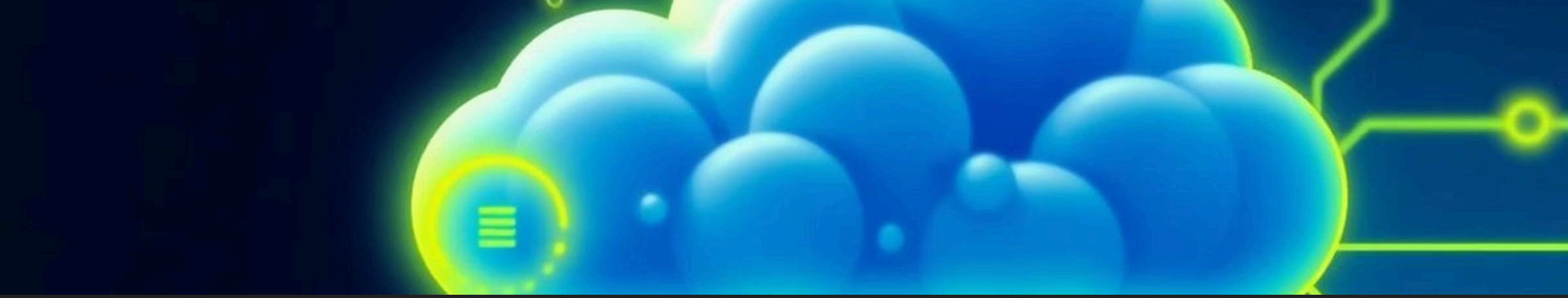
Despliegue de Infraestructura Cloud para E-commerce

Arquitectura escalable en AWS para "Ponchoneta Fútbol", automatizando el despliegue con CloudFormation y siguiendo buenas prácticas DevOps.

Alexis Jaramillo, Felipe Velasco, Kevin Banguero



Solución Propuesta: Arquitectura AWS

Hemos diseñado una arquitectura robusta con VPC, RDS MySQL, backend en EC2 con Auto Scaling y monitoreo con CloudWatch.

Red Privada (VPC)

Aislamiento y seguridad de la infraestructura.

RDS MySQL

Base de datos relacional segura y administrada.

EC2 + ALB

Backend escalable con balanceador de carga.

CloudWatch + SNS

Monitoreo y alertas automáticas.

Arquitectura General del Proyecto

La solución integra componentes clave de AWS para una operación eficiente y segura.

VPC

Red privada con subredes públicas y privadas.

EC2 + ALB

Backend con alta disponibilidad.

RDS

MySQL administrado, sin acceso público.

S3 (logs) + CloudTrail

Registro de eventos para auditoría.

SNS + CloudWatch

Notificaciones automáticas y monitoreo.

Frontend y Pasarela de Pagos Wompi

El frontend en HTML/JS muestra productos, gestiona el carrito e integra Wompi para pagos seguros.

Mostrar Productos

Desde la API del backend.

