



Universidad Nacional de Lanús

**DESARROLLO DE SOFTWARE EN SISTEMAS
DISTRIBUIDOS**

Mensajería con Apache Kafka

Departamento: Desarrollo Productivo y Tecnológico

Carrera: Licenciatura en Sistemas

Año: 2024

Cuatrimestre: 2° Cuatrimestre

Docentes:

**Ing. Diego Andrés Azcurra
Lic. Marcos Amaro**

Alumnos:

Fernando Jose Luis Camacho	(DNI: 35.459.731)
Franco Alberto Astorga	(DNI: 42.359.479)
Franco Ariel Figueroa	(DNI: 42.684.291)
Juan Leandro Gauna	(DNI: 31.654.945)

Índice

<u>1. Código fuente</u>	3
<u>2. Roles y tareas desempeñadas</u>	3
<u>3. Estrategia de resolución</u>	3
<u>3.1. Base de datos</u>	3
<u>3.2. Servidor</u>	5
<u>3.3. Apache Kafka</u>	5
<u>3.4. Interfaz gráfica</u>	6
<u>3.5. Ejecutables</u>	7
<u>4. Pruebas realizadas</u>	7
<u>5. Pendiente del primer trabajo práctico</u>	7

1. Código fuente

Enlace al repositorio público de Github del segundo trabajo práctico:

<https://github.com/Franco2900/proveedorKafkaGrupo5.git>

2. Roles y tareas desempeñadas

- Desarrollo de la base de datos y del backend del servidor del proveedor: Franco Figueroa
- Desarrollo del frontend del servidor del proveedor: Franco Astorga
- Desarrollo de los cambios necesarios en el primer trabajo práctico: Franco Astorga, Fernando Camacho, Leandro Gauna y Franco Figueroa

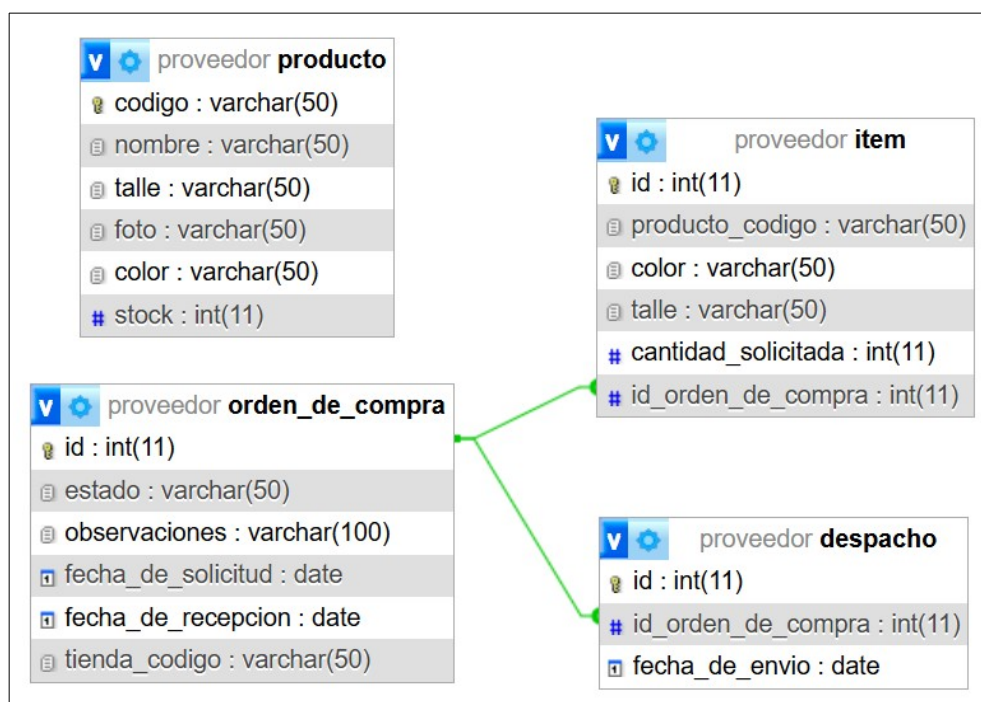
3. Estrategia de resolución

3.1. Base de datos

Como el sistema del proveedor se levanta como un proyecto aparte, se hizo una nueva base de datos SQL llamada “proveedor” para el nuevo proyecto. Dicha base de datos se usa para almacenar y trabajar con los datos que le llegan por medio de los mensajes de los topics que consume, como el topic orden-de-compra. Usa una colación utf8mb4_general_ci y para el manejo de la misma fue elegido XAMPP como herramienta de desarrollo.

