Platform Engineering: Impulsando eficiencia en equipos de desarrollo a través de las Plataformas de Desarrollo Interno (IDP)

Alejandra Bricio

Prácticas modernas para crear software con calidad y sabor

#SGVirtual

Platform Engineering

Transformando la eficiencia de los equipos de desarrollo

👋

Alejandra Bricio

cloud, scripting, k8s
@alebricio

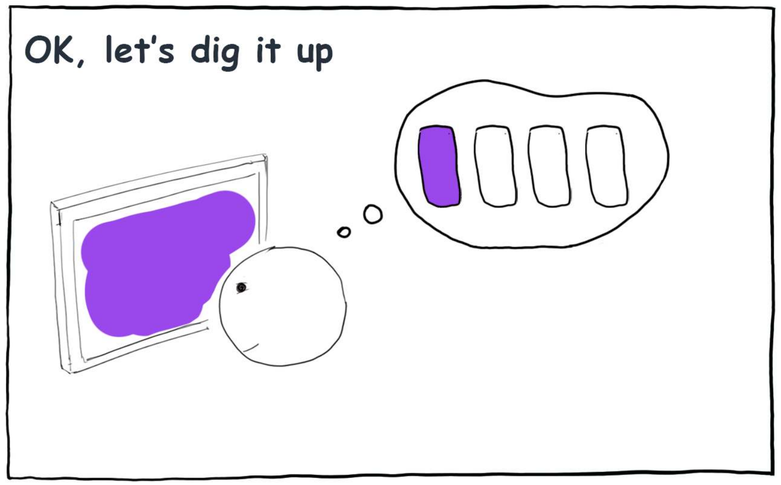
Agenda

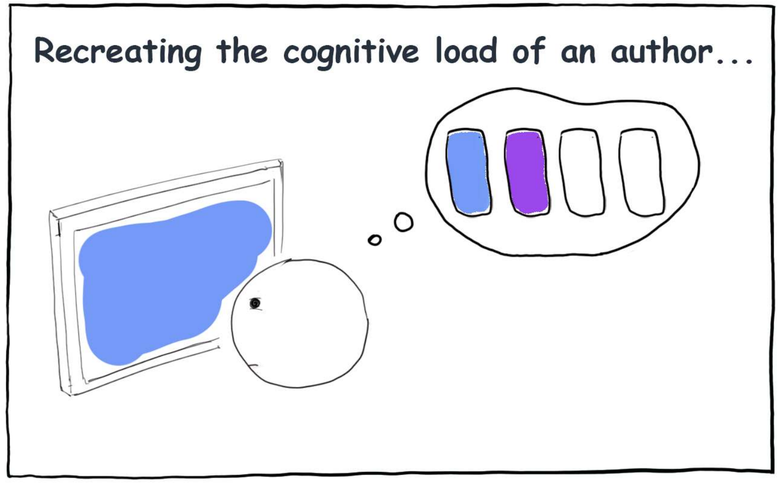
1. Carga Cognitiva
2. Platform as a Product
3. Plataformas de Desarrollo Interno (IDPs)
4. Adopción

“Paso **más tiempo** intentando entender cómo hacer testing, build y deploy de una aplicación **que en el desarrollo real** del código que quiero añadir.”

