



PICCOLING

R E S T A U R A N T

Specialist in delicious Food



OUR BEST CHEF



ISABELLA
PÉREZ



LUZ ANGELA
CARABALI



NICOLAS
ZAPATA



PROBLEMA



Desafíos en la industria: congestión en la toma de pedidos debido a interfaz poco eficiente, falta de organización en la cocina y atención al cliente, y gestión ineficiente de la base de datos de productos y usuarios.

Impacto: demoras en la entrega de alimentos, baja calidad de servicio y deterioro de la reputación del restaurante.

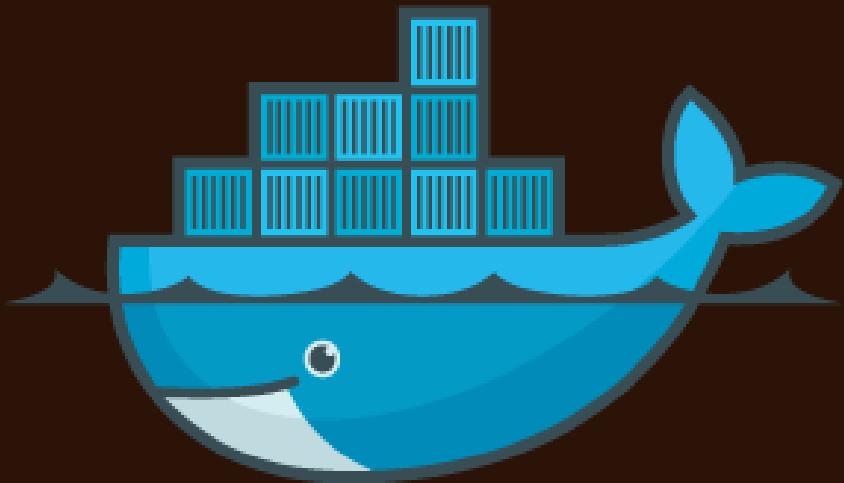
SELECCION DATASET

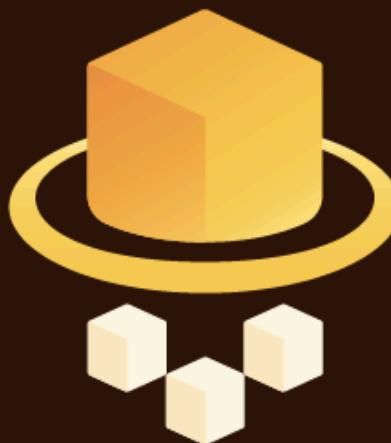
- Fuentes de Datos: Sitios web de restaurantes y suministros de bases de datos.
- Criterios de Selección: Nombre del plato y precio como mínimo.
- Validación y Limpieza: Eliminación de duplicados y corrección de errores.
- Tamaño del Dataset: Amplio número de registros.