Carrito de Compras

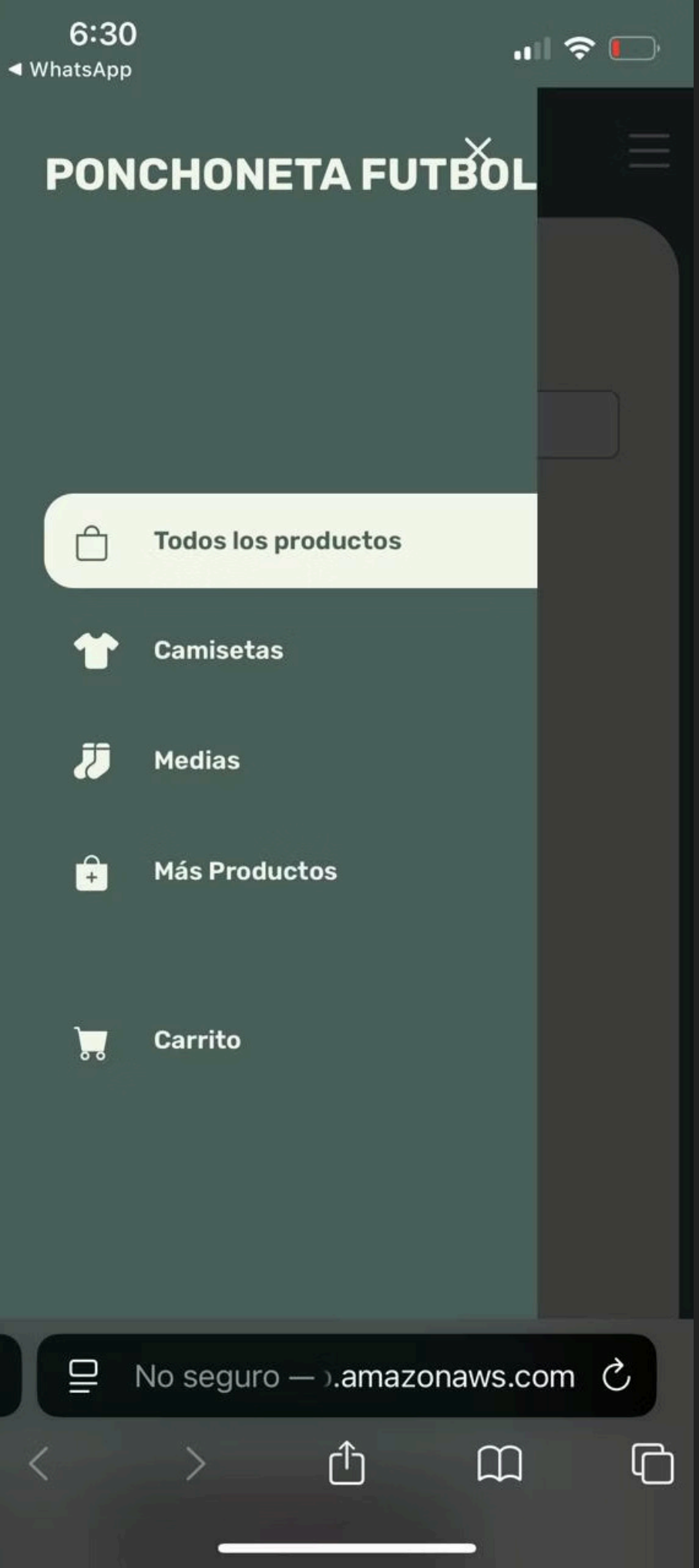
Gestión con localStorage.

