PRESENTACIÓN ESTILO COLAS

Proyecto de Investigacion

PUNTOS A INVESTIGAR

- Estilo arquitectonico en colas (flujo de datos)
- Frontend: Vue
- Backend: Go
- Gestor de datos: KafKa
- MiddleWare: Express

Estilo arquitectónico en colas

Que es?

Es un enfoque de diseño de software en el que las aplicaciones se comunican entre sí mediante el intercambio de mensajes en una cola de mensajes centralizada

Un evento envía un mensaje que contiene información sobre ese evento a una cola de mensajes y así otras aplicaciones que estén interesadas en ese evento pueden suscribirse a la cola de mensajes y recibir los mensajes a medida que se producen.

Ventajas

01

Facil escalabilidad

02

Altamente flexible y adaptable

03

Alta tolerancia a fallos

04

Permite alta moduralizacion

Desventajas

01

Es complejo

02

Problemas de latencia

03

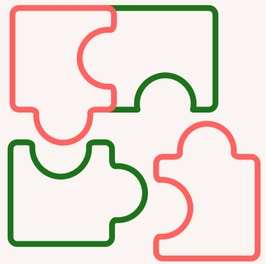
Problema de sobrecarga de mensajes

Principios SOLID

- Única Responsabilidad: cada componente debe estar diseñado para manejar una única tarea.
- Abierto Cerrado: el sistema esta perfectamente diseñado para integrar nuevos sintemas sin modificar este.
- Liskov: puede intercambiar sin afectar su integridad por la linealidad de este.
- Segregacion de interfaces: los componentes pueden estar centrados directamente a sus interfaces y las funcionalidades necesarias

Atributos de calidad

- Escalabilidad: permite manejar grandes cantidades de datos y traficos en tiempo real.
- Disponibilidad: permite que los componentes sean independientes entre si.
- Fiabilidad: Puede procesar varios datos en segundo plano.
- Rendimiento: al utilizar cola para gestionar su fluejo de datos puede procesar los datos de forma asincrona.
- Flexibilidad: permite un sistema de publicación/suscriptor



VUE

Que es?

Framework progresivo de JavaScript utilizado para crear aplicaciones web SPA en la parte de frontend.

La estructura de VUE se presenta en varias capas:

- Render declarativo
- Sistema de componentes
- Enrutamiento de cliente
- Manejo de estados

Ventajas

01
Unión de React y Angular

02
Desarrollo progresivo

03
Adaptable a otros proyectos desarrollados en otros frameworks

04
Facil de aprender

Desventajas

01
Existe poca documentación

02
Al utilizarse en proyectos grandes no suele ser facil de utilizar como otros frameworks

CASOS DE ESTUDIO PRINCIPALES DE VUE

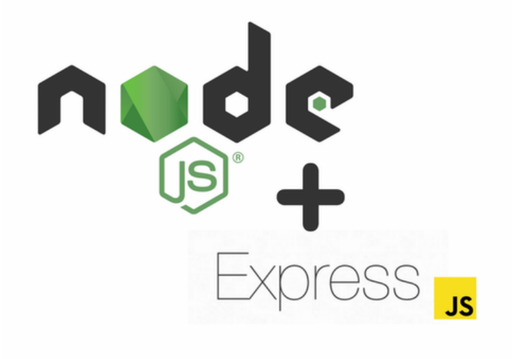
Xiaomi

Alibaba

GitLab

Adobe

Codeship



EXPRESS

Que es?

Framework popular de node js utilizado para construit aplicaciones web y API's

Mayormente utilizado para agregar funcionalidades a las aplicaciones web:

- autenticación
- registro de solicitudes
- manejo de peticiones y respuestas HTTP

Ventajas

01
Eficiencia en manejo de solicitudes HTTP

02
Brinda gran variedad de herramientas y bibliotecas

03
Facil de aprender

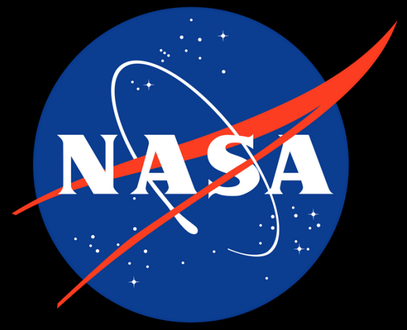
Desventajas

01
Framework minimalista sin estructura

02
Poca escabilidad y organización

03
No tiene mecanismos de seguridad integrados

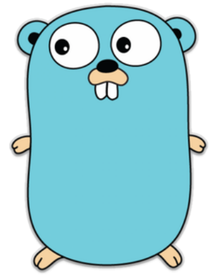
CASOS DE ESTUDIO PRINCIPALES DE EXPRESS



Telemetría y procesamiento de datos

Implementación de APIs

Plataforma en la nube



Go

Ventajas

01

Buen funcionamiento para aplicaciones que consuman mucho rendimiento

02

Es de sintaxis simple

03

Tiene incorporado características de seguridad

04

Puede manejar gran concurrencia

Desventajas

01

Pocas ayudas por ser nuevo

02

Tiene características avanzadas a pesar de su simpleza

Que es?