ALTERNATIVAS DE SOLUCIÓN

EMPAQUETADO



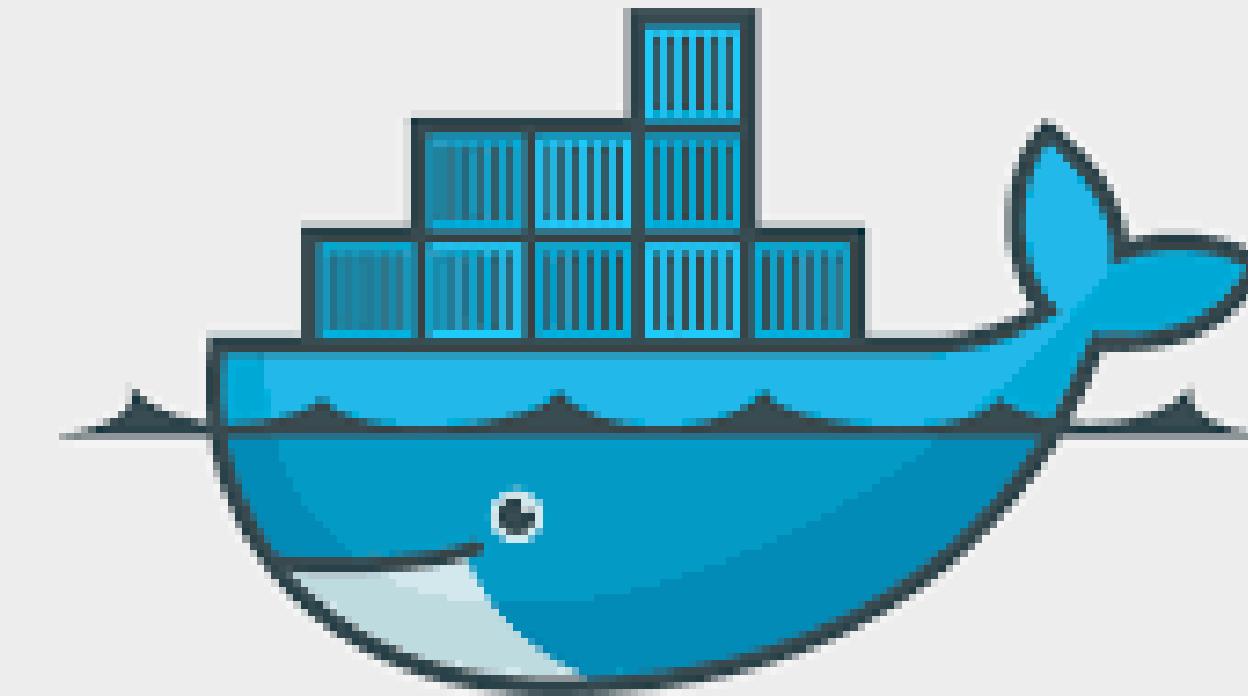
 AWS Fargate

PROCESAMIENTO DISTRIBUIDO