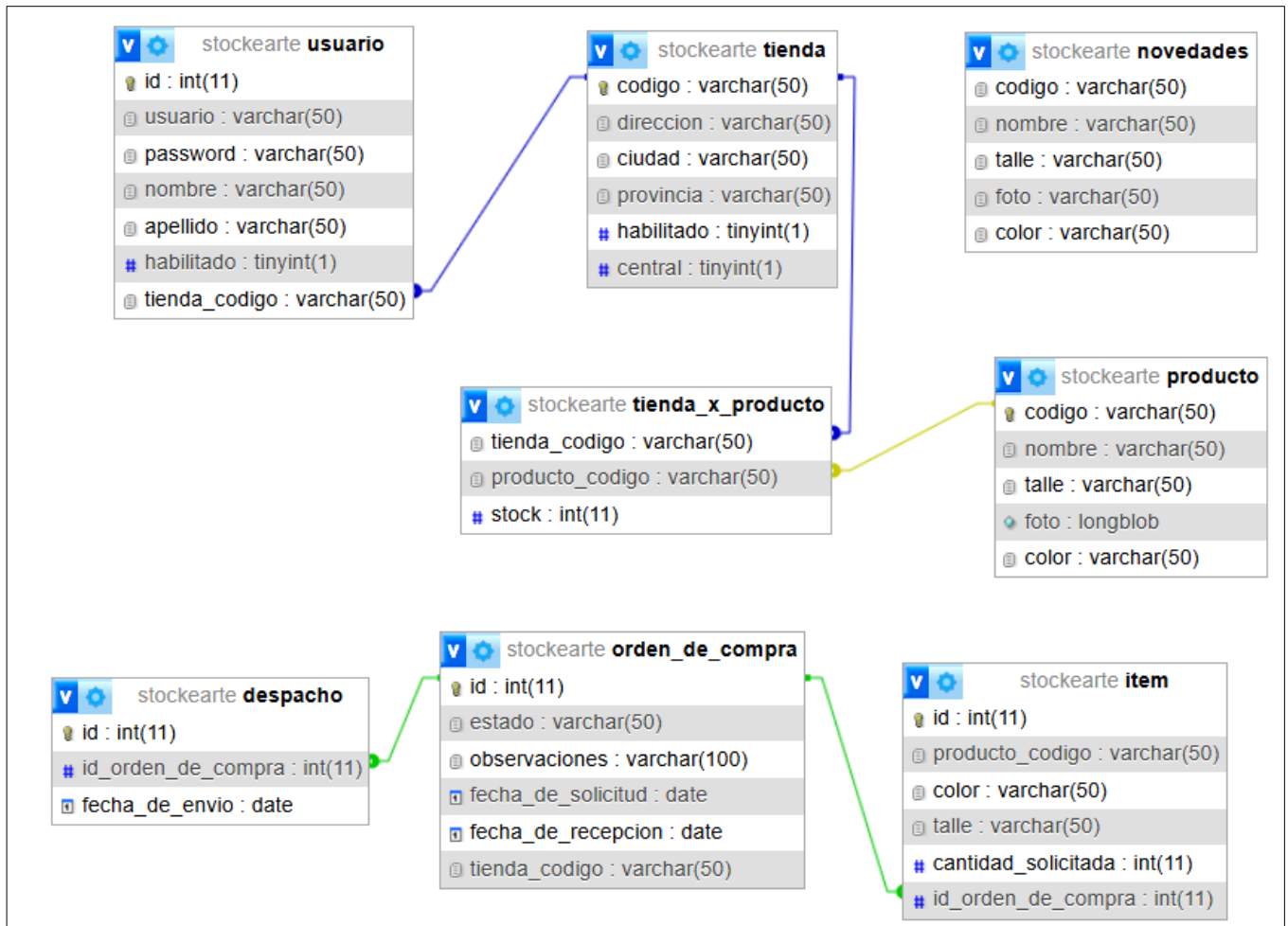
La base de datos “proveedor” esta compuesta por 4 tablas: producto, orden_de_compra, item y despacho. Como una orden de compra puede tener muchos items, sus tablas tienen una relación uno a muchos. Como una orden de compra solo puede tener un despacho, sus tablas tienen una relación uno a uno. La tabla producto no tiene relación con ninguna otra tabla y se usa para almacenar los productos que vende/distribuye el proveedor.