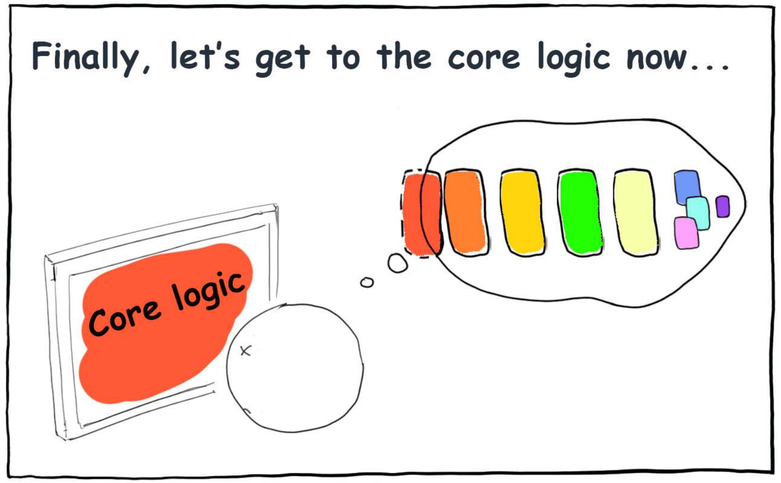
- Developer Anónimo

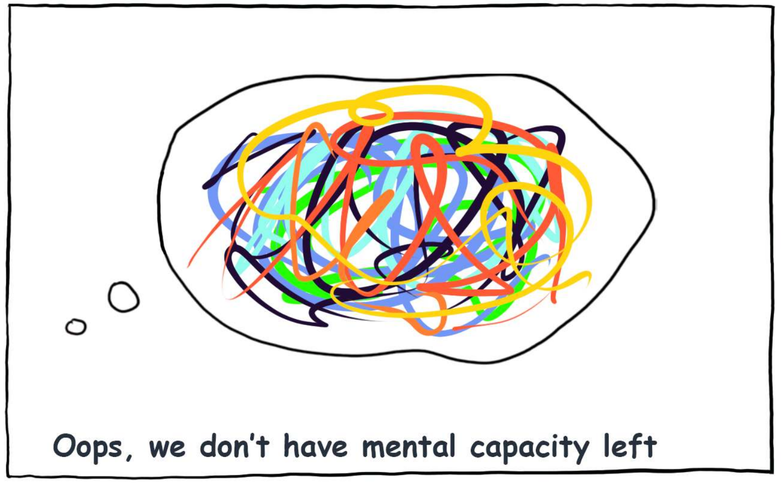
¡Platform Engineering al rescate!





4 HOURS LATER





 github.com/zakirullin

Carga Cognitiva Extrínseca

🧠++ ¿Dónde puedo encontrar la documentación completa sobre este sistema?

🧠++ ¿Hay algún estándar o plantilla que deba seguir para mi aplicación?

🧠++ ¿Hay algún repositorio específico o rama en la que deba trabajar?

🧠++ ¿Existe una suite de pruebas automatizadas o algún framework de pruebas que deba usar?

🤯 ¿Cuál es el proceso para desplegar cambios en producción?

Carga Cognitiva

Cómo defino las clases en Python. → **Carga Intrínseca**

Resolver problemas de configuración que no están directamente relacionados con la tarea principal de consumir mensajes. → **Carga Extrínseca**

Conceptualizar una arquitectura que sea escalable y pueda atender futuras expansiones o cambios en el proyecto. → **Carga Germana**

Al **minimizar la Carga Intrínseca** y **eliminar la Extrínseca**, podemos concentrarnos en la Carga Germana.

Reduce el burn out

Al **minimizar la Carga Intrínseca** y **eliminar la Extrínseca**, podemos concentrarnos en la Carga Germana.

Incrementa la motivación

Al **minimizar la Carga Intrínseca** y **eliminar la Extrínseca**, podemos concentrarnos en la Carga Germana.

Fomenta el flujo de desarrollo

Al **minimizar la Carga Intrínseca** y **eliminar la Extrínseca**, podemos concentrarnos en la Carga Germana.

Trae valor al negocio 💰

“El propósito de un Platform Team es permitir a los equipos de desarrollo entregar el trabajo con **autonomía**.

[Éste] proporciona **servicios internos** para **reducir la carga cognitiva** que se requeriría de los equipos para desarrollar estos servicios.”

- Team Topologies, Matthew Skelton and Manuel Pais

Componentes de Platform Engineering

Configuración de Aplicación

Orquestación de Infraestructura

Gestión de Ambientes de Desarrollo

Gestión de Despliegue

Control de Acceso Basado en Roles

https://internaldeveloperplatform.org/core-components/

Componentes de Platform Engineering

Configuración de Aplicación

Orquestación de Infraestructura

Gestión de Ambientes de Desarrollo

Gestión de Despliegue

Control de Acceso Basado en Roles

InfraOps?

DevOps?