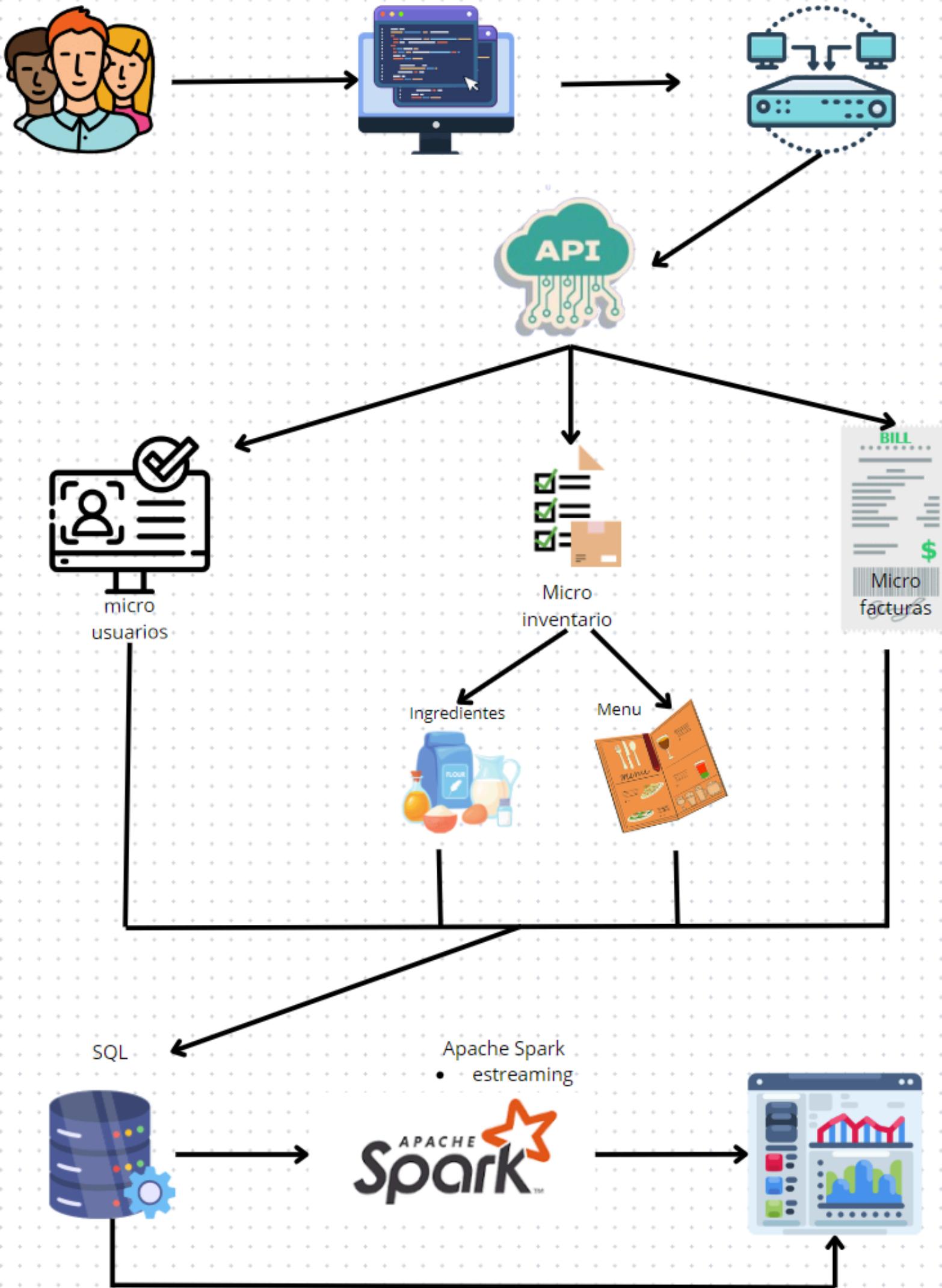


ALTERNATIVAS SELECCIONADAS

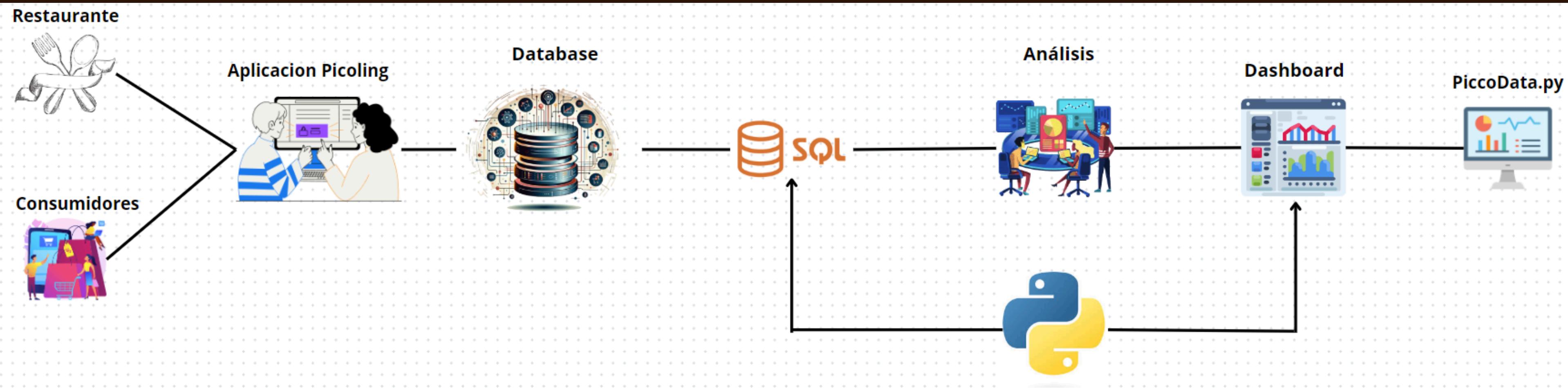