Lenguaje de programación concurrente y compilado inspirado en C, orientado a objetos y con seguridad en memoria y recolección de basura. Ha sido desarrollado por Google.

Sus principales características son que es un lenguaje de tipado estático y es compilado como C y C++ y que a pesar de que tiene la característica de tener programación orientada a objetos este no dispone de herencia y de palabras clave que soporten el paradigma. Permite el uso de delegación y polimorfismo además de permitir concurrencia a través de “gorutinas”

CASOS DE ESTUDIO PRINCIPALES DE GO

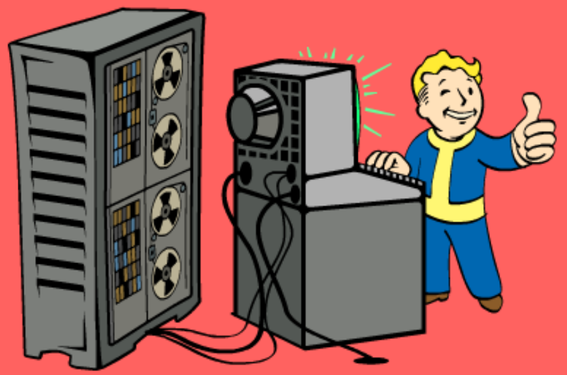
Uber Eats/Uber Money

Plataforma de streaming

Aprendiendo como desarrollar

Aprovechamiento de recursos

Infraestructura en la nube

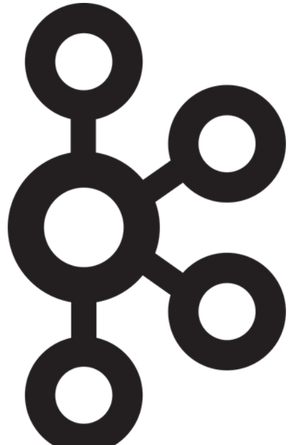


Kafka

Que es?

Es una plataforma de streaming de código abierto desarrollada por Apache Software Foundation, se basa en el concepto de “publicación/suscripción” por lo que los datos solo llegan a los consumidores suscritos a estos todo en tiempo real

Kafka es un sistema altamente escalable que puede manejar grandes volúmenes de datos y que puede ejecutarse en un clúster de servidores por lo que se amplía horizontalmente para satisfacer sus necesidades de crecimiento y también posee arquitectura distribuida por lo que puede ejecutarse en varios servidores