SysAdmin?

https://internaldeveloperplatform.org/core-components/

“Platform Engineering no es más que **DevOps** con una **mentalidad de producto**”

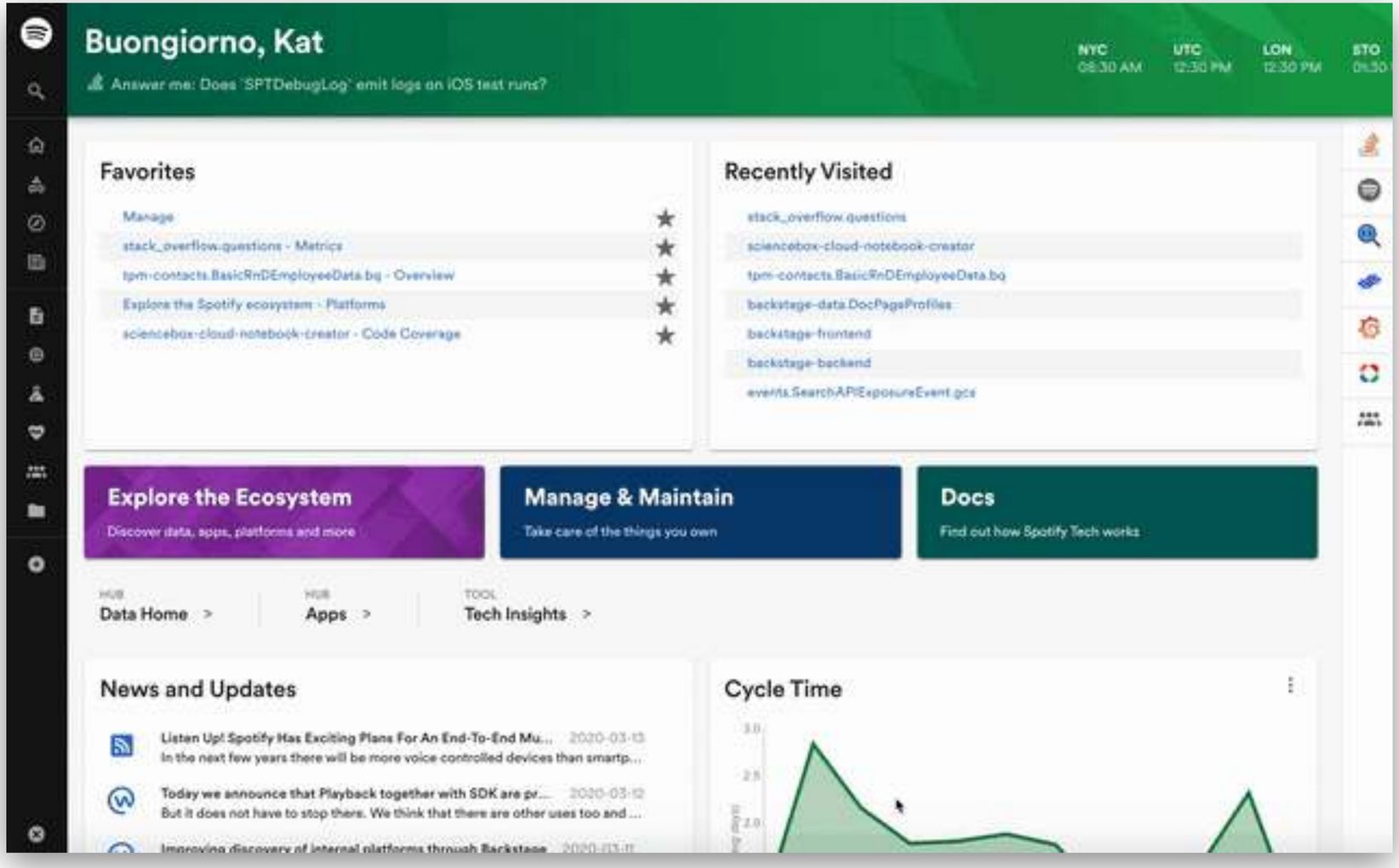
- Luca Galante

Colaboración Vs. Consumir servicios

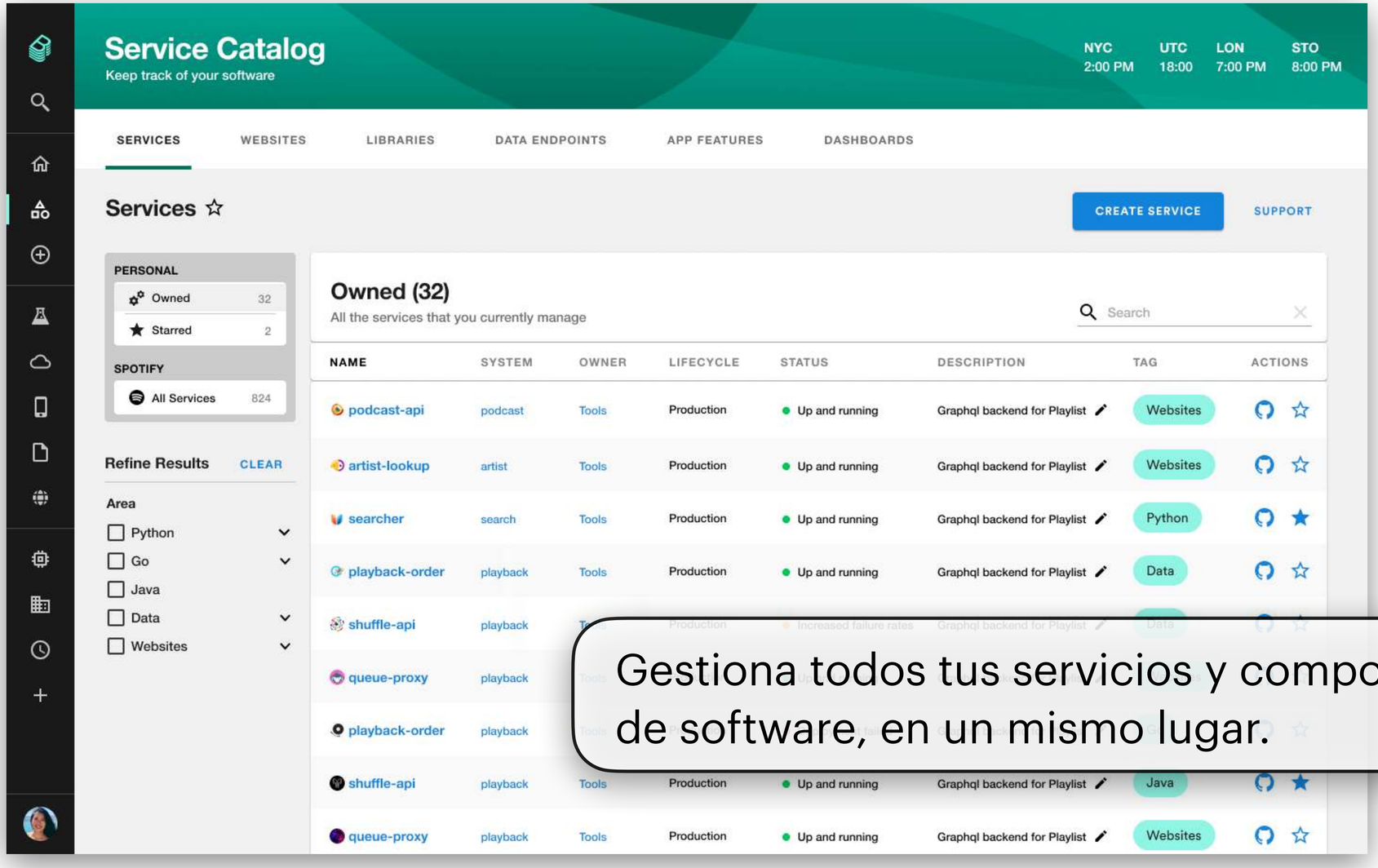
"Cierto grado de colaboración entre los equipos es esperada, pero a menudo ésta **no escala a toda la organización**. Consumir *cosas-como-servicio* suele ser **más eficaz** a medida que el número crece."

- Team Topologies, Matthew Skelton and Manuel Pais

Internal Development Platforms (IDP)

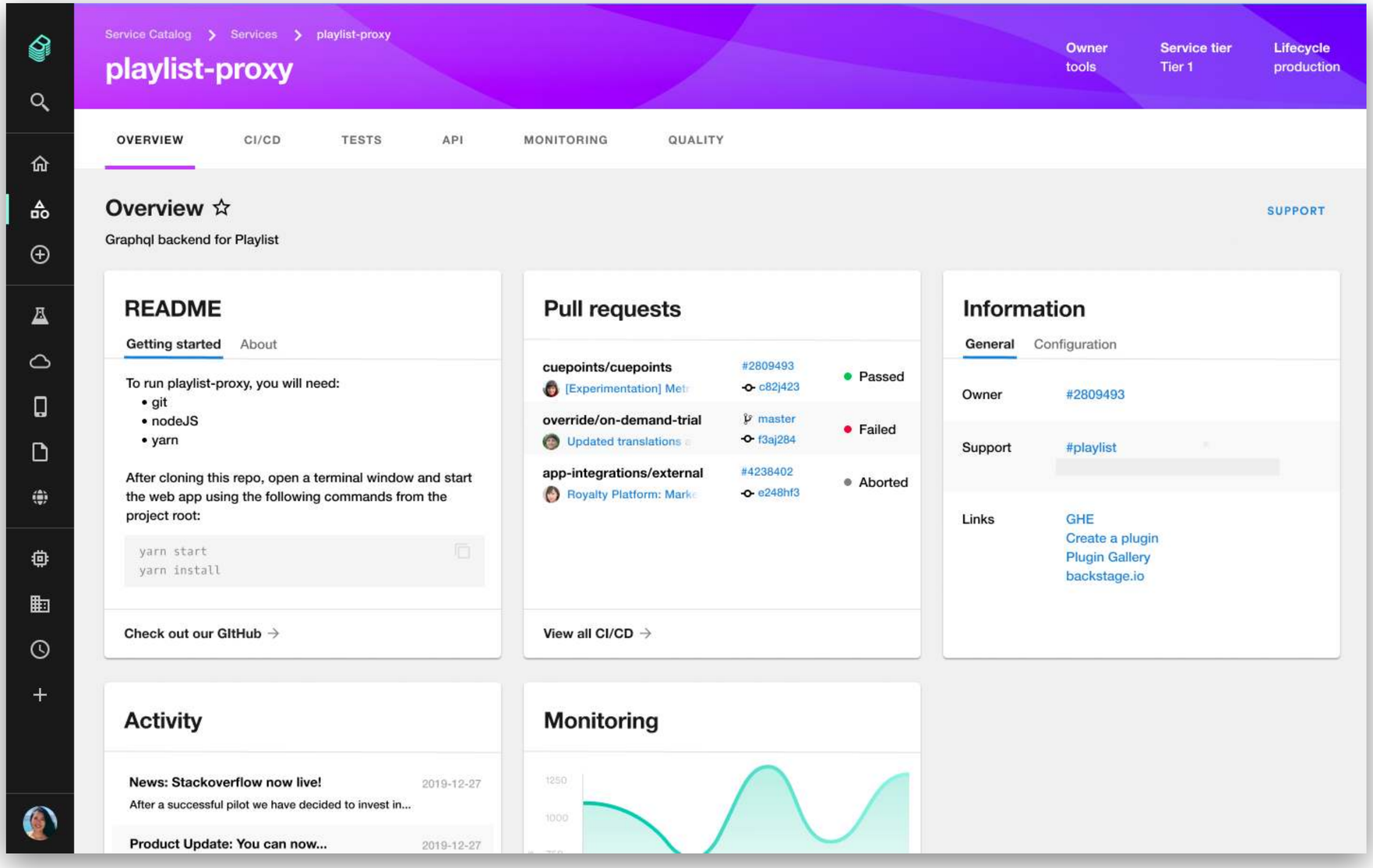


Service Catalog



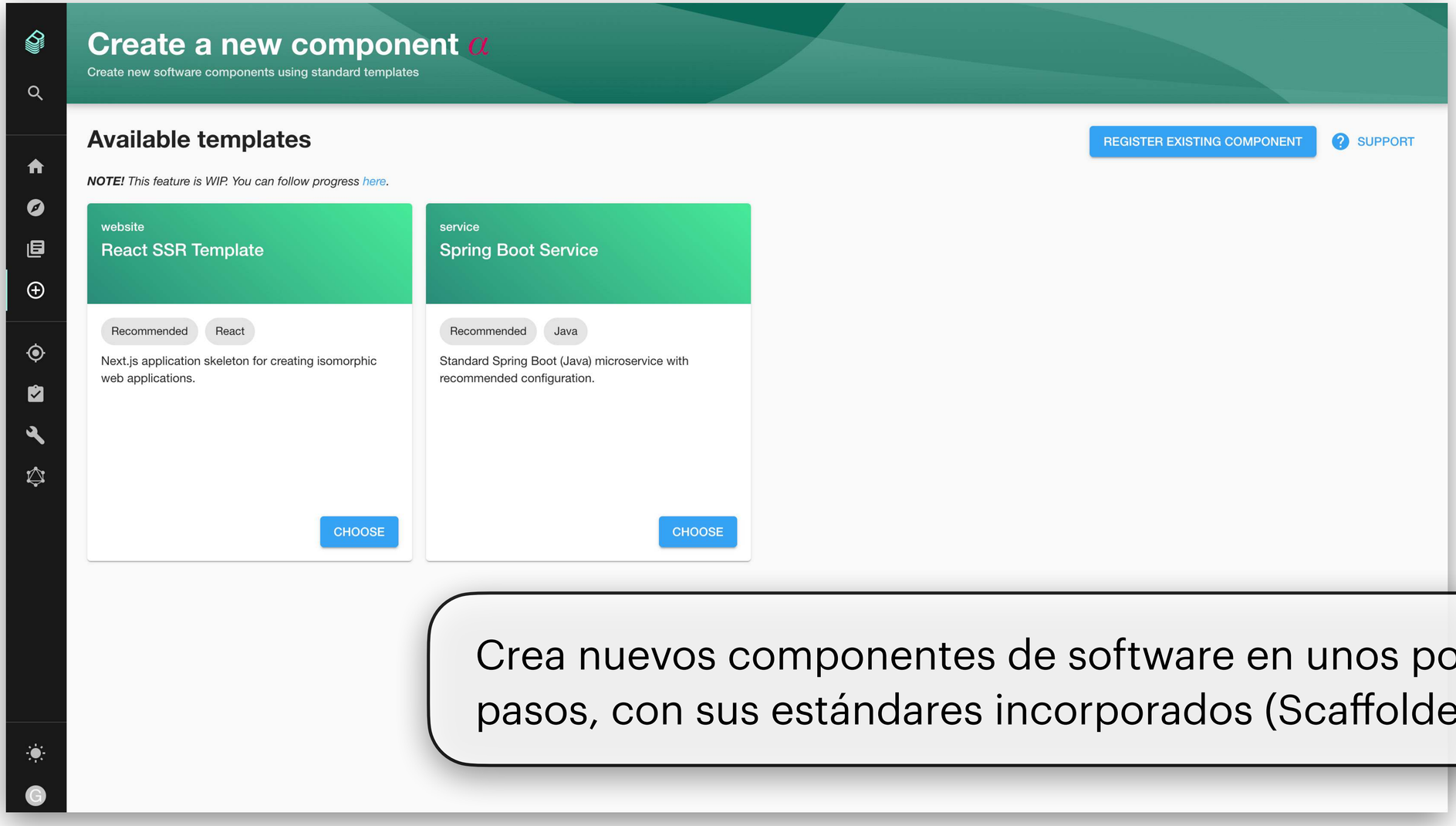
Gestiona todos tus servicios y componentes de software, en un mismo lugar.

Service Catalog

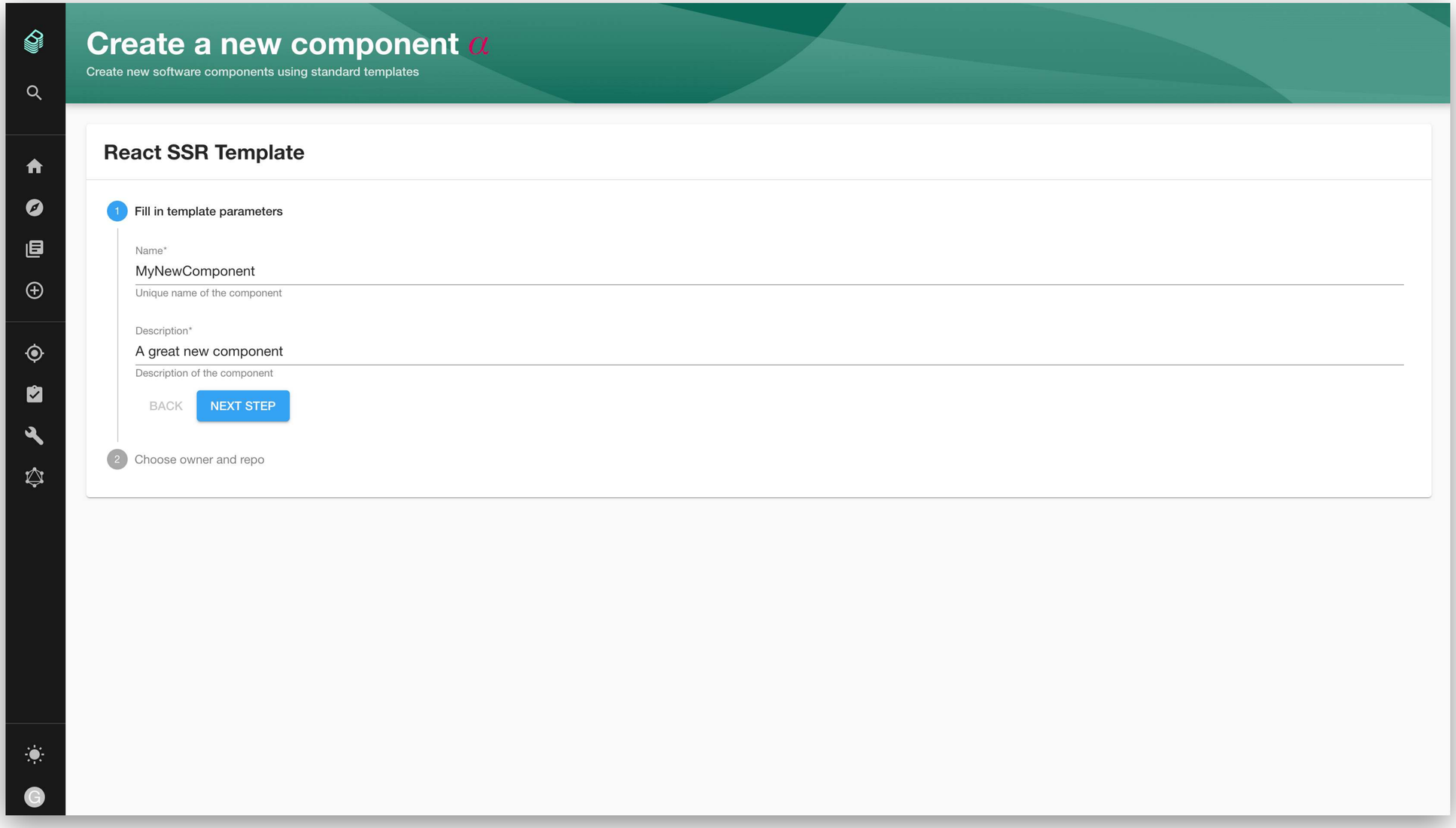


- CI/CD
- Tests
- API
- Monitoring
- Quality

Software Templates



Crea nuevos componentes de software en unos pocos pasos, con sus estándares incorporados (Scaffolder).



Software Templates

“A los desarrolladores de software les encanta construir plataformas y, sin [...] Product Management, crearán una plataforma **más grande de lo necesario**.”

- Allan Kelly, Agile

Carga Cognitiva

🧠++ ¿Dónde puedo encontrar la documentación completa sobre este sistema?

🧠++ ¿Hay algún estándar o plantilla que deba seguir para mi aplicación?

🧠++ ¿Hay algún repositorio específico o rama en la que deba trabajar?

🧠++ ¿Existe una suite de pruebas automatizadas o algún framework de pruebas que deba usar?

🤯 ¿Cuál es el proceso para desplegar cambios en producción?

Adopción

Cognitive load assessment

¿Cómo es la experiencia de construir sus servicios?
*
Cosas a considerar: ¿es la construcción una tarea clara y repetible? ¿Es lo suficientemente rápida? ¿Qué sucede cuando las construcciones fallan? ¿Son fáciles de diagnosticar los fallos?

¿Cómo es la experiencia de probar sus servicios?
*
Cosas a considerar: ¿es la prueba una tarea clara y repetible? ¿Es lo suficientemente rápida? ¿Qué sucede cuando las pruebas fallan? ¿Son fáciles de diagnosticar los fallos? ¿Los entornos de prueba son adecuados? ¿Son fáciles de acceder/iniciar/limpiar/introducir datos de prueba en los entornos de prueba?

https://github.com/TeamTopologies/Team-Cognitive-Load-Assessment

Dependencias de los equipos

Team name/focus	Depends on Team	Type (blocking/slowing/ok)	Short description of dependency (artifacts, approvals, other)

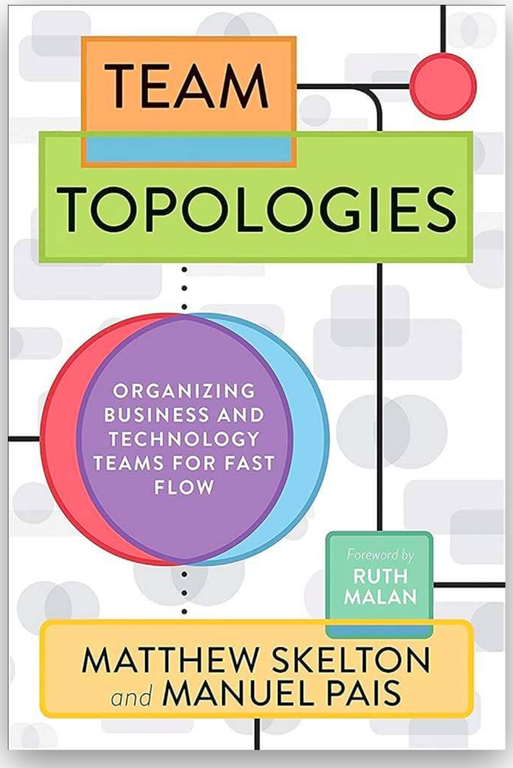
https://github.com/TeamTopologies/Team-Dependencies-Tracking

Team API

```
Fecha:
Nombre y enfoque del equipo:
Tipo de equipo:
¿Parte de una Plataforma? (s/n) Detalles:
¿Ofrecemos un servicio a otros equipos? (s/n) Detalles:
¿Qué tipo de expectativas de nivel de servicio tienen otros equipos de nosotros?
Software gestionado y evolucionado por este equipo:
Métodos de versionado:
Términos de búsqueda en la Wiki:
Canales de herramienta de chat: #_____________ #_____________ #_____________
Hora de la reunión diaria de sincronización:
```

https://github.com/TeamTopologies/Team-API-template

¡Quiero saber más!



Internal Developer Platform

PLATFORM CON 23

"**Minimizar la carga cognitiva para los demás**" es una de las heurísticas más útiles para un buen desarrollo de software.

- Team Topologies, Matthew Skelton and Manuel Pais

¡Gracias!

¿Preguntas?

👋 @alebricio

¡Gracias!

¿Preguntas?

Twitter: @alebricio

LinkedIn: /alejandrabricio