docker

APACHE
Spark™ Streaming



ARQUITECTURA DEL SISTEMA



PIPELINE

Diagrama componentes Piccoling



COMPONENTES

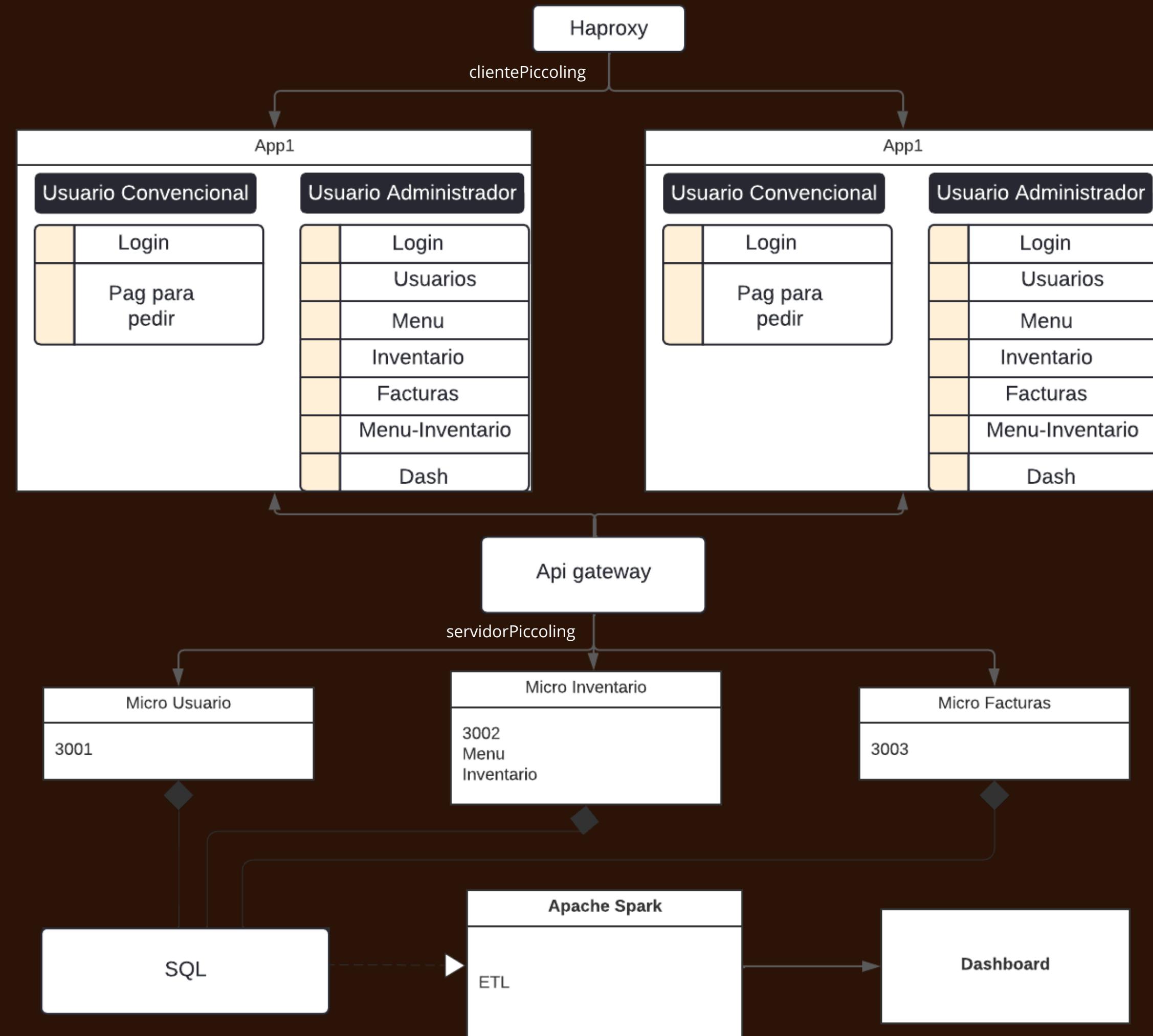
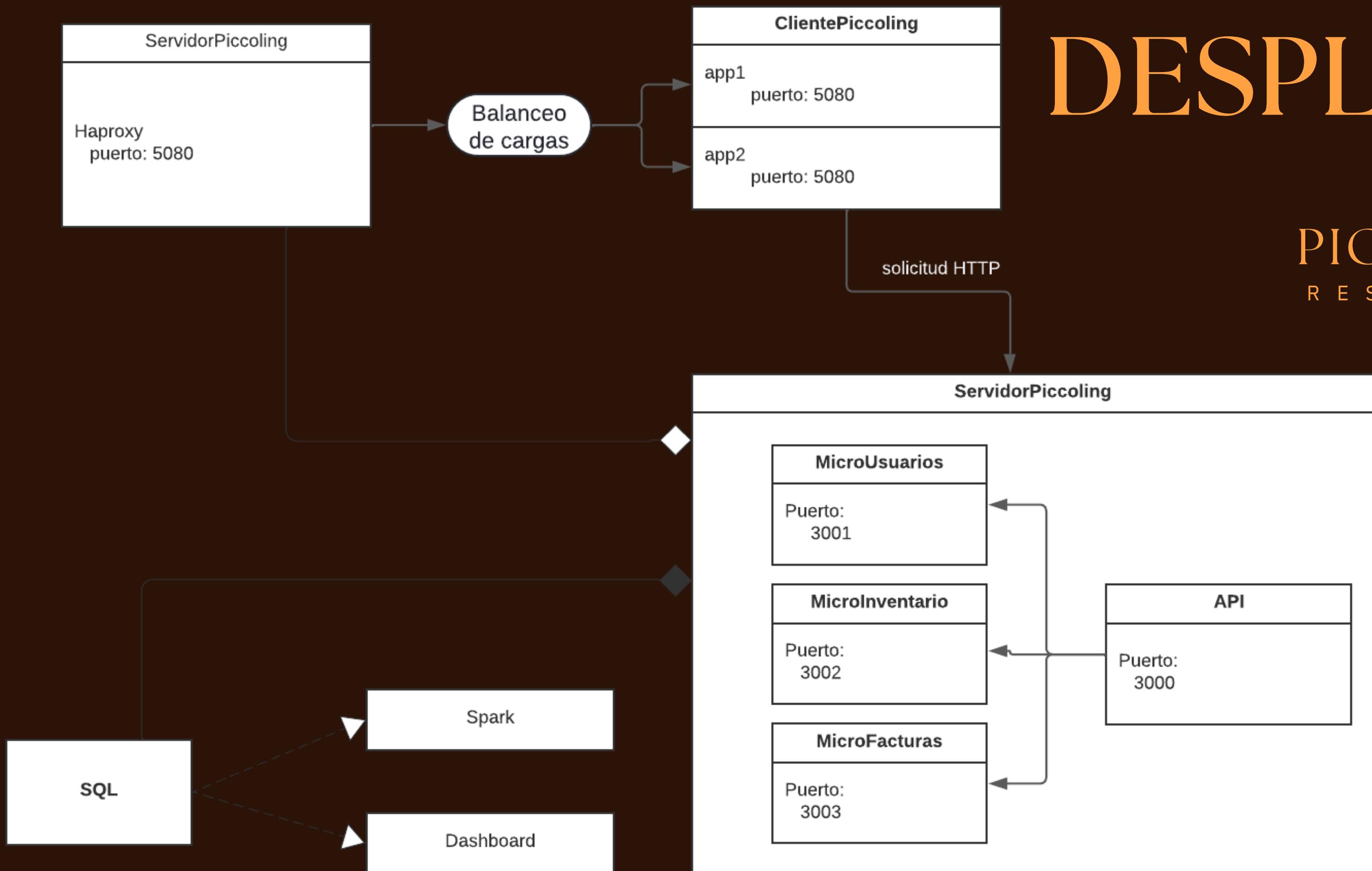