Ventajas

01

Buena disponibilidad de datos a través de replicacion

02

Rapido y eficiente en grandes flujos de datos

03

Buena integración en varios lenguajes

04

Buen procesamiento de datos en tiempo real

Desventajas

01

Mayor complejidad en su configuracion y mantenimiento

02

Consume mas recursoso que otras opciones

03

Presenta problemas en proyectos con alto grado de estructuracion

CASOS DE ESTUDIO RINCIPALES DE KAFKA

Spotify

Procesamiento de datos

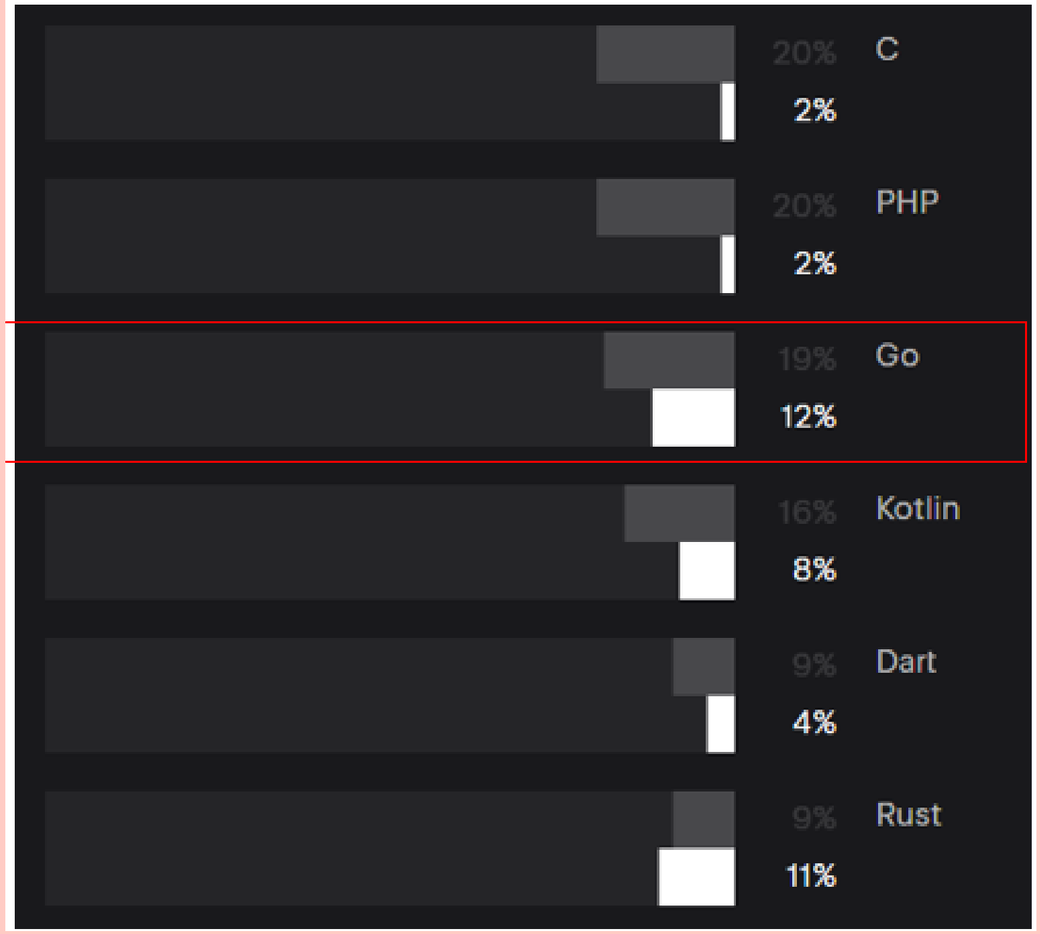
Pinterest

Proceso de aplicaciones en redes sociales

Sistema de mensajeria

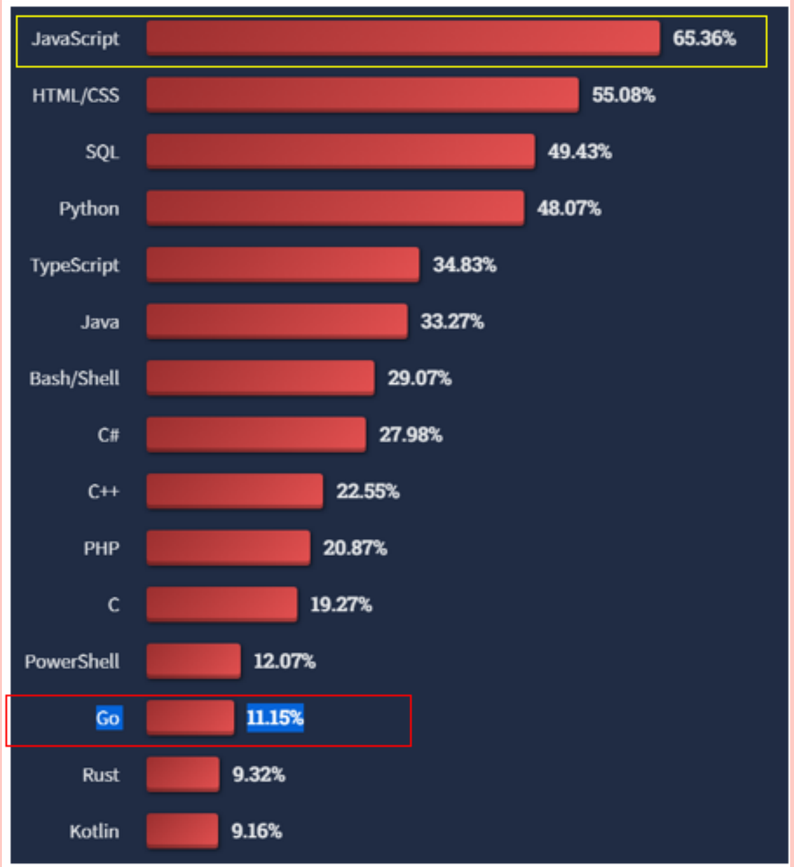
ESTADISTICAS DE USO

