



Microservices for the Masses with Spring Boot, Angular, and JHipster

Matt Raible | [@mraible](https://twitter.com/mraible)

October 22, 2018

Photo by Jeff Krause <https://www.flickr.com/photos/jeffkrause/8498448553>



Do you think microservices are hard?







Hi, I'm Matt Raible!

Father, Husband, Skier, Mountain
Biker, Whitewater Rafter

Open Source Connoisseur

Web Developer and Java Champion

Okta Developer Advocate



Bus Lover

Blogger on raibledesigns.com and
developer.okta.com/blog





Allison White
Photography



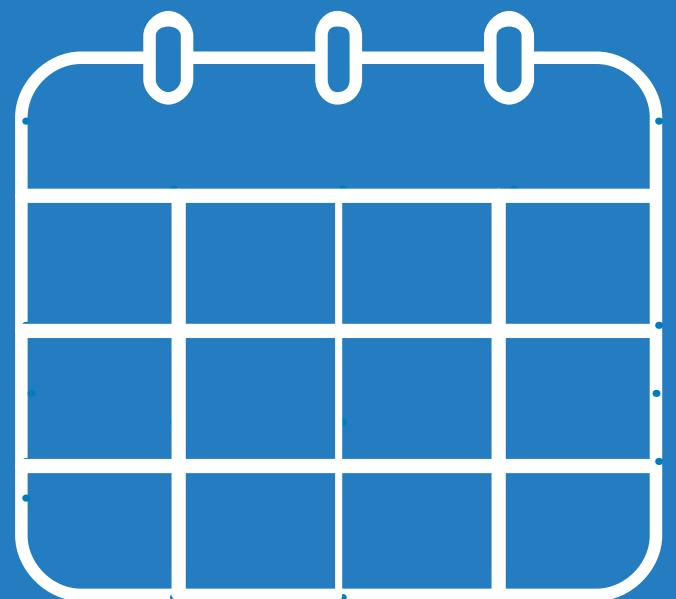
{okta}

developer.okta.com

What About You?

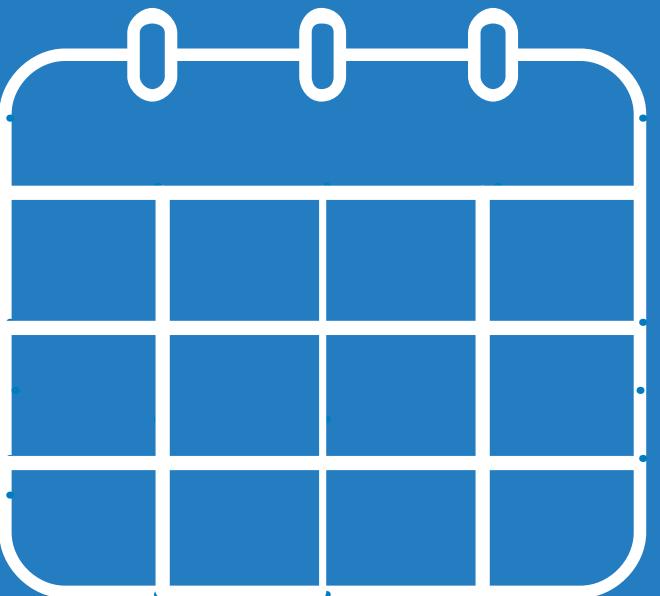


Agenda



1. Introduction to Microservices
2. Microservices with JHipster
3. Deploying to the Cloud
4. Developing Mobile Apps with JHipster
5. JHipster Roadmap

Part 1



Introduction to Microservices

History of Microservices

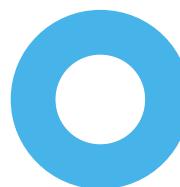
Microservices Architecture Philosophy

Why Microservices?