Diagrama despliegue Piccoling



DESPLIEGUE

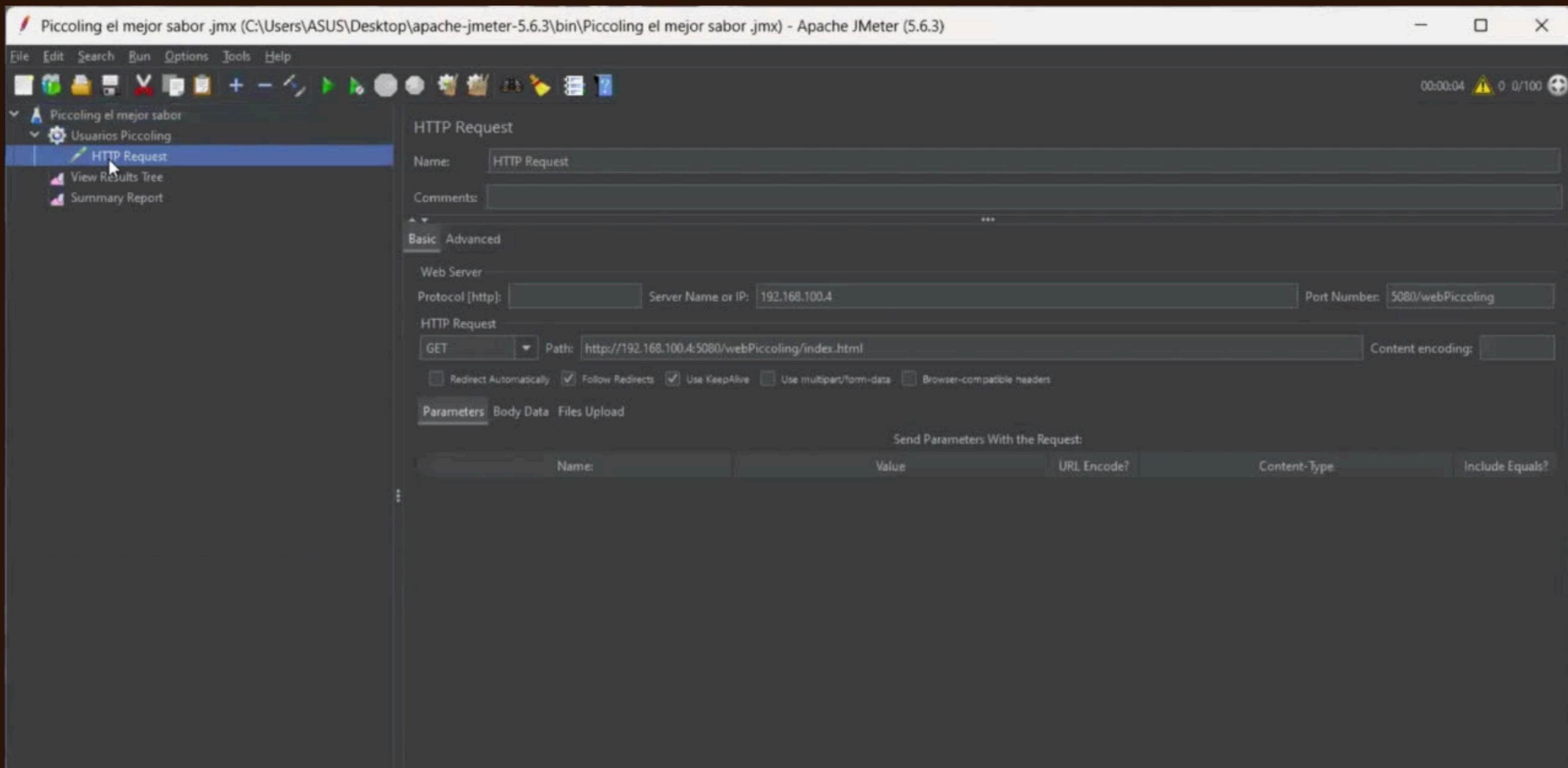


PICCOLING
RESTAURANT

IMPLEMENTACION DE SOLUCION

```
tree -L 1 ./src
.
├── docker-compose.yml
├── haproxy
│   ├── Dockerfile
│   └── haproxy.cfg
├── microFacturas
│   ├── Dockerfile
│   └── src
│       ├── controllers
│       │   └── facturasController.js
│       ├── index.js
│       └── models
│           └── facturasModel.js
└── microInventario
    ├── Dockerfile
    └── src
        ├── controllers
        │   └── inventarioController.js
        ├── index.js
        └── models
            └── inventarioModel.js
├── microUsuarios
    ├── Dockerfile
    └── src
        ├── controllers
        │   └── usuariosController.js
        ├── index.js
        └── models
            └── usuariosModel.js
└── README.md
└── webPiccoling
    ├── actualizarInventarioPicco.php
    ├── admin-facturasPicco.php
    ├── admin-inventarioPicco.php
    ├── admin-menuPicco.php
    ├── adminPicco.php
    ├── admin-preparacionPicco.php
    ├── crearInventarioPicco.php
    ├── crearMenuPicco.php
    ├── crearPreparacionPicco.php
    ├── crearUsuarioPicco.php
    └── dash.php
```

PRUEBAS CARGA Y ESCALABILIDAD





IMPLEMENTACION DE ANALISIS

The screenshot shows a web browser window displaying a food ordering application. The main content area lists various food items with their names, prices, and quantities. A modal dialog box is open, asking if the user wants to translate the page from English to Spanish. The modal includes a 'Traducir' (Translate) button and a 'Logout nico' link.