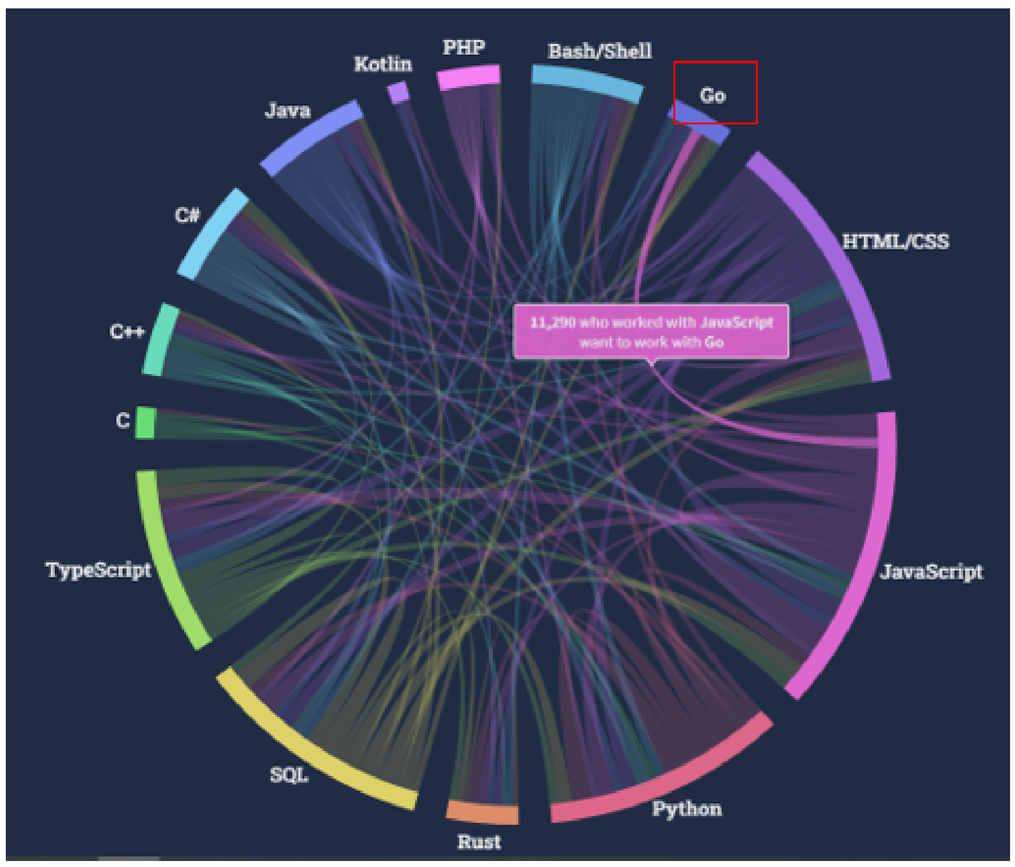
2017	2018	2019	2020	2021	2022	
65 %	64 %	69 %	70 %	69 %	65 %	JavaScript
60 %	55 %	61 %	61 %	60 %	54 %	HTML/CSS
47 %	51 %	50 %	54 %	49 %	48 %	Java
42 %	47 %	56 %	56 %	54 %	49 %	SQL
32 %	41 %	49 %	55 %	52 %	53 %	Python
30 %	26 %	29 %	27 %	32 %	20 %	PHP
20 %	22 %	24 %	22 %	21 %	23 %	C#
17 %	18 %	20 %	27 %	23 %	25 %	C++
15 %	16 %	17 %	23 %	19 %	20 %	C
12 %	17 %	25 %	28 %	29 %	34 %	TypeScript
10 %	8 %	11 %	8 %	6 %	5 %	Ruby
9 %	8 %	11 %	9 %	7 %	7 %	Swift
8 %	12 %	18 %	19 %	17 %	19 %	Go
7 %	5 %	6 %	4 %	3 %	3 %	Objective-C
7 %	5 %	6 %	5 %	3 %	3 %	Scala
2 %	9 %	16 %	17 %	14 %	16 %	Kotlin
–	2 %	5 %	7 %	6 %	9 %	Rust
–	29 %	40 %	39 %	37 %	34 %	Shell



