

Ralph Küpper – Skelpo Inc.

Building Microservices



About the Speaker



STARTED
PROGRAMMING AT 8
YEARS OLD



WORKED ON
WEBSITES AS EARLY AS
12 YEARS OLD



A LOT OF PHP HISTORY
(SADLY)



STARTED SKELPO AT 17
YEARS OLD



WORKED ON PROJECTS
FOR HONDA, ADIDAS,
KIA AND OTHERS

Outlook

Icebreakers

About Microservices

Preparations

User Manager

Product Manager

(break)

Order Manager

Hosting Microservices

Icebreakers

Who started working with Vapor less than two weeks ago?

A large blue icebreaker ship is shown from a low angle, sailing through white, choppy waves. The ship's hull is dark blue, and it cuts through the water, creating white spray. The background is a bright, overexposed sky.

Icebreakers

Who has built Microservices before?

A large blue silhouette of a ship, resembling an icebreaker, is positioned on the left side of the slide. It is moving from the bottom left towards the top right, leaving a wake of white and light blue spray. The background consists of white and light blue abstract shapes representing water and ice.

Icebreakers

Who has run Swift in production?

Icebreakers

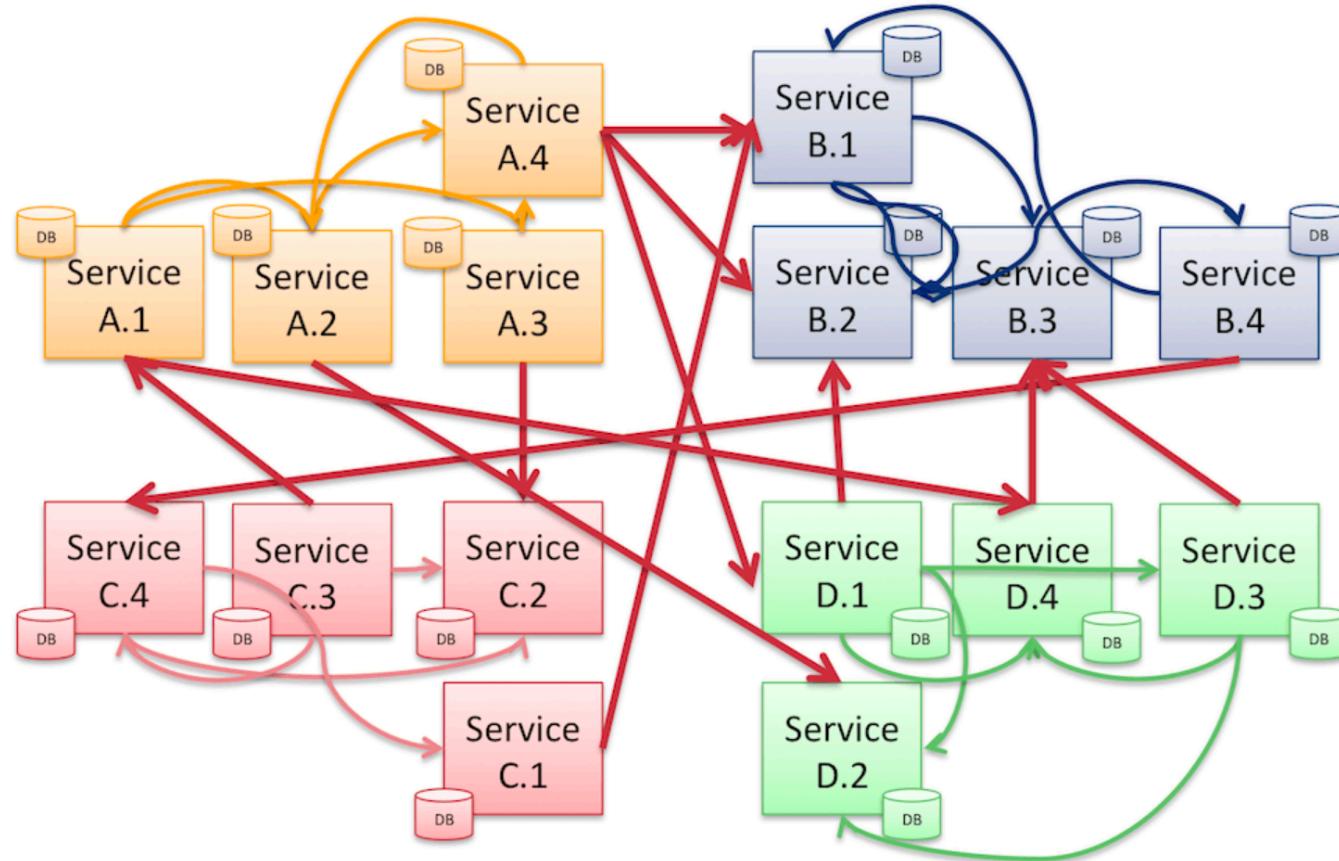
Who has set up CI/CD for Server-Side Swift before?

About Microservices

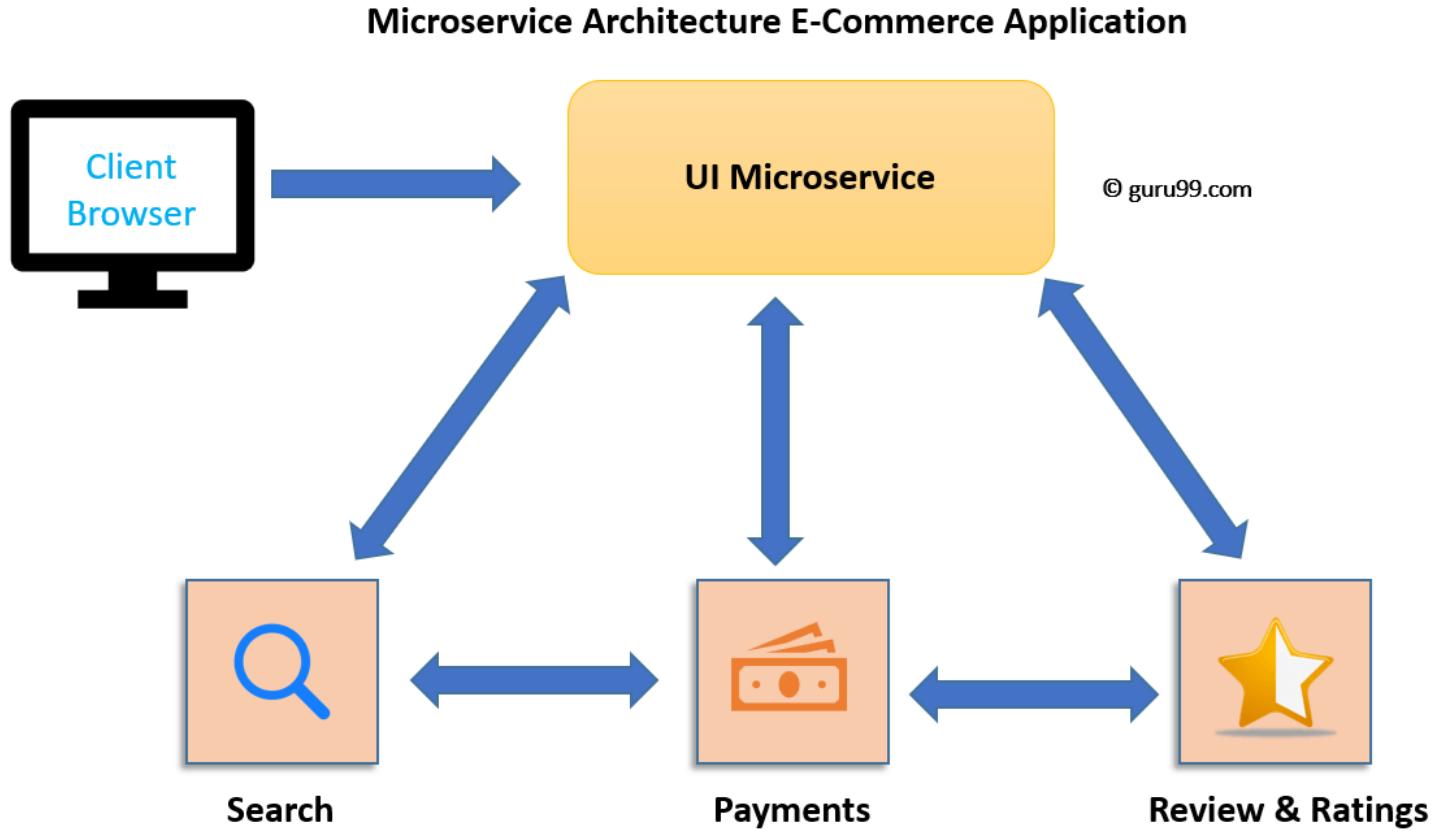
What are microservices?

- A small (=micro) service performing “one” business logic.
- A part of a group of services that operate as an application (backend).
- A self-contained service.

What are microservices?



What are
microservices?



Microservice Examples

User
Authentification

User
Management

Product
Management

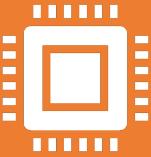
Order
Management

Shipping
Management

Status Update
Management

Mail
Management

Who is using Microservices?



Apple, IBM, Amazon, Capital One, Netflix, Uber, Airbnb, Facebook, Twitter, Pinterest, Instagram, Slack



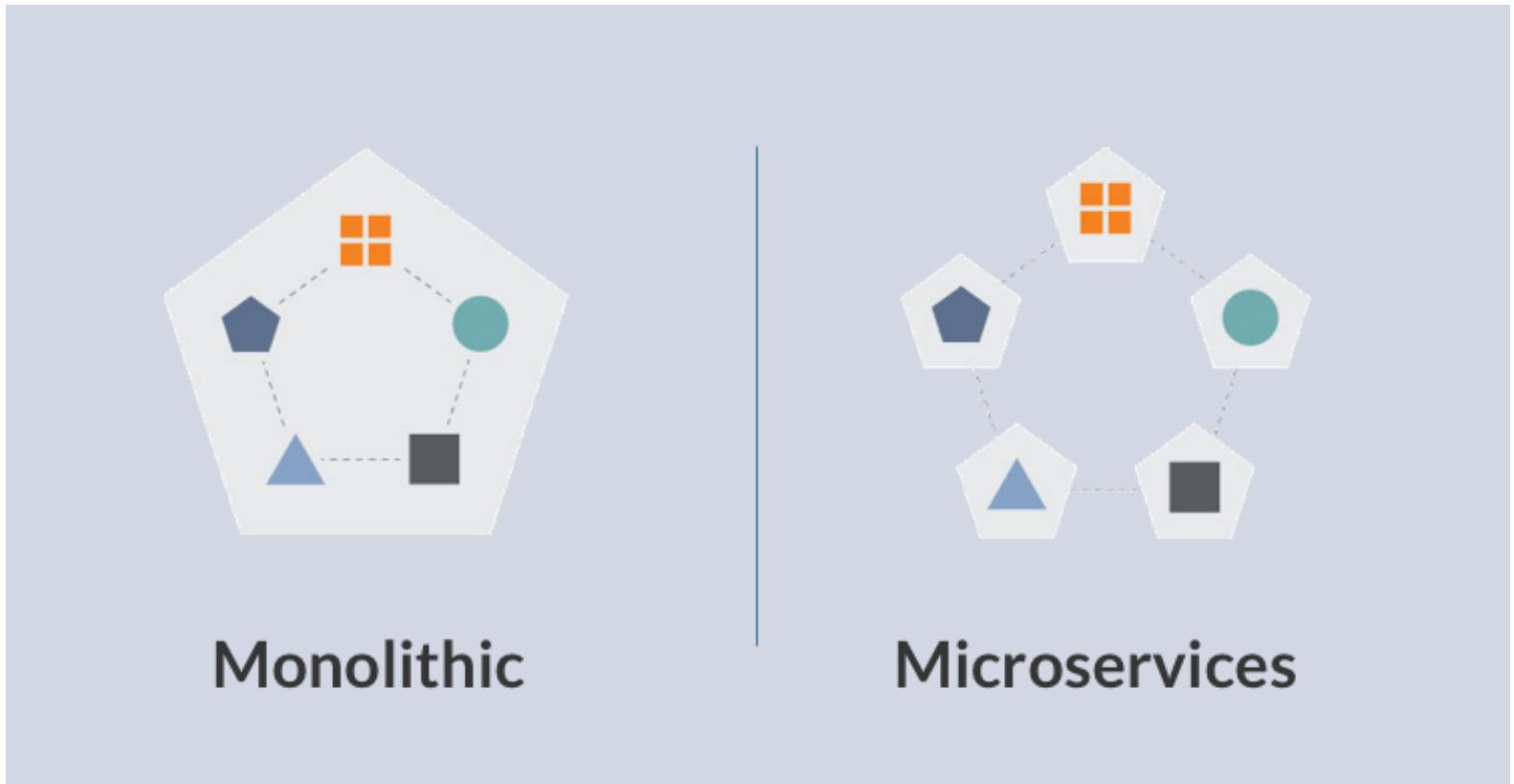
Most of them started out of monolithic applications

Don't make that mistake ;)



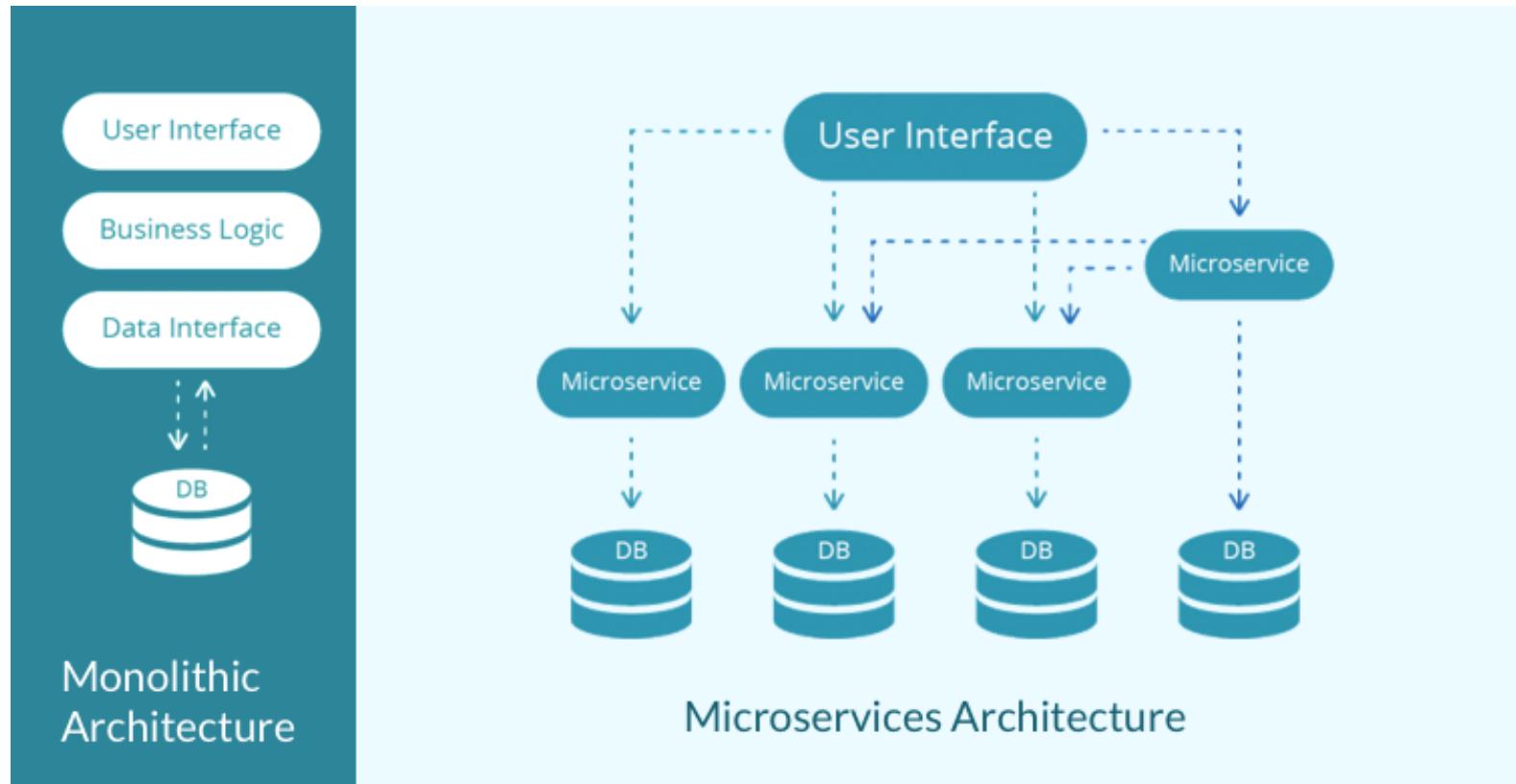
Every application that wants to be scalable should implement a microservice architecture.

Microservice vs Monolith



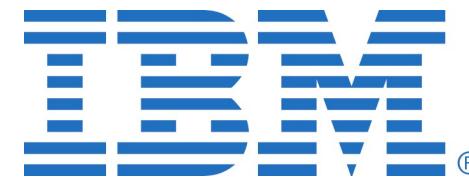
Source: <https://dzone.com/articles/monolithic-vs-microservices-which-is-the-better-ar>

Microservice VS Monolith



Source: <https://dzone.com/articles/monolithic-vs-microservices-which-is-the-better-ar>

Who is using Swift Microservices?



Microservices – An interesting subject



Interdiscipline of Software Development and DevOps



Endless possibilities and approaches



You need to make choices:

What programming language for what service?

How and if microservices need to communicate with each other?

Authentification and user management?



All this to say => ...

Your expectation?

Common Issues related to Microservices



AUTHENTICATION



CI/CD



LOCAL DEVELOPMENT

JWT



Decent solution for many cases



As secure or insecure as you make it

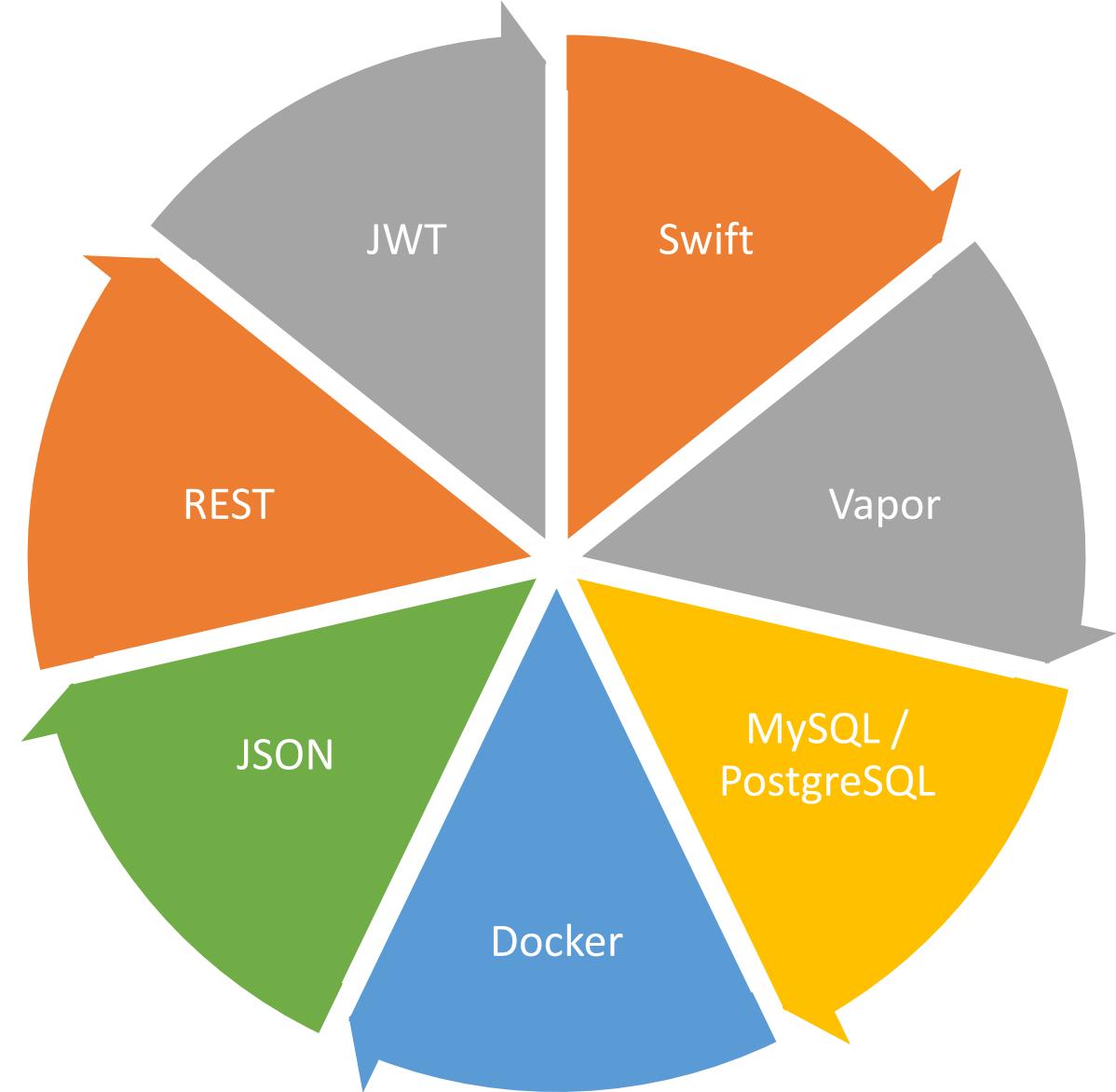


A good starting point

Central authentication can always be added on top of it.

We are
looking an
Online Shop!

The Stack:



Let's build microservices!



We are building the oversimplified API for an ecommerce app.



We need
three
services:

User Manager
Product Manager
Order Manager



The client is (almost) entirely irrelevant for this exercise.

Assumptions

None of this is production ready

You may like another part of the stack better, you can change it later. ☺

It's supposed to help you learn the principles, not necessarily concrete implementations.

You may have better idea's than what I'm suggesting => Speak up! ☺

Preparations

- Download the following Git repository: <http://tiny.cc/es77ez>
- Open Terminal and navigate to the folder of your repository.
- Verify you have Swift 5.1+ installed (“swift –version”)
- Verify you have Docker 18+ installed (“docker –version”)
- Verify you have docker-compose 1.20+ installed (“docker-compose –version”)
- Verify you have Vapor Toolbox 3.1+ installed (“vapor –version”)
- If not: Go ahead and install whatever you are missing.

Let's start with one central piece: The User Service



A lot of apps need user management

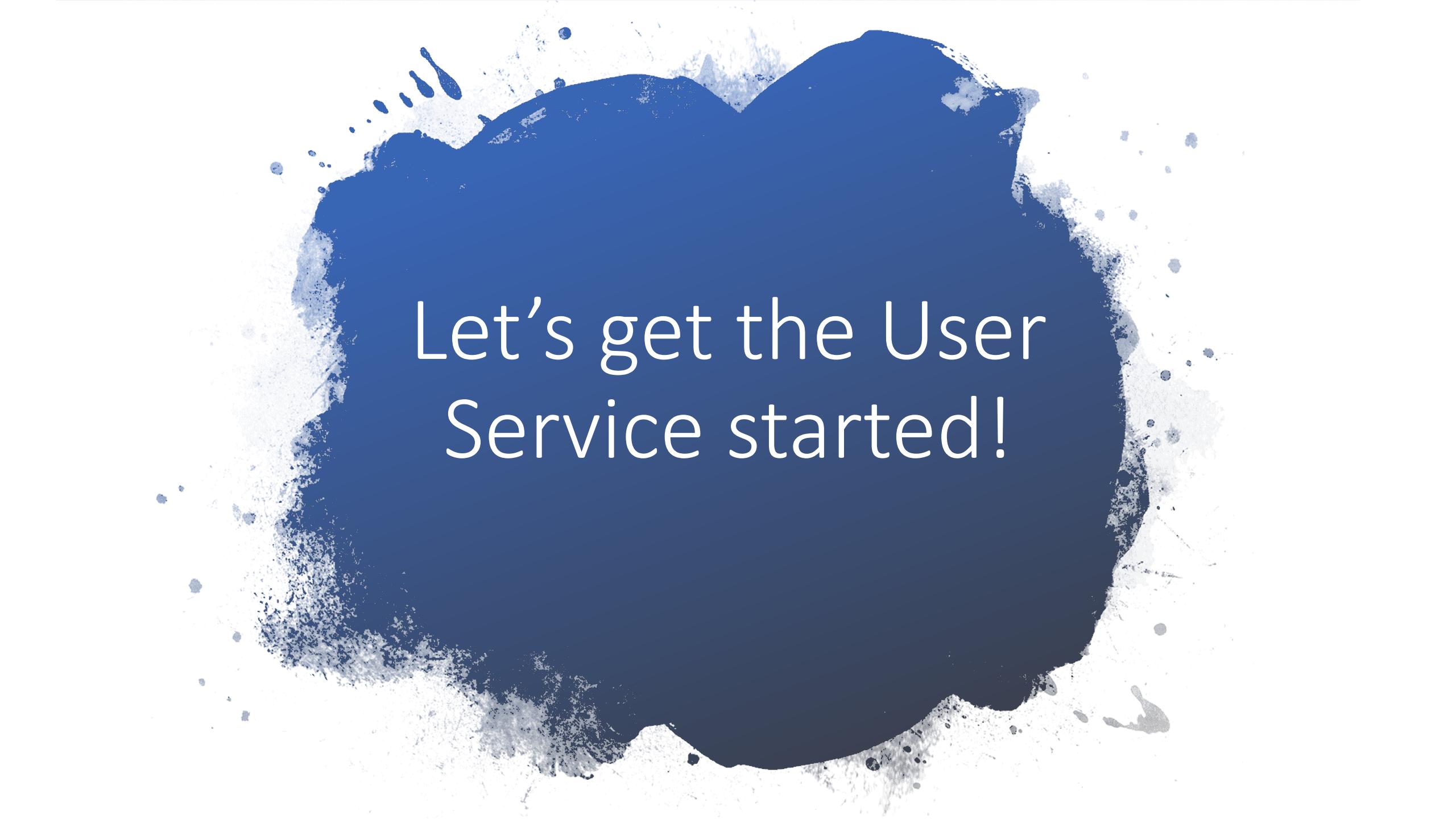


It's central and lays out the ground rules



It's re-usable.

Let's review a
microservice template!



Let's get the User
Service started!



Let's get the product
service started

Quick Review

- We should have 2 services up and running now
- They represent half of the API already.
- The services do not interact with each other yet.
- A frontend could now offer easily:
 - User Login, Registration etc.
 - Product overview, categories

What's next?



We need to be able to order products.



What do we need?

Discussion

Let's work on the connection

- Open up the Order Service
- Write the needed code ☺

A photograph of a black coffee cup and saucer filled with dark coffee. A thin plume of white steam rises from the coffee. The cup has a black handle and a thin gold-colored rim. It sits on a matching black saucer. The background is a solid dark brown. The entire image is framed by a thick white border.

Coffee Break

Seeing CI in Action



Questions,
Discussion
and more

Thank you for
having me

ralph@skelpo.com