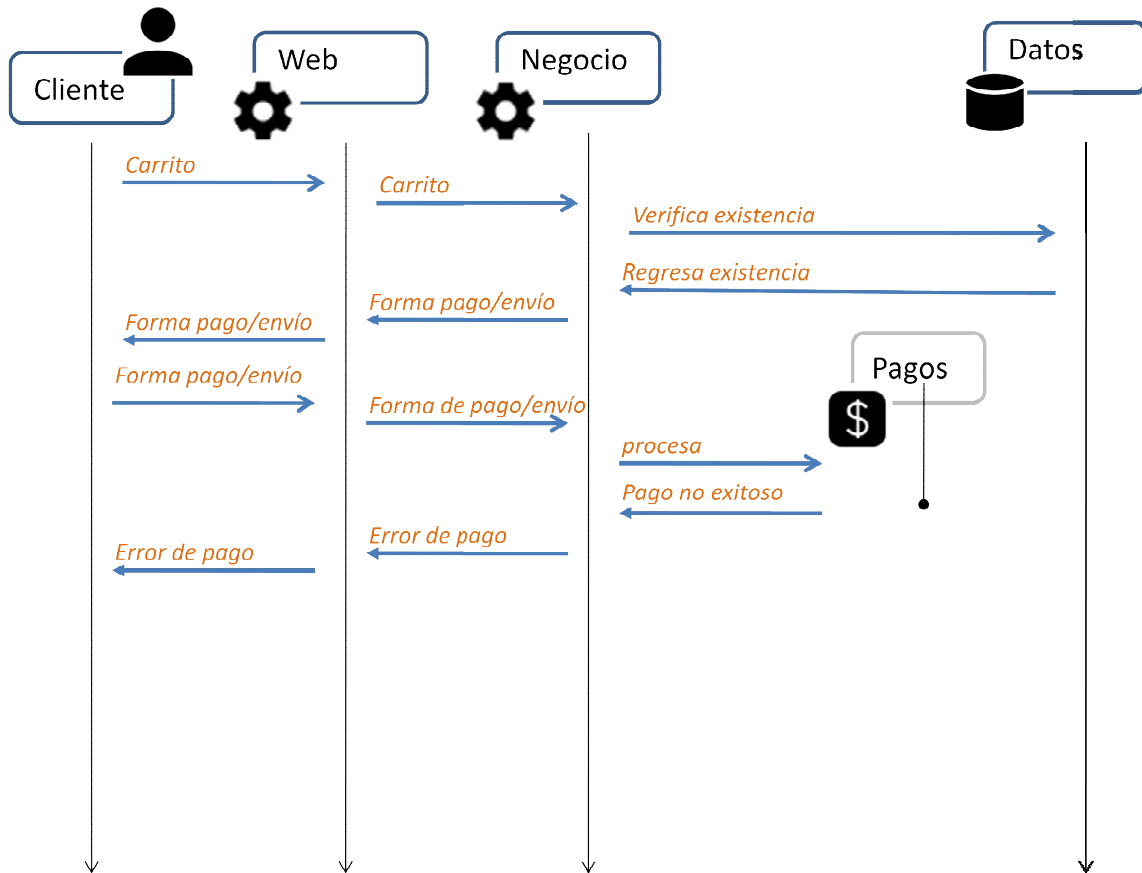
### **Resultado alternativo**

Si se no procede el pago y/o envío o no se puede conectar con los servidores de estos servicios, debe regresar error.