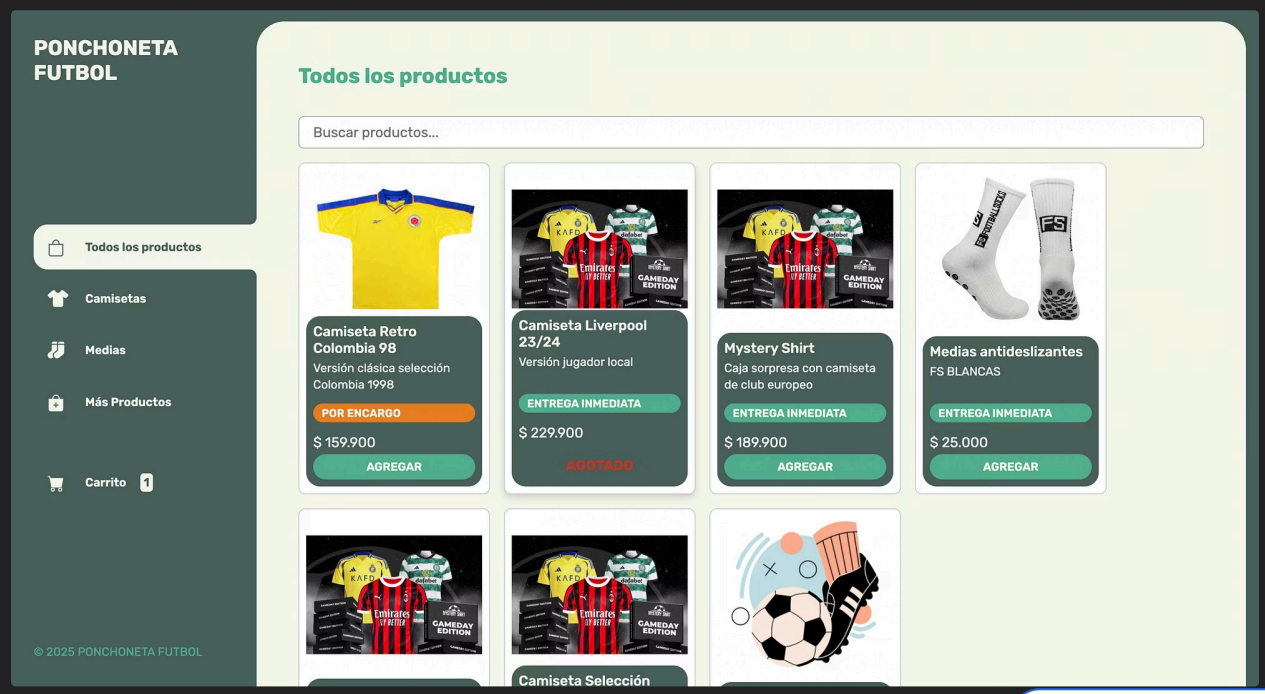
Integración Wompi

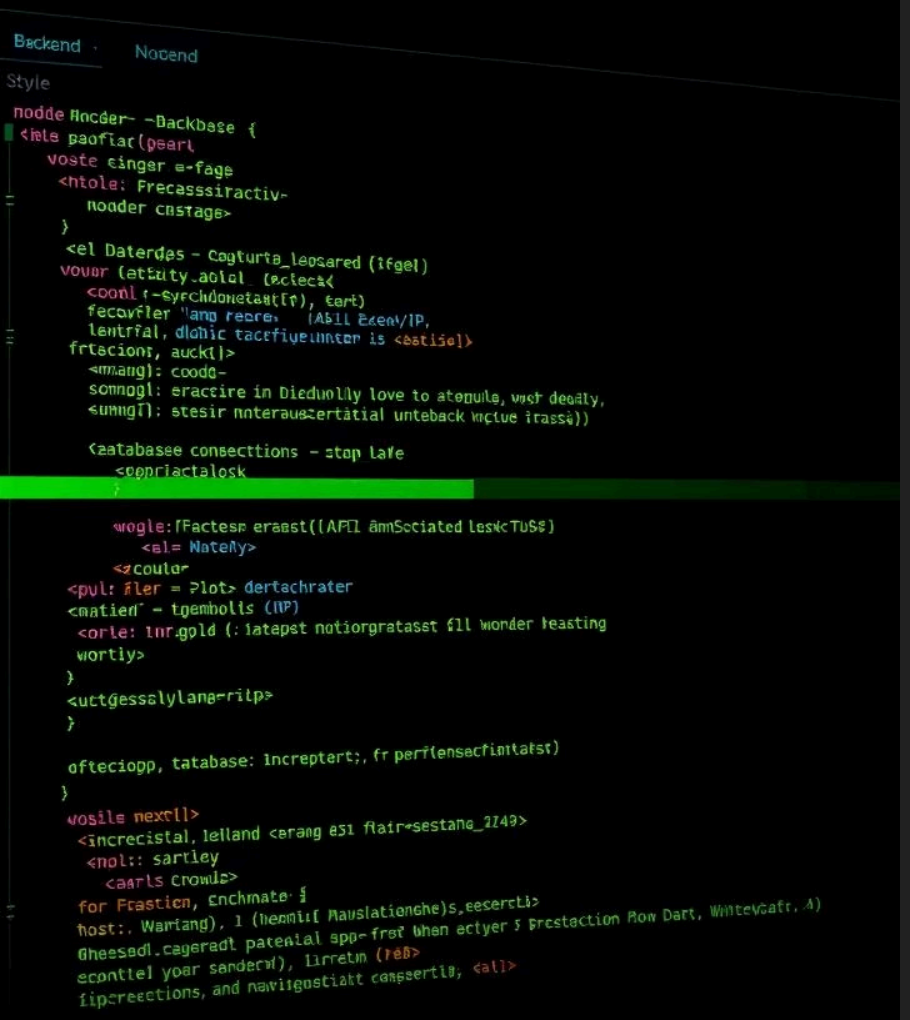
Firma de integridad y checkout seguro.

Registro de Orden

En la base de datos tras el pago.







Configuración del Backend Node.js

El servidor Node.js maneja la conexión a RDS y expone endpoints REST para productos, órdenes y la integración con Wompi.

Conexión a RDS

Uso de mysql2/promise para la base de datos.

Endpoints REST

Para productos, órdenes y firma SHA256 de Wompi.

Webhook Wompi

Recepción de eventos de pago.

.env Protegido

Claves y parámetros sensibles seguros.

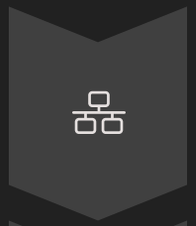
Pasos para desplegar App

1. Crear llave con el nombre "llave"
2. Correr estos comandos:

```
git clone https://github.com/felipevelasco7/PonchonetaStore.git
cd PonchonetaStore
chmod +x script.sh
```

Despliegue Automático con Script

Un script de Bash automatiza el despliegue de la infraestructura en tres stacks de CloudFormation.

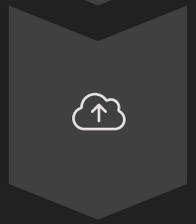


WebAppNetwork

Crea la red completa (VPC, subredes, NAT, IGW).