Demo: A Microservices Architecture with
Spring Boot and Spring Cloud

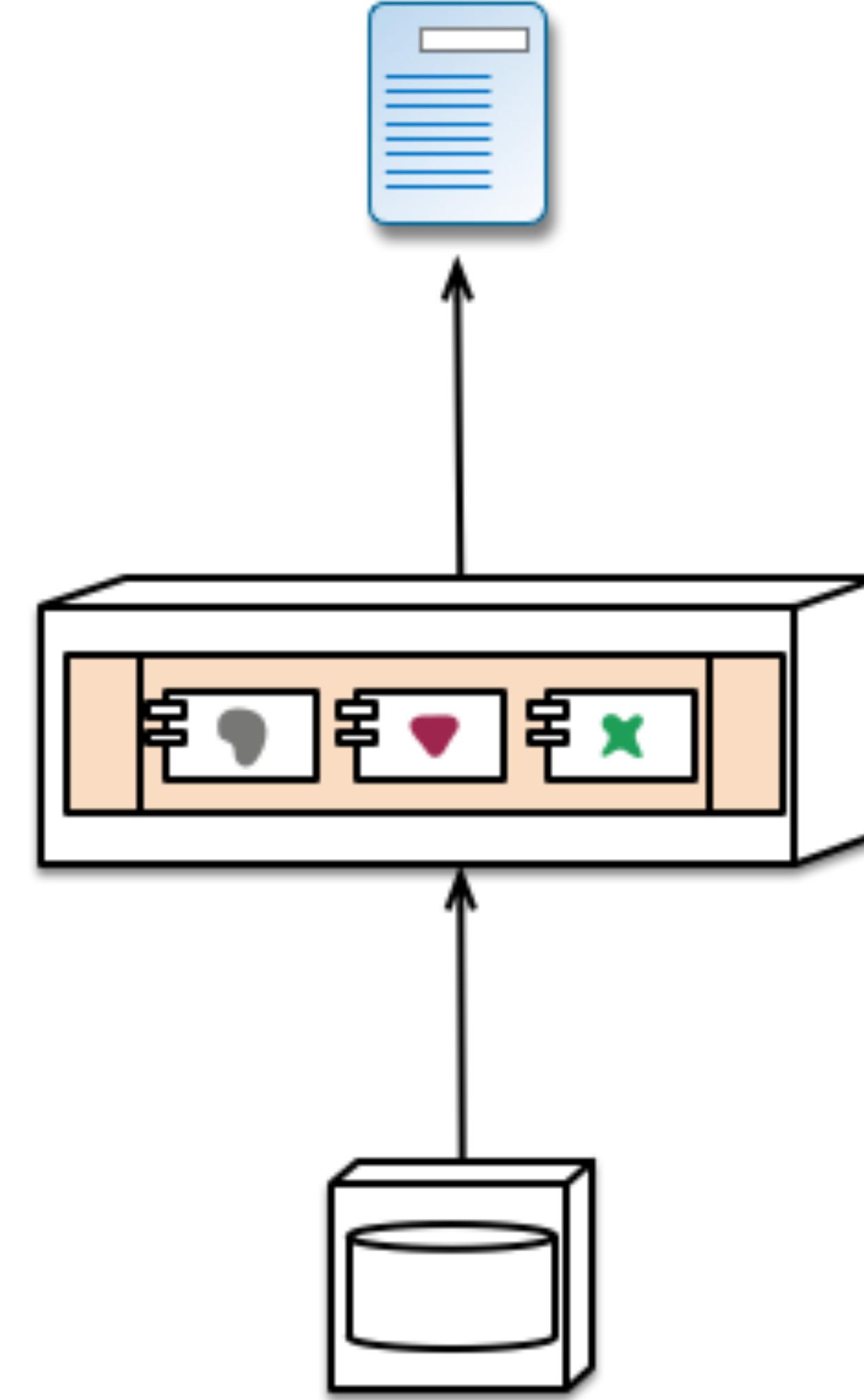


Microservices Visionaries





Siloed functional teams...



... lead to siloed application architectures.
Because Conway's Law

Conway's Law

“Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.”

Melvin Conway 1967



“Do one thing and do it well.”

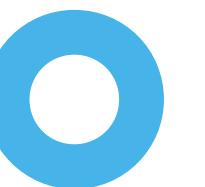
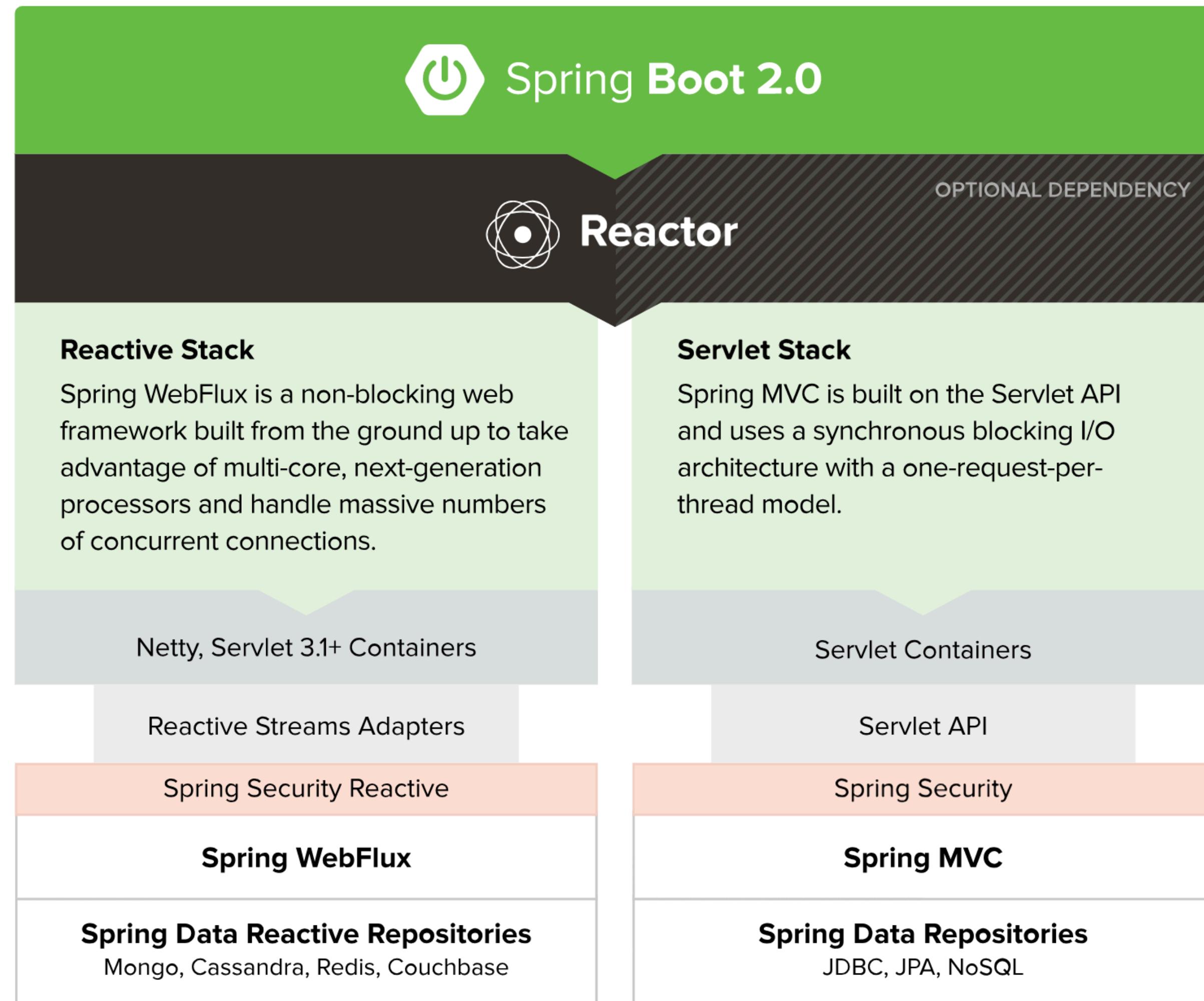