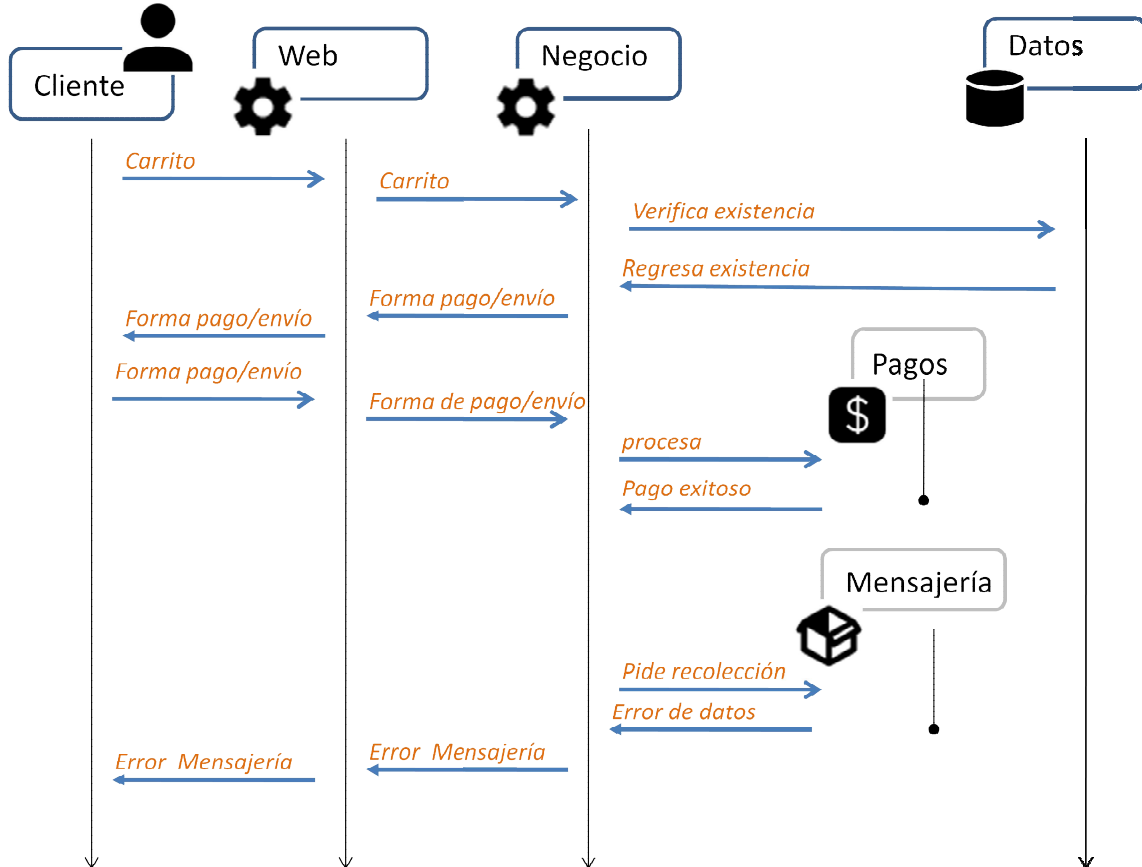
## Proceso de nuevo pedido – Resultado Exitoso



## Proceso de nuevo pedido. Error de pago



## Proceso de nuevo pedido – Error en Mensajería



## Manejo de inventarios

### Actores

Servicio de Negocio, Servicio de Datos.

### Precondiciones

El servicio de Negocio ha lanzado un evento de bajo stock para el producto X.

### Desarrollo

Cuando se realiza un pedido el servicio de Negocios verifica la existencia de los productos de la compra, si alguno de los productos tiene stock por debajo del mínimo, entonces solicita resurtir el producto.

### Post condición



Se envía un correo electrónico con las.