Jetbrains- Estado del ecosistema de desarrolladores

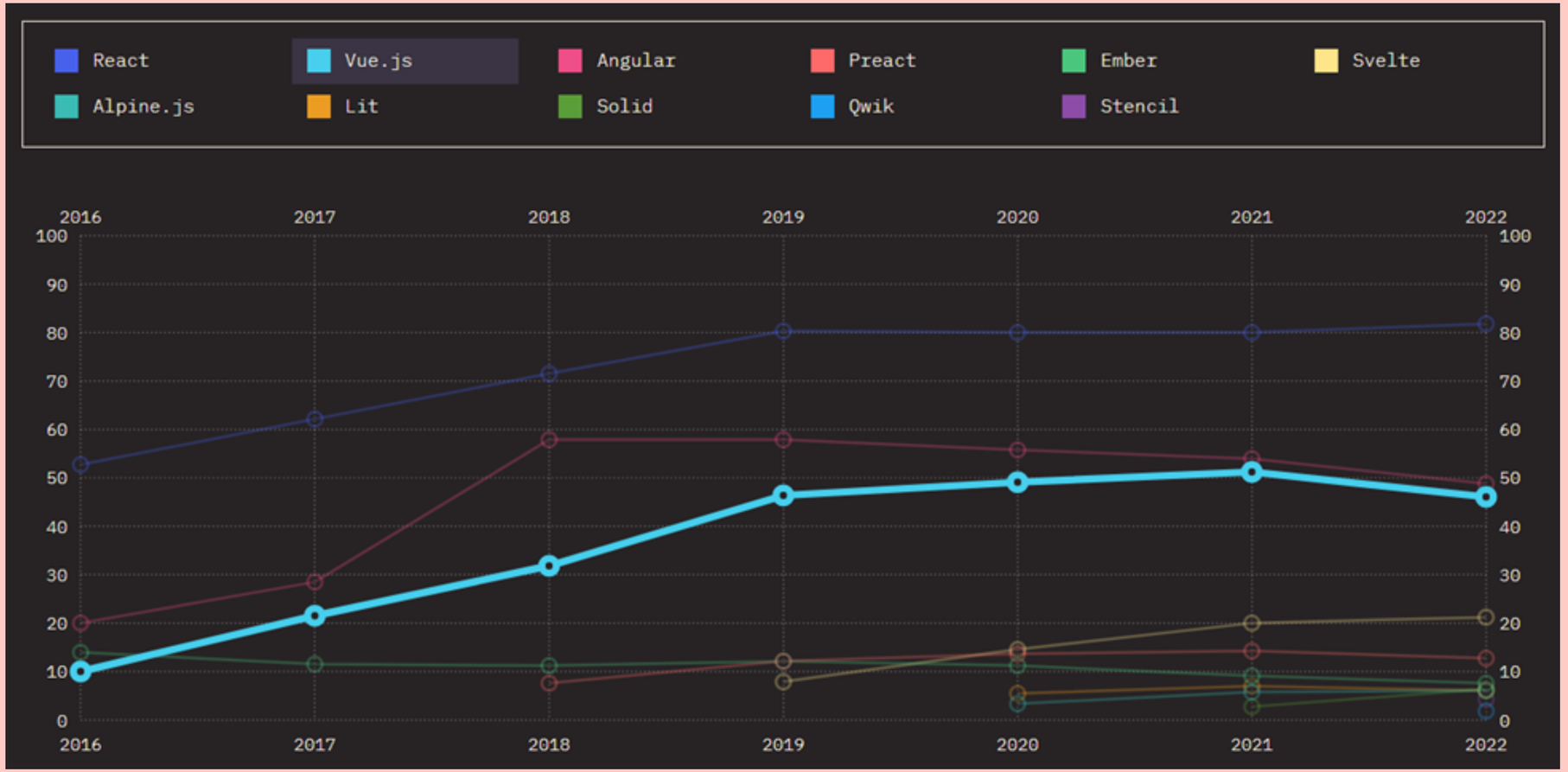
ESTADÍSTICAS DE USO





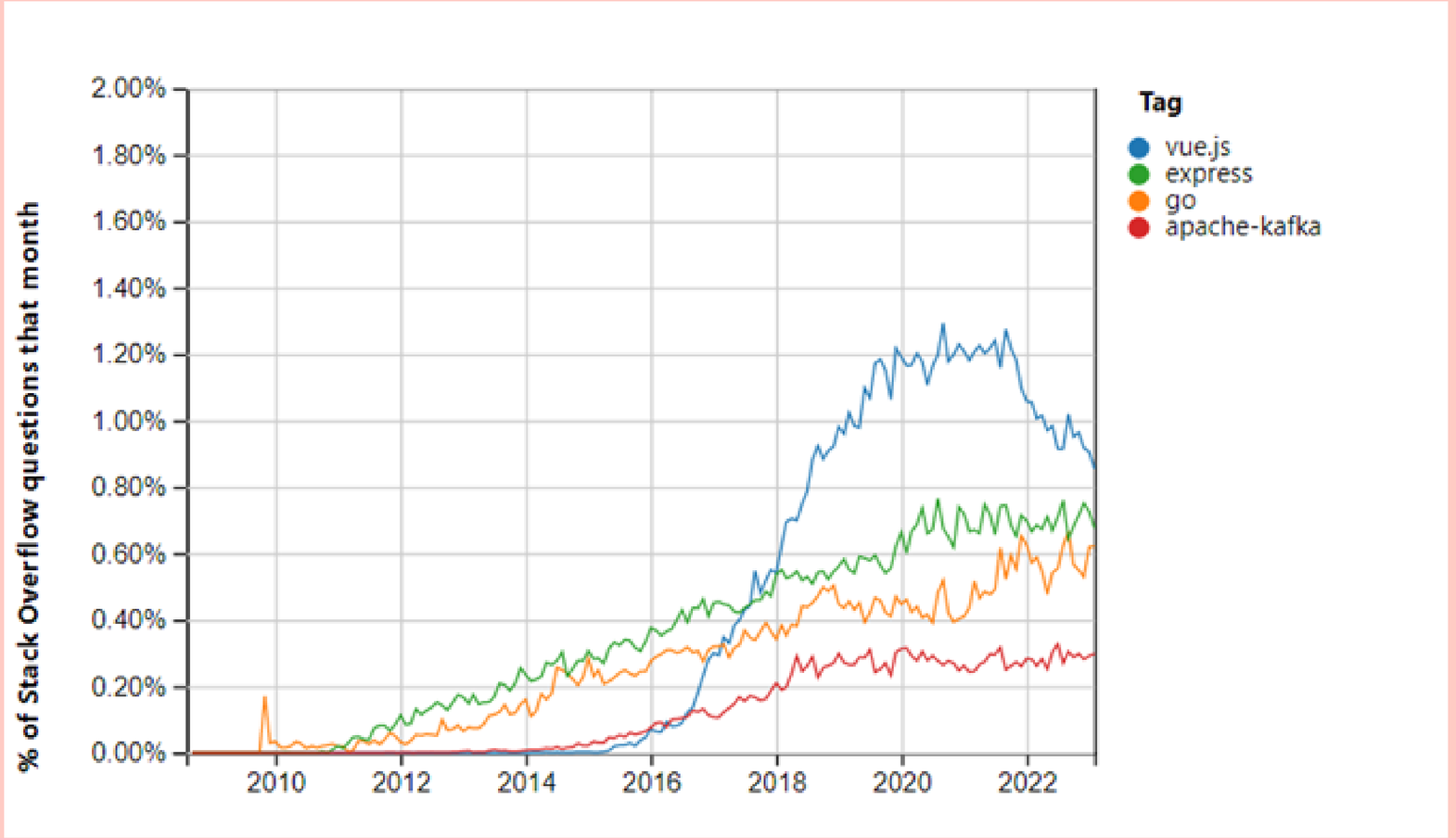
StackOverflow- Developer Survey

ESTADÍSTICAS DE USO



Frameworkas mas usados- The state of JS 2022

ESTADÍSTICAS DE USO



Todas las tecnologias en uso- Stack overflow

Caso practico: digiturno de eps