Diagrama Entidad Relación (DER) de la base de datos “proveedor”



La base de datos del primer trabajo práctico, llamada “stockearte”, fue modificada para que, al igual que la base de datos anteriormente mencionada, pueda trabajar con los datos que le llegan por medio de los mensajes de los topics que consume.

Diagrama Entidad Relación (DER) de la base de datos “stockearte”



Como se ve en la imagen, la base de datos “proveedor” fue prácticamente copiada en la base de datos “stockearte” y esto se debe a que básicamente ambas trabajan con los mismos datos. La principal diferencia es que la tabla producto contiene los productos que vende nuestra sucursal (entiéndase sucursal como el conjunto de todas las tiendas) mientras que la tabla novedades contiene los productos que vende el proveedor, los cuales puede tener nuestra sucursal o no.

3.2. Servidor

El servidor del proveedor esta desarrollado con Node.js, Su archivo principal es el archivo servidor.js que se encarga del enrutamiento de las vistas del front-end y de determinar que topics se consumen. El archivo consumo.js tiene todas las funciones que se encargan de trabajar los mensajes de los topics que el proveedor consume.

La carpeta Rutas contiene los archivos con las rutas URLs que maneja el servidor. Para poder enrutar dichas URLs, el servidor usa el modulo express. También usa dicho modulo para iniciar el servidor en localhost:2000.

Para poder trabajar con Apache Kafka, el servidor usa el módulo kafkajs.

3.3. Apache Kafka

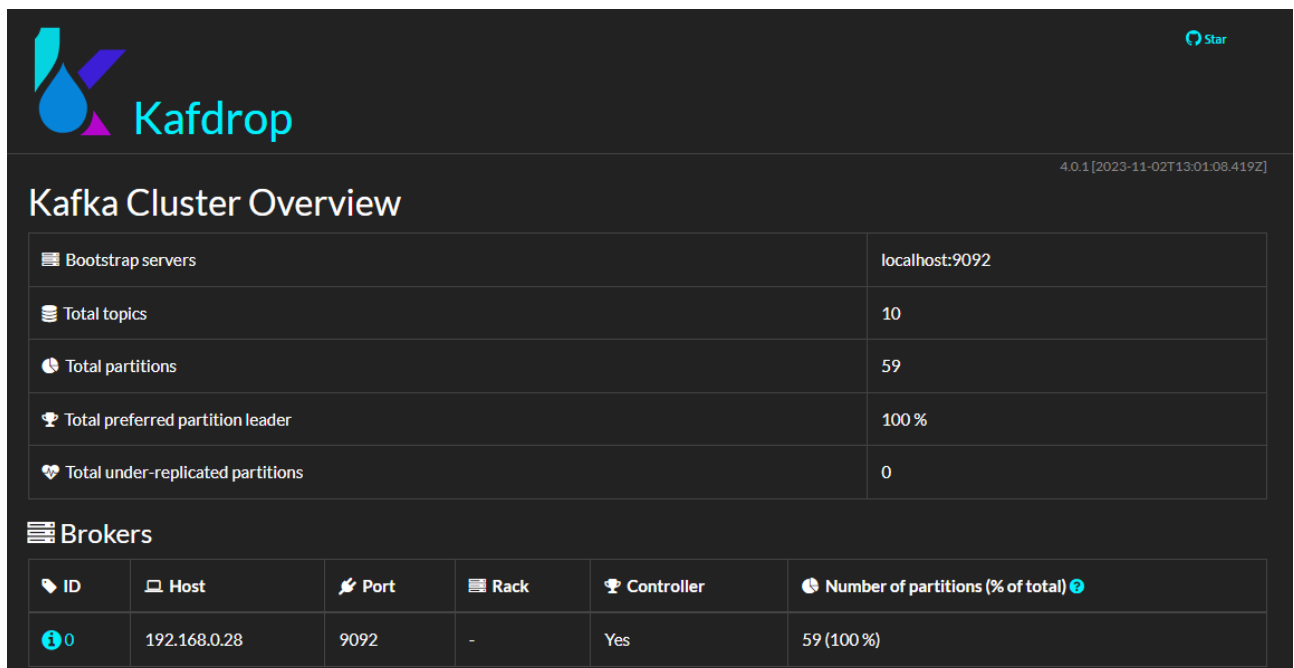
El sistema del primer trabajo práctico (las tiendas) y el sistema del segundo trabajo práctico (el proveedor) se comunican mediante mensajes en los topics de Apache Kafka. Mientras que uno envía mensajes a determinados topics, el otro los consume.

La relación productor/consumidor de los topics sería la siguiente:

Topic	Productor	Consumidor
orden-de-compra	Tienda	Proveedor
{código de tienda}-solicitudes	Proveedor	Tienda
novedades	Proveedor	Tienda
recepcion	Tienda	Proveedor

La herramienta que usamos para monitorear los mensajes de Kafka es un ejecutable java (.jar) llamado Kafdrop. Para poder usarlo solo hay que poner la dirección URL especificada en el archivo application.properties en un navegador (en nuestro caso es localhost:8888).

Index de Kafdrop



Kafdrop

4.0.1 [2023-11-02T13:01:08.419Z]

Kafka Cluster Overview

Bootstrap servers	localhost:9092
Total topics	10
Total partitions	59
Total preferred partition leader	100 %
Total under-replicated partitions	0

Brokers

ID	Host	Port	Rack	Controller	Number of partitions (% of total)
0	192.168.0.28	9092	-	Yes	59 (100 %)

3.4. Interfaz gráfica

La interfaz gráfica del proveedor esta desarrollada con el sistema de plantillas EJS (Embedded JavaScript – JavaScript Embebido) y es bastante simple ya que las únicas funciones que el usuario proveedor puede hacer son actualizar el stock y agregar productos nuevos.

Index de la interfaz gráfica del proveedor

Agregar

Codigo	Nombre	Talle	Color	Stock	
AAAAA	Campera	XL	Gris	3	<div>Editar</div>
BBBBB	Gorra	S	Naranja	12	<div>Editar</div>
CB123	Camisa Básica	M	Rojo	0	<div>Editar</div>
DDDDD	Guantes	L	Verde	8	<div>Editar</div>
PJ456	Pantalones Jeans	L	Azul	1	<div>Editar</div>

Alta de un producto

Agregar producto

Codigo

Nombre

Talle

Foto

Color

Stock

Guardar

Cancelar

Actualización de stock

Actualizando Stock

Stock anterior

12

Stock nuevo

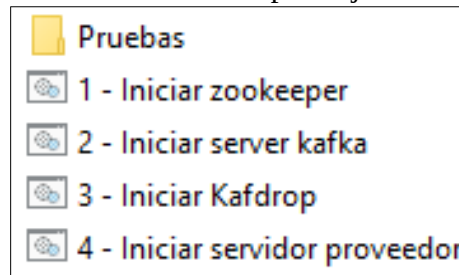
Actualizar

Cancelar

3.5. Ejecutables

Esta carpeta tiene archivos .bat que sirven para ejecutar comandos automáticamente en la consola de comandos de Windows (cmd). Los ejecutables que estan en dicha carpeta permiten iniciar Apache Kafka, Kafdrop y el servidor del proveedor. También hay un ejecutable en la carpeta Pruebas para cargar productos directamente a la base de datos del proveedor pero solo se uso para verificar si los productos se muestran en la interfaz gráfica.

Archivos .bat en la carpeta Ejecutables



4. Pruebas realizadas

Todas las pruebas fueron realizadas en Windows 10 Home. Para poder realizar las pruebas es necesario tener Xampp, Node.js, Python y Apache Kafka instalados.

Enlace al video de las pruebas realizadas:

https://drive.google.com/file/d/1CUIKJlv6AQcwguADgC8liig1fMfQ_0Tu/view?usp=sharing

5. Pendiente del primer trabajo práctico

Enlace al repositorio público de Github del primer trabajo práctico:

<https://github.com/Franco2900/stockearteGrupo5.git>

Enlace para mostrar que se completo la interfaz gráfica faltante en el primer trabajo práctico:

<https://drive.google.com/file/d/1vYXWuByox76j7lS-PXYUAszyV8IWnE6M/view?usp=sharing>