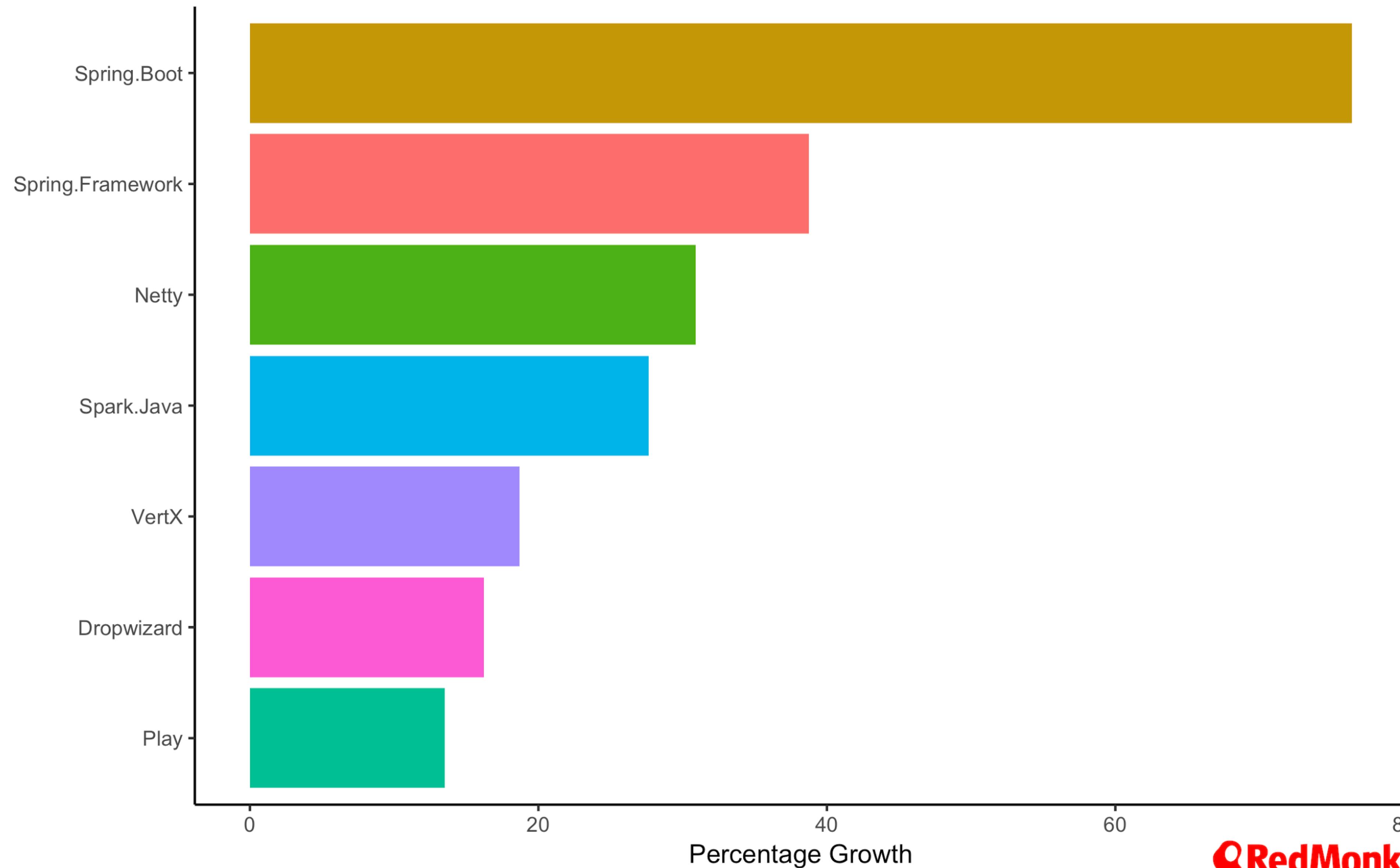
“You shouldn't start with a microservices architecture. Instead begin with a monolith, keep it modular, and split it into microservices once the monolith becomes a problem.”