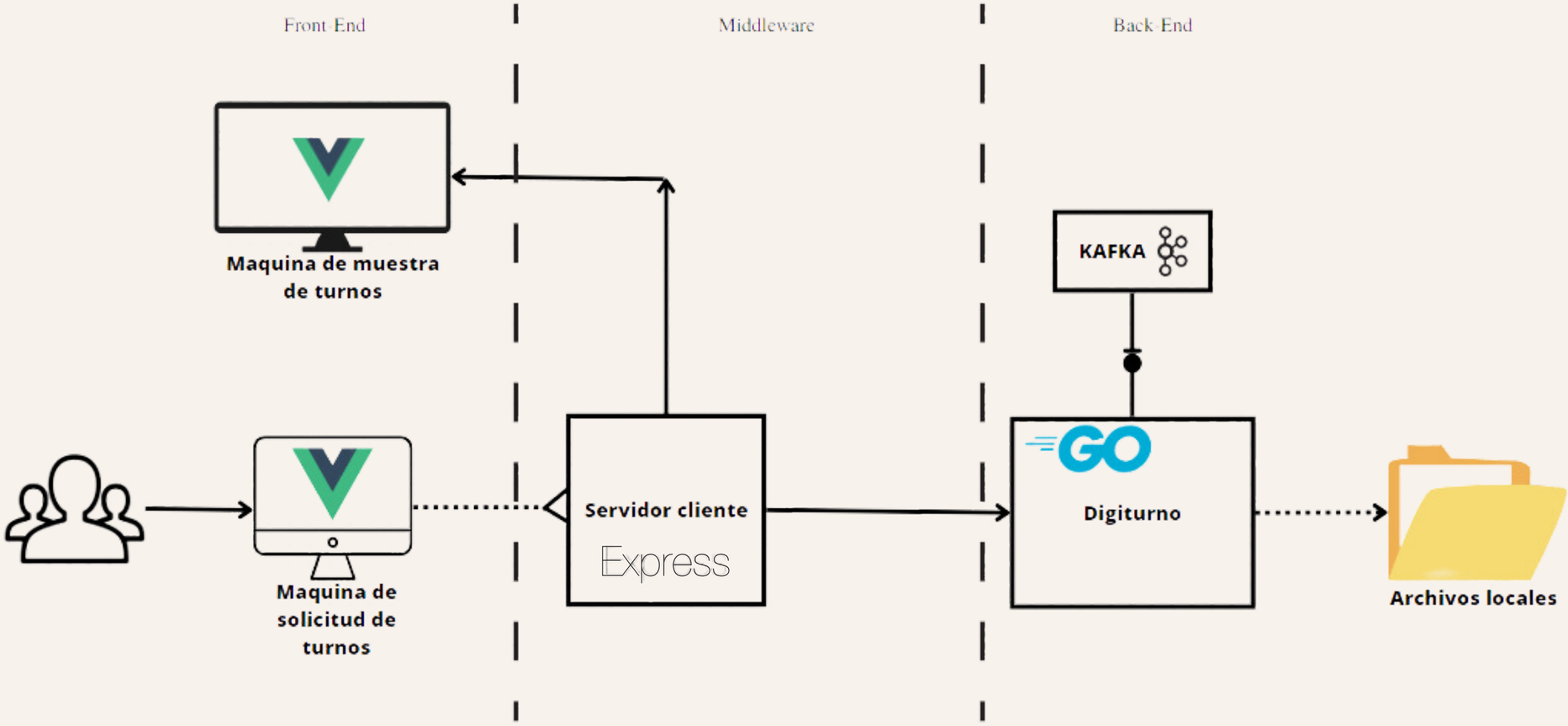


Diagrama de alto nivel

Caso practico: digiturno de eps

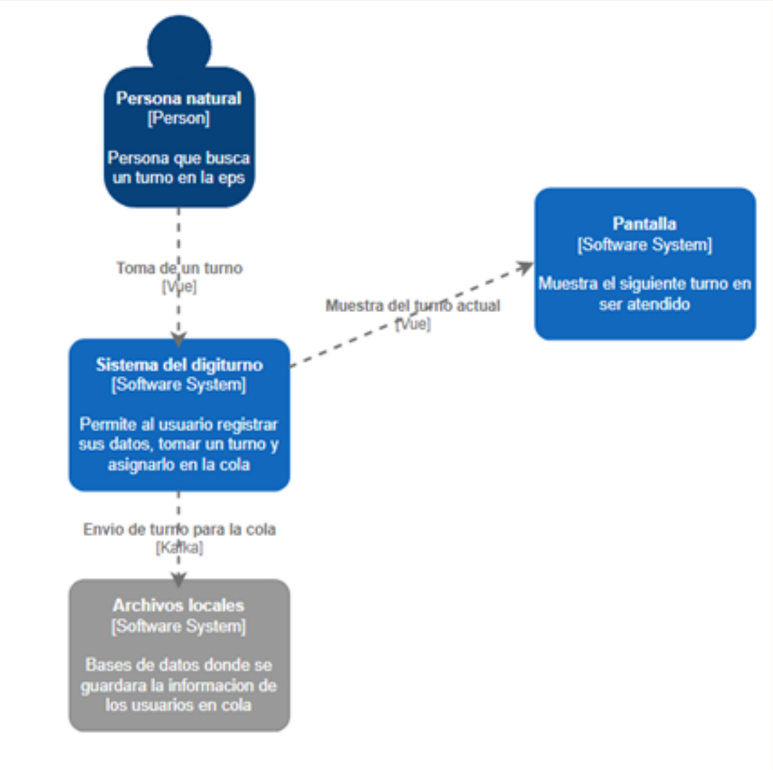


Diagrama de contexto

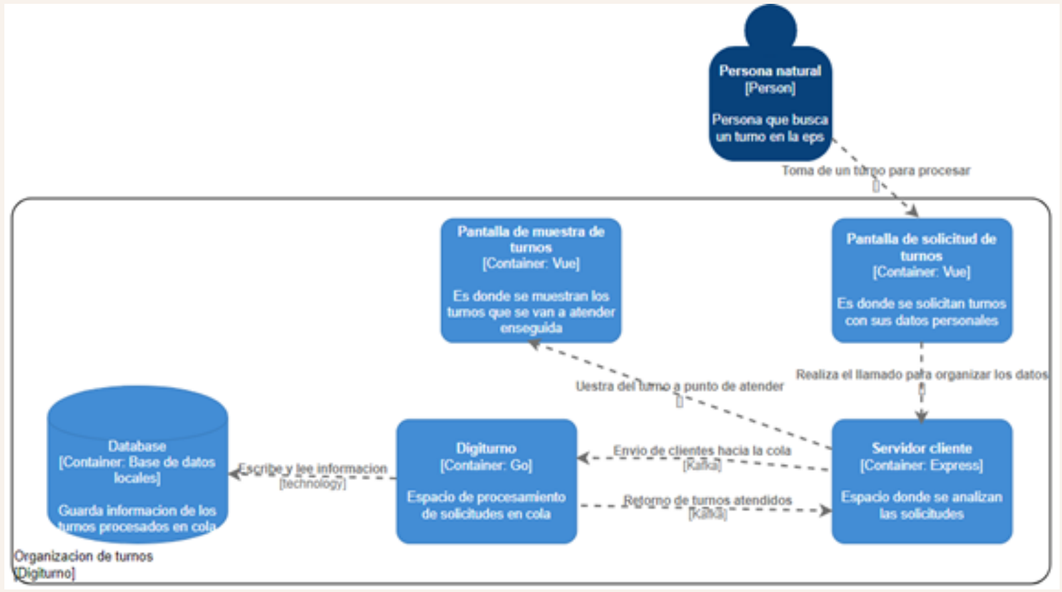


Diagrama de contenedores

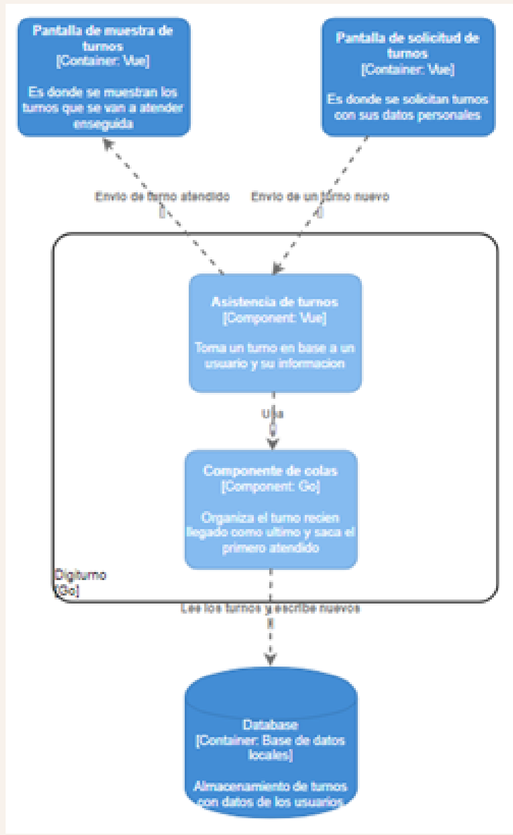


Diagrama de componentes

Conclusiones y lecciones aprendidas

- Existencia de varias tecnologías en varias formas.
- Hay una gran utilidad de varios medios para varias situaciones.
- Existen todo tipo de datos acerca de framework nuevos y actuales que se centran en distintos métodos de programación.
- La preparación es lo mas importante para el desarrollo.
- la utilidad de un framework o lenguaje de programacion no se definen por su tiempo sino por su utilidad.
- No existe una solucion perfecta para todos los problemas de la programación.
- El estilo arquitectónico en colas es especifico pero eficaz.
- Para la eleccion de un estilo arquitectonico hay que tener en cuenta en que problema se centra, su forma y como puede ayudar para tener un camino claro.

¡MUCHAS GRACIAS POR SU ATENCIÓN!