WebAppDatabase

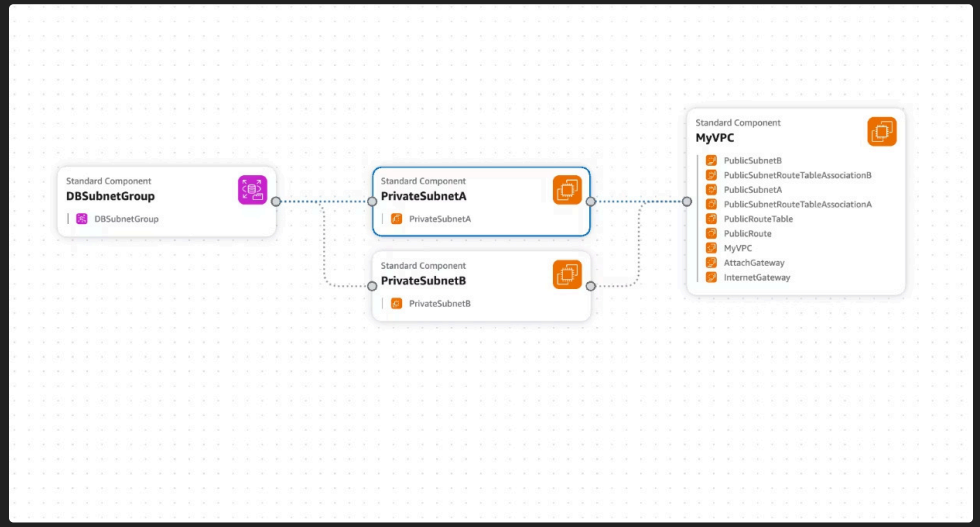
Despliega RDS MySQL en subredes privadas.

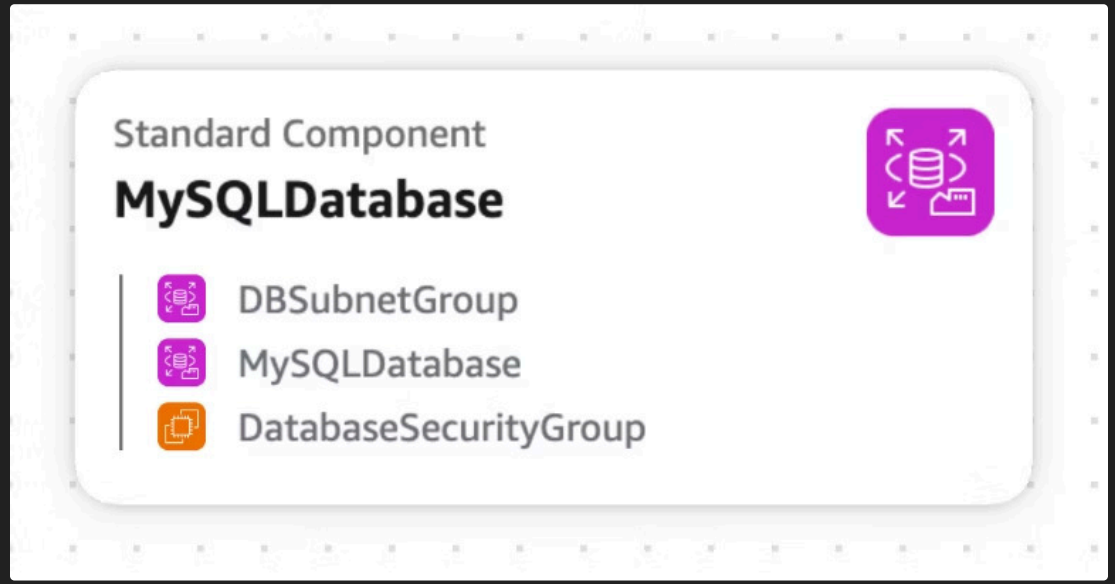


WebAppApp

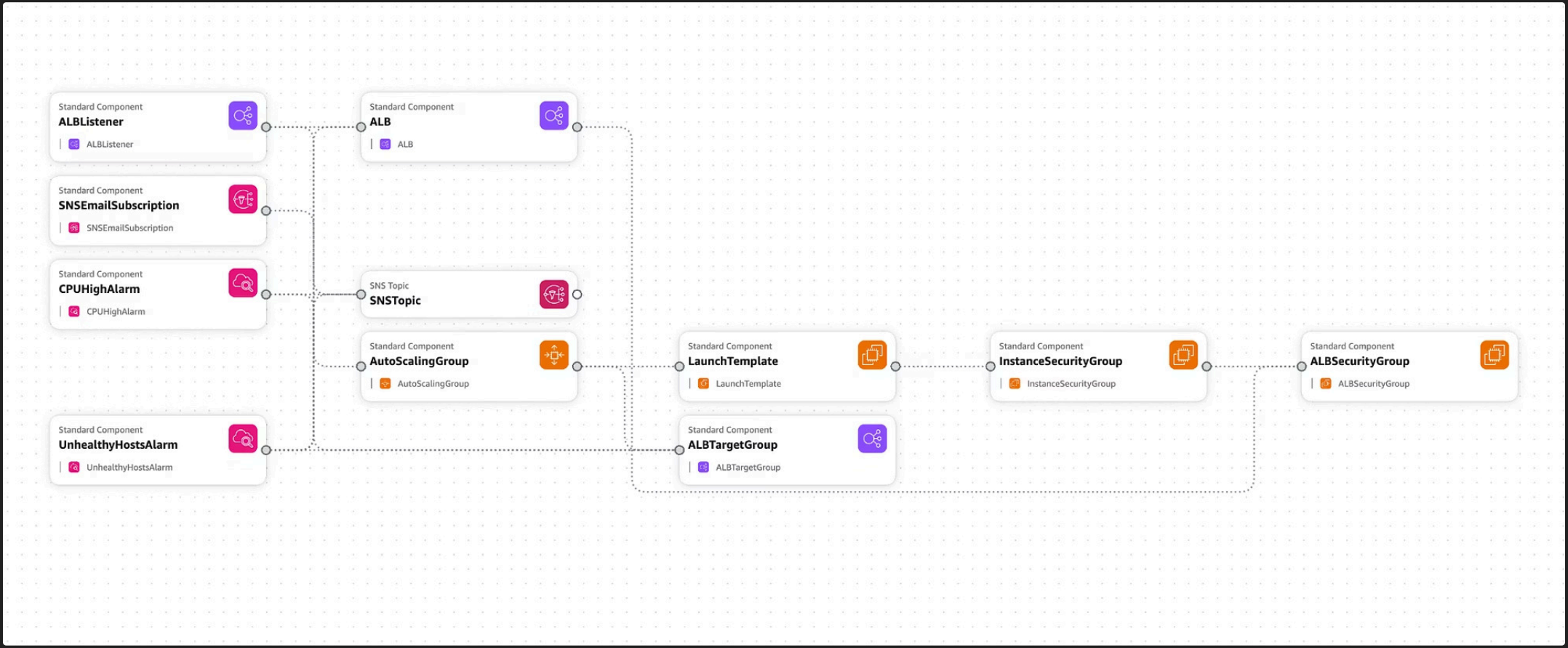
Lanza backend EC2, ALB, AutoScaling y alertas.

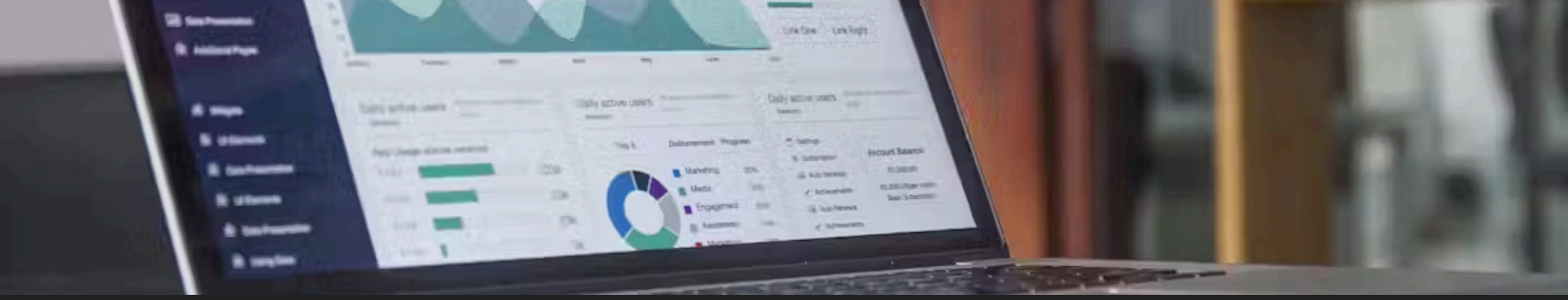
Network stack y DB





App Stack





Resultados y Aprendizajes Clave

Logramos una infraestructura escalable y reproducible, con pagos exitosos y automatización completa.

1

Infraestructura Escalable

Despliegue reproducible.

2

Aplicación Funcional

Desde ALB.

3

Base de Datos Segura

En red privada.

4

Pagos Exitosos

Con Wompi Sandbox.