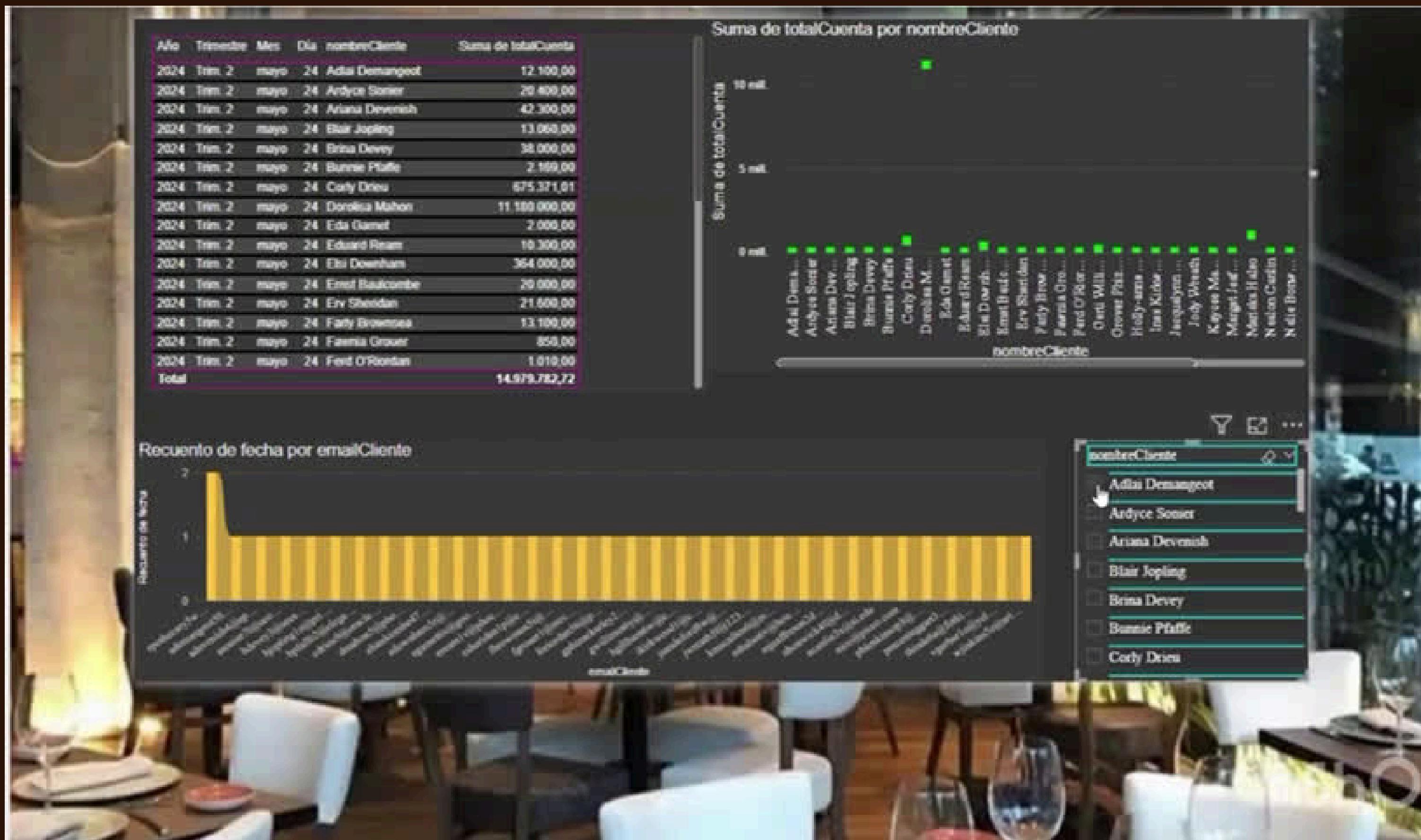
Nombre Platillo	Precio	Cantidad	Cantidad a pedir
Pie Filling - Pumpkin	12044.49	80	0
Soup - Campbells, Butternut	75944.75	39	0
Gin - Gilbeys London, Dry	64268.24	71	0
Rice - Jasmine Sented	91452.02	22	0
Shrimp - Black Tiger 13/15	3544.74	90	0
Chips - Potato Jalapeno	77370.06	90	0
Bag Stand	25693.81	70	0
Bagel - Everything Presliced			