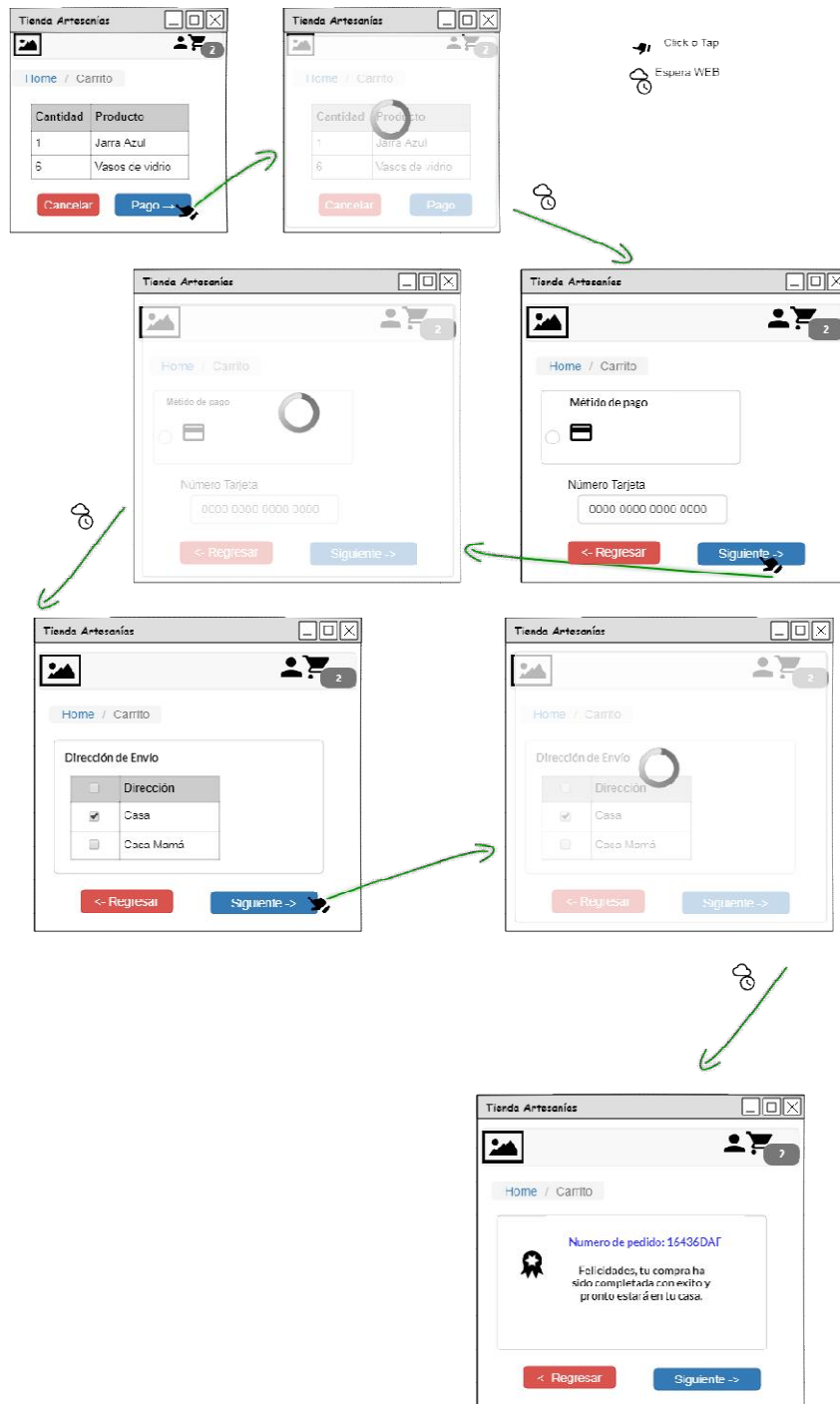
**Resultado alternativo**

No aplica.

# Diagramas de UI

## Diagrama de flujo de datos de un pedido

UI. Flujo de colocación de un Pedido



# Glosario

**Sistema.** El producto que será entregando

## Glosario de actores

**Usuario.** Persona que accede al Sistema con el fin de comprar productos.

**Administrador.** Persona que accede al sistema con fines de administración del Sistema.

**Servicio Web.** El un servicio (no importa si es Cloud u On premise) que se encarga de recibir las peticiones de los usuario para ser procesadas por el Servicio de Negocio. Este servicio también se encarga de proveer de protección de los datos mediante SSL y del proxy para prevenir ataques DOS.

**Servicio de Negocio.** Este servicio (no importa si es Cloud u On premise) se encarga de hacer controlar la lógica del negocio, como es, tipo de compras, promociones, control de compras y entregas, etc.

**Servicio de Datos.** Este servicio (no importa si es Cloud u On premise) se encarga de almacenar la información de los Sistema de forma permanente. En este contexto no importa si es una base de datos Relacional o un sistema NoSql.

**Servicio de Pagos.** Este servicio es proveído por un tercero que permite procesar el cobro de los productos vendidos.

**Servicio de Mensajería.** Este servicio es proveído por un tercero y permite organizar la recolección y entrega de mensajería por parte del proveedor externo.