Martin Fowler March 2014





Java Frameworks
Percentage Growth, Github Stars
Sep 2016 - June 2017



A close-up photograph of a dense field of vibrant green grass. The blades are long, thin, and slightly curved, creating a textured pattern across the frame. The lighting is bright, highlighting the freshness of the grass.

start.spring.io

Spring Initializr x +

https://start.spring.io

SPRING INITIALIZR bootstrap your application now

Generate a Maven Project with Java and Spring Boot 2.0.6

Project Metadata

Artifact coordinates

Group

Artifact

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

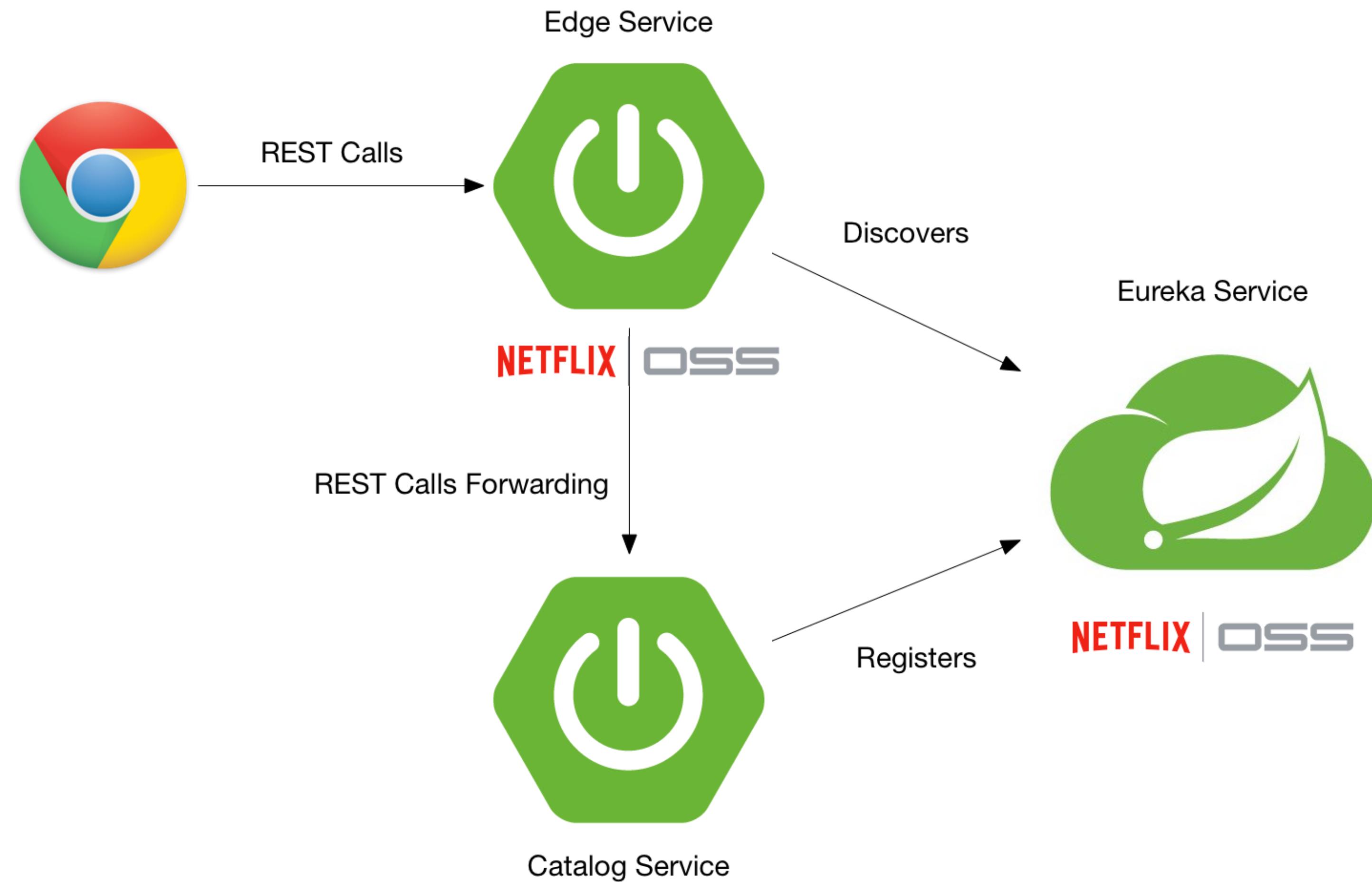
Selected Dependencies

Generate Project ⌘ + ↵

Don't know what to look for? Want more options? [Switch to the full version.](#)

start.spring.io is powered by [Spring Initializr](#) and [Pivotal Web Services](#)

Microservices with Spring Boot



Demo



Using start.spring.io, create:

A service registry

A gateway

A catalog service

Create an endpoint in the catalog service

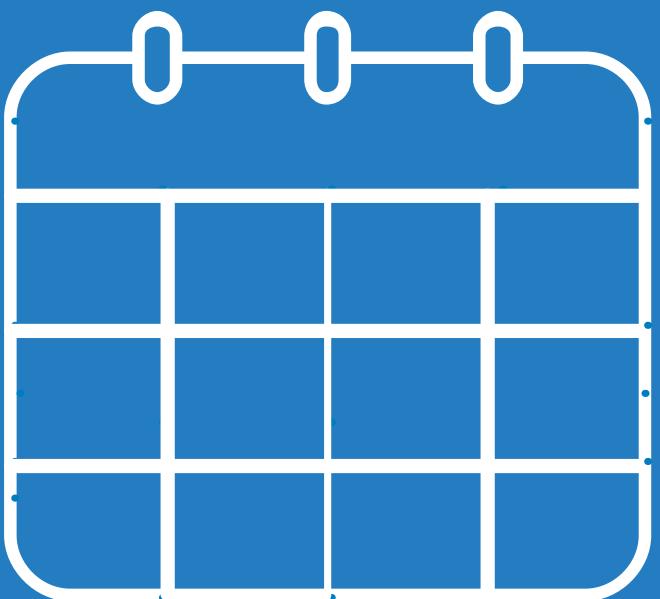
Create a filtered endpoint in the gateway