Agregar a la orden

root@servidorPiccoling:~/piccoling-whit-docker/db# ls
init.sql piccodata.py
root@servidorPiccoling:~/piccoling-whit-docker/db#

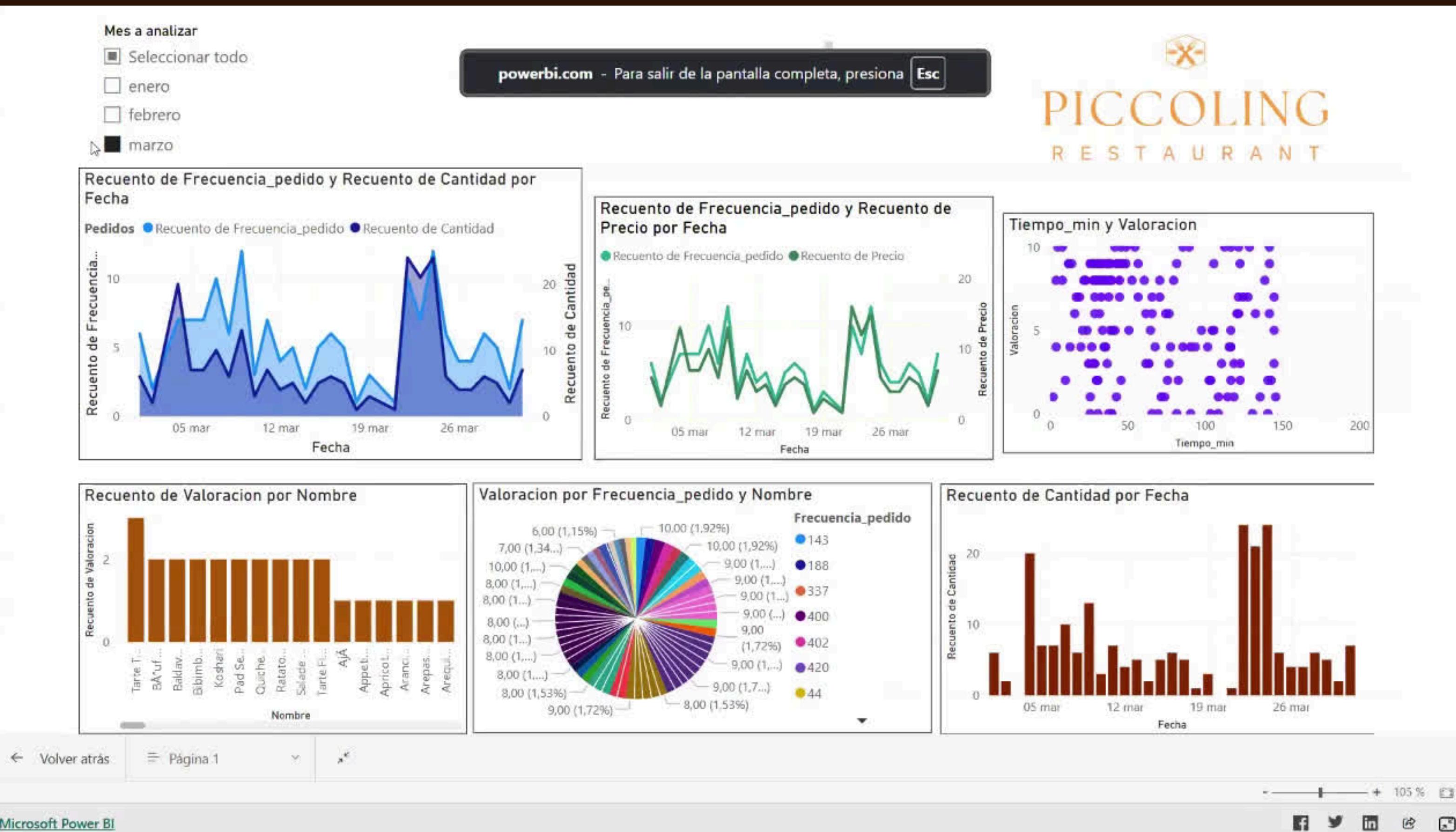


IMPLEMENTACION DE ANALISIS





SISTAVAN DE NOVOCOIN





CONCLUSIONES



PICCOLING
RESTAURANT

THANK YOU