Show failover capabilities

Show Spring Security OAuth

<https://github.com/oktadeveloper/spring-boot-microservices-example>

Part 2



Microservices with JHipster

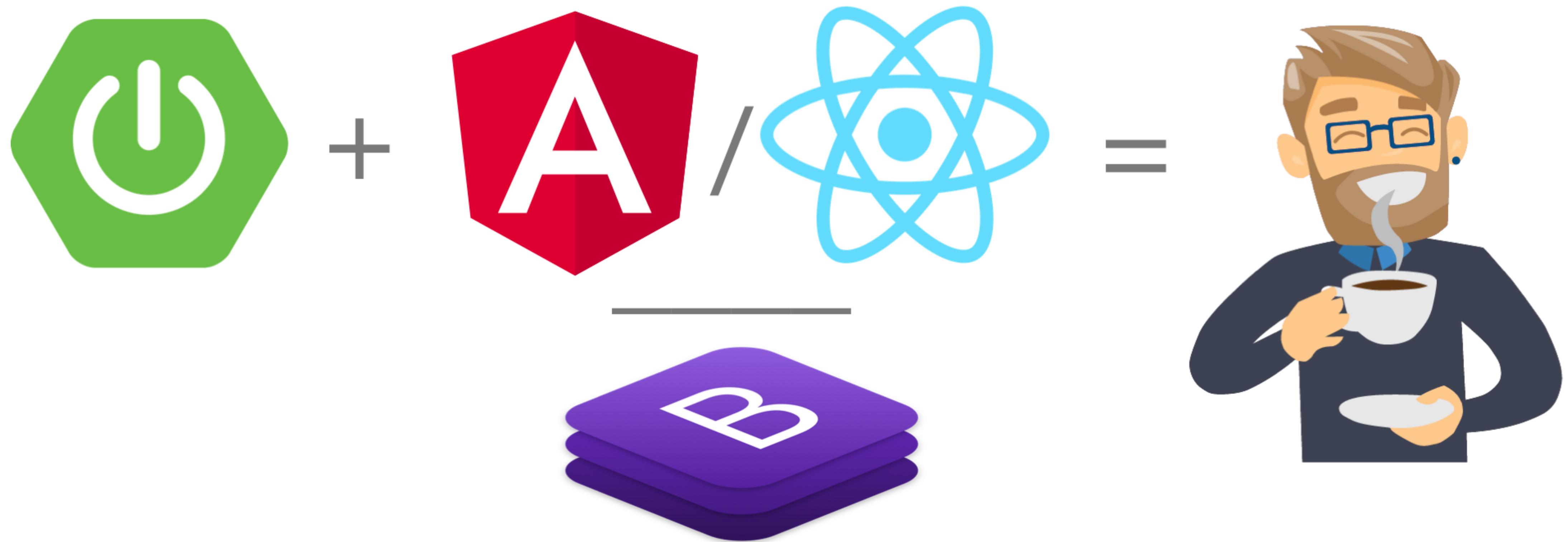
What is JHipster?

Installing and Using JHipster

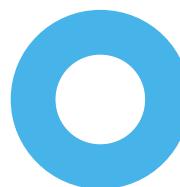
JHipster's Microservice Features

Progressive Web Applications Overview

What is JHipster?



<https://www.jhipster.tech>



JHipster Goals

A high-performance and robust Java stack on the server side with Spring Boot

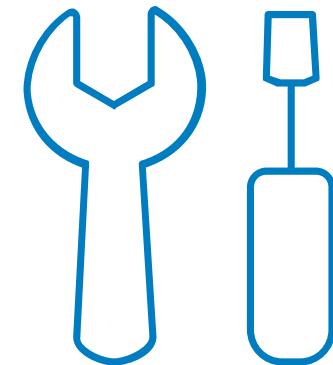
A sleek, modern, mobile-first front-end with Angular and Bootstrap

A robust microservice architecture with JHipster Registry, Netflix OSS, Elastic Stack, and Docker

A powerful workflow to build your application with Yeoman, Webpack/Gulp and Maven/Gradle

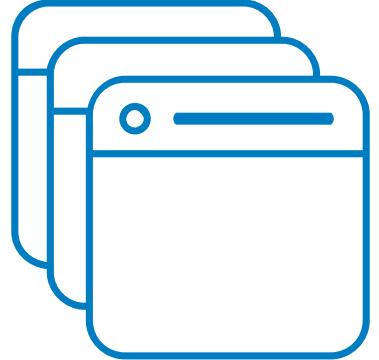


How to Use JHipster



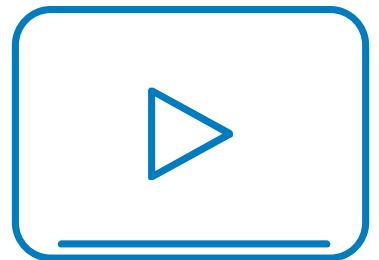
Install JHipster and Yeoman, using npm:

```
npm install -g generator-jhipster
```



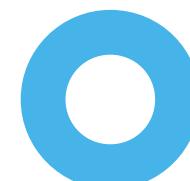
Create a directory and cd into it:

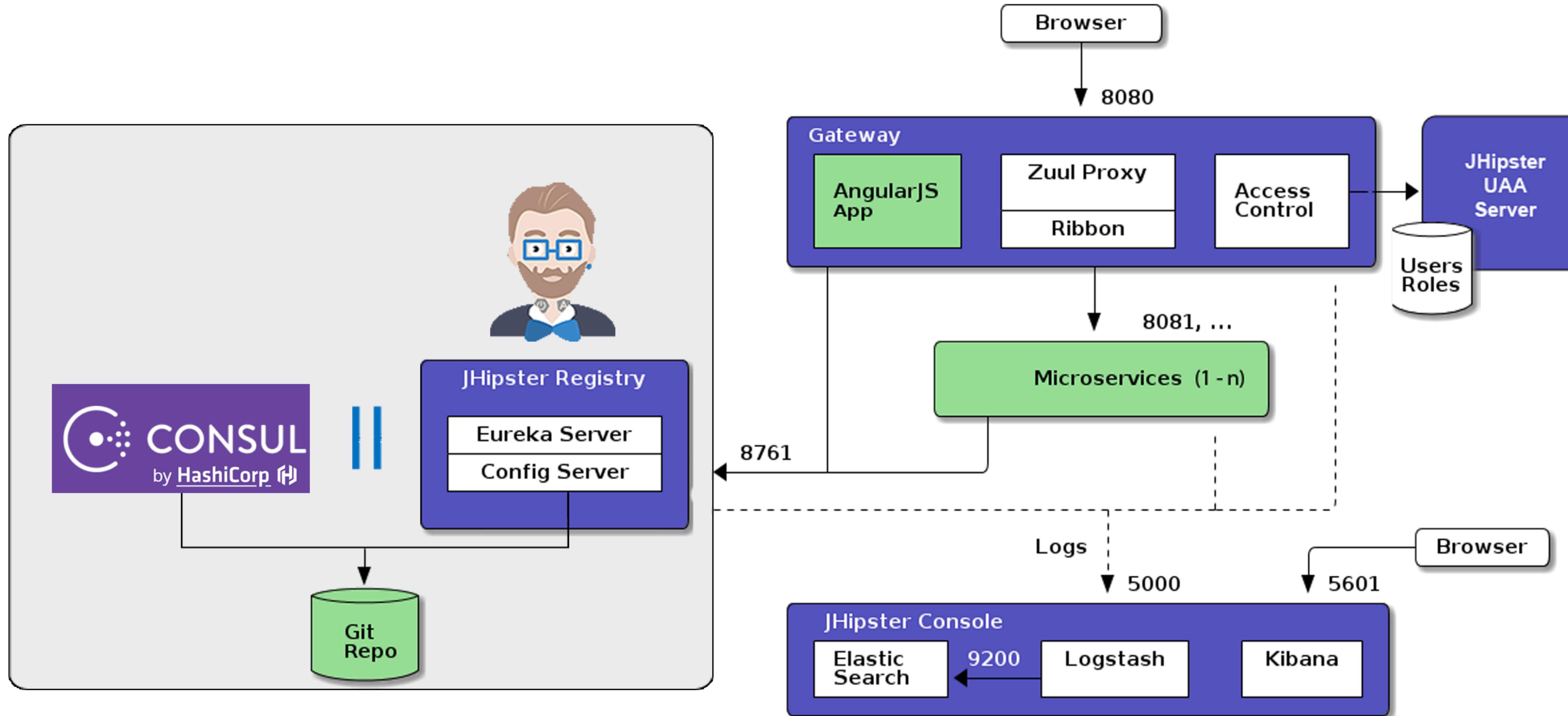
```
mkdir newapp && cd newapp
```



Run it!

```
jhipster
```

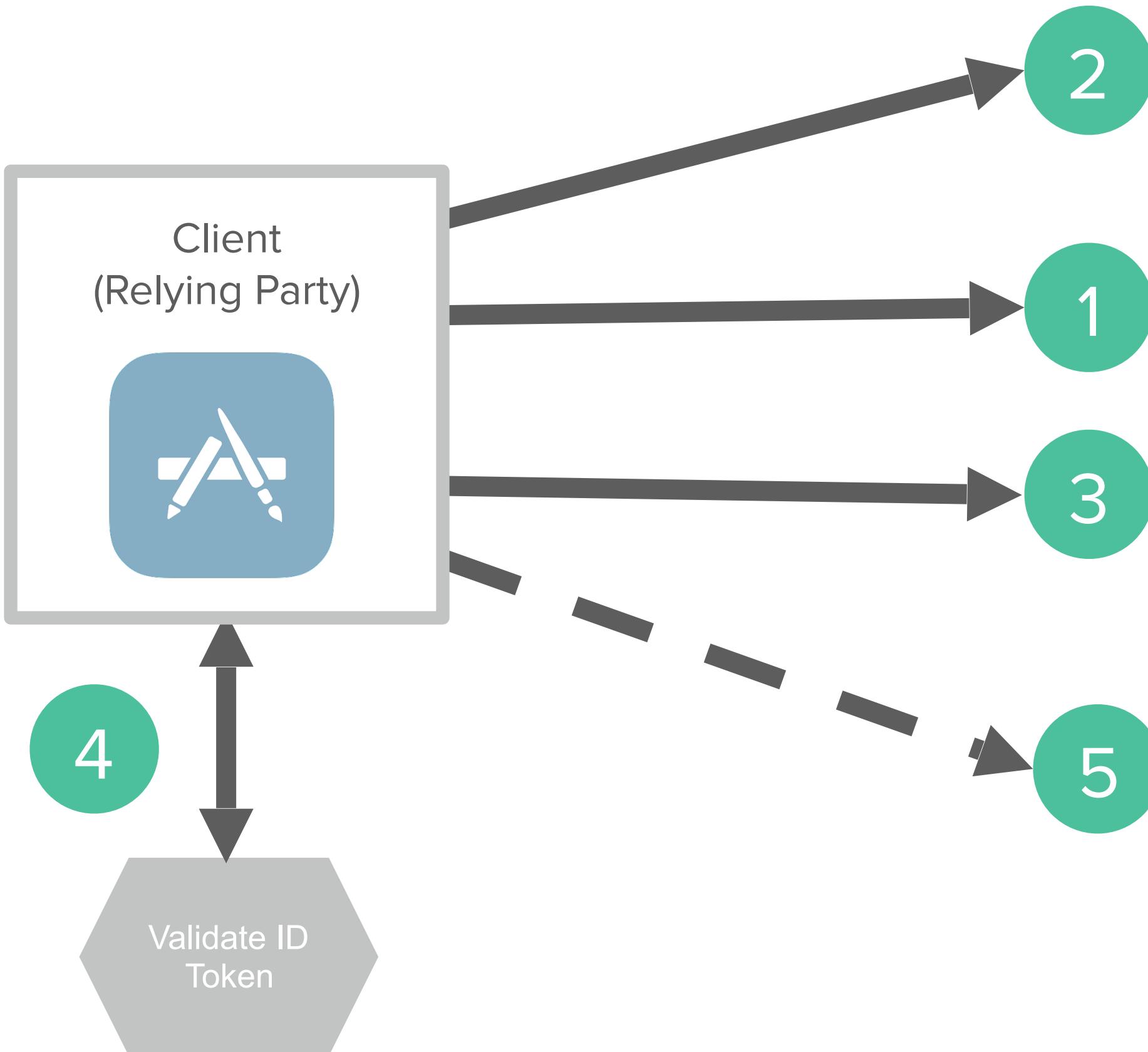




<https://www.jhipster.tech/microservices-architecture>

OAuth 2.0 and OIDC

- 1 Discover OpenID Provider Metadata
- 2 Perform OAuth flow to obtain a **ID token** and/or **access token**
- 3 Get JSON Web Key Set (JWKS) for signature keys
- 4 Validate **ID token** (JSON Web Token)
- 5 Get additional user attributes with **access token** from UserInfo endpoint



OAuth 2.0 Authorization Server & OpenID Connect Provider (OP)

Token Endpoint

Authorization Endpoint

/well-known/
openid-configuration

JWKS Endpoint

Userinfo Endpoint

OAuth 2.0 Resource Server

Monolith Examples

Development Blog v0.0.1-SNAPSHOT

Home Entities Administration Language Account

Welcome, Java Hipster!

This is your homepage

You are logged in as user "admin".

If you have any question on JHipster:

- [JHipster homepage](#)
- [JHipster on Stack Overflow](#)
- [JHipster bug tracker](#)
- [JHipster public chat room](#)
- [follow @java_hipster on Twitter](#)

If you like JHipster, don't forget to give us a star on [GitHub](#)!

This is your footer

Register a new account'."/>

21-Points Health v6.0.1

Home Language Account About

Welcome!

21-Points Health is here to track your health and improve your life. 😊

To get started, please ➔ [Sign In](#)

Learn more about [21-Points Health](#)

If you have any questions about the JHipster Mini-Book or 21-Points Health:

- [JHipster Mini-Book Blog](#)
- [JHipster on Stack Overflow](#)
- [Contact @java_hipster on Twitter](#)
- [Report an issue](#)

Don't have an account yet? [Register a new account](#).

JHipster 5 Demo

github.com/mraible/jhipster5-demo

youtu.be/-VQ_SVkaXbs

21-Points Health

github.com/mraible/21-points

infoq.com/minibooks/jhipster-mini-book



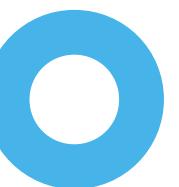
Progressive Web Apps

Originate from a **secure origin**, load while **offline**, and reference a **web app manifest**.



Progressive Web Apps

Can be **installed** on your mobile device, look and act like a **native** application, but are distributed through the **web**.



Progressive Web Apps

Are **fast!**



Enable PWA in JHipster

gateway/src/main/webapp/index.html

```
<script>
  if ('serviceWorker' in navigator) {
    window.addEventListener('load', function() {
      navigator.serviceWorker.register('/service-worker.js')
        .then(function () {
          console.log('Service Worker Registered');
        });
    });
  }
</script>
```



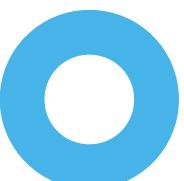
Force HTTPS in Spring Boot

gateway/src/main/java/com/okta/developer/gateway/config/OAuth2SsoConfiguration.java

```
@Configuration
public class OAuth2SsoConfiguration extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http.requiresChannel()
            .requestMatchers(r -> r.getHeader("X-Forwarded-Proto") != null)
            .requiresSecure();
    }
}
```

developer.okta.com/blog/2018/07/30/10-ways-to-secure-spring-boot



Demo



Using JHipster, create:

A gateway

A store microservices app

A blog microservices app

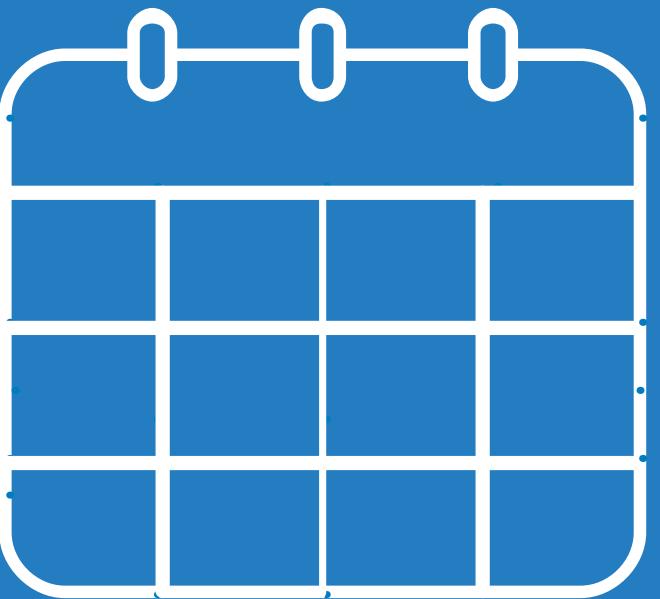
Generate entities in apps and on gateway

Convert gateway to be a PWA

Run everything in Docker

<https://github.com/oktadeveloper/okta-jhipster-microservices-oauth-example>

Part 3



Deploy to the Cloud

Options for Deploying JHipster

Heroku

Cloud Foundry

AWS

Google Cloud

Microsoft Azure



For monoliths:

jhipster heroku

For microservices:

Deploy JHipster Registry

Build and deploy microservice

Build and deploy gateway

<http://bit.ly/heroku-jhipster-microservices>



For monoliths:

jhipster cloudfoundry

For microservices:

Deploy JHipster Registry

Build and deploy microservice

Build and deploy gateway

<http://www.jhipster.tech/cloudfoundry>



Using Elastic Container Service

jhipster aws-containers

Using Elastic Beanstalk

jhipster aws

Boxfuse

boxfuse run -env=prod

<http://www.jhipster.tech/aws>

<http://www.jhipster.tech/boxfuse>



Google Cloud Platform

`mvn package -Pprod jib:dockerBuild`

`jhipster kubernetes`

`./kubectl-apply.sh`

`kubectl get svc gateway`

<https://developer.okta.com/blog/2017/06/20/develop-microservices-with-jhipster>

Demo



Build Your JHipster App for Production

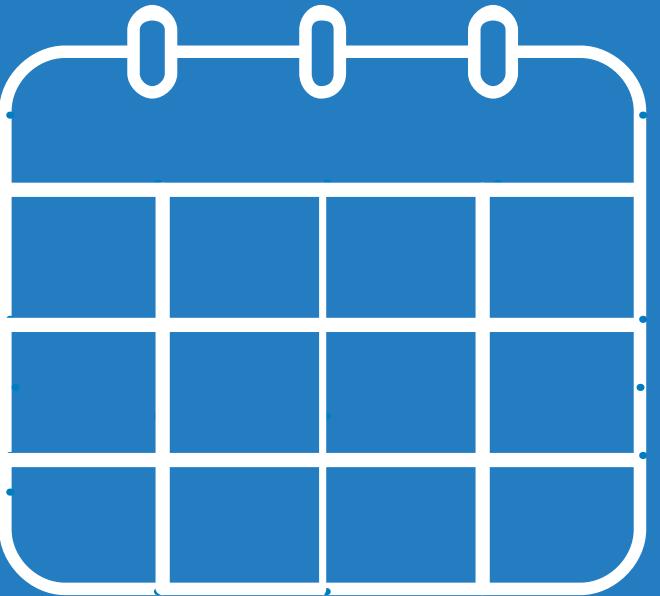
Deploy with Kubernetes and Minikube

Deploy to Google Cloud

[https://github.com/oktadeveloper/](https://github.com/oktadeveloper/jhipster-microservices-example)

jhipster-microservices-example

Part 4



Developing Mobile Apps with JHipster

What is Ionic?

Why?

Ionic Module for JHipster

JWT and OIDC Support

Entity Generator

Ionic



Ionic Framework

Develop Hybrid & PWA
Apps

<https://ionicframework.com>



Stencil

Vanilla Web
Components

<https://stenciljs.com>



PWA Toolkit

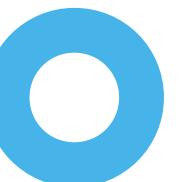
Lightning fast PWAs
<https://github.com/ionic-team/ionic-pwa-toolkit>



Why?

The first version of 21-Points Health I wrote with JHipster 2.x was **painful** to use on a mobile device. Versions 4.x and 5.x are better, but still not great.

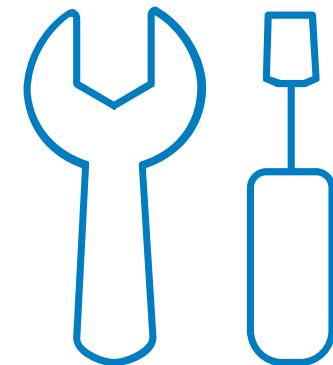
I want to develop the **best user experience**. Native apps are painful to distribute, but work better than PWAs (on iOS).



How is a mobile app for JHipster
different from its Angular UI?

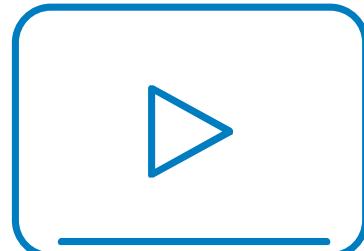


Ionic Module for JHipster



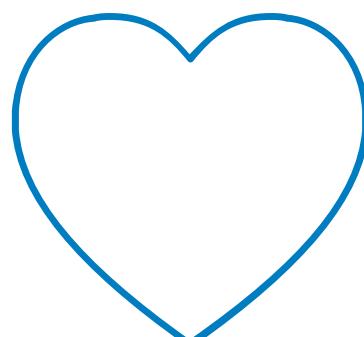
Install Ionic and the Ionic Module for JHipster, using npm:

```
npm install -g ionic generator-jhipster-ionic
```



Run it!

```
yo jhipster-ionic
```

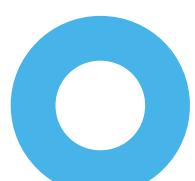


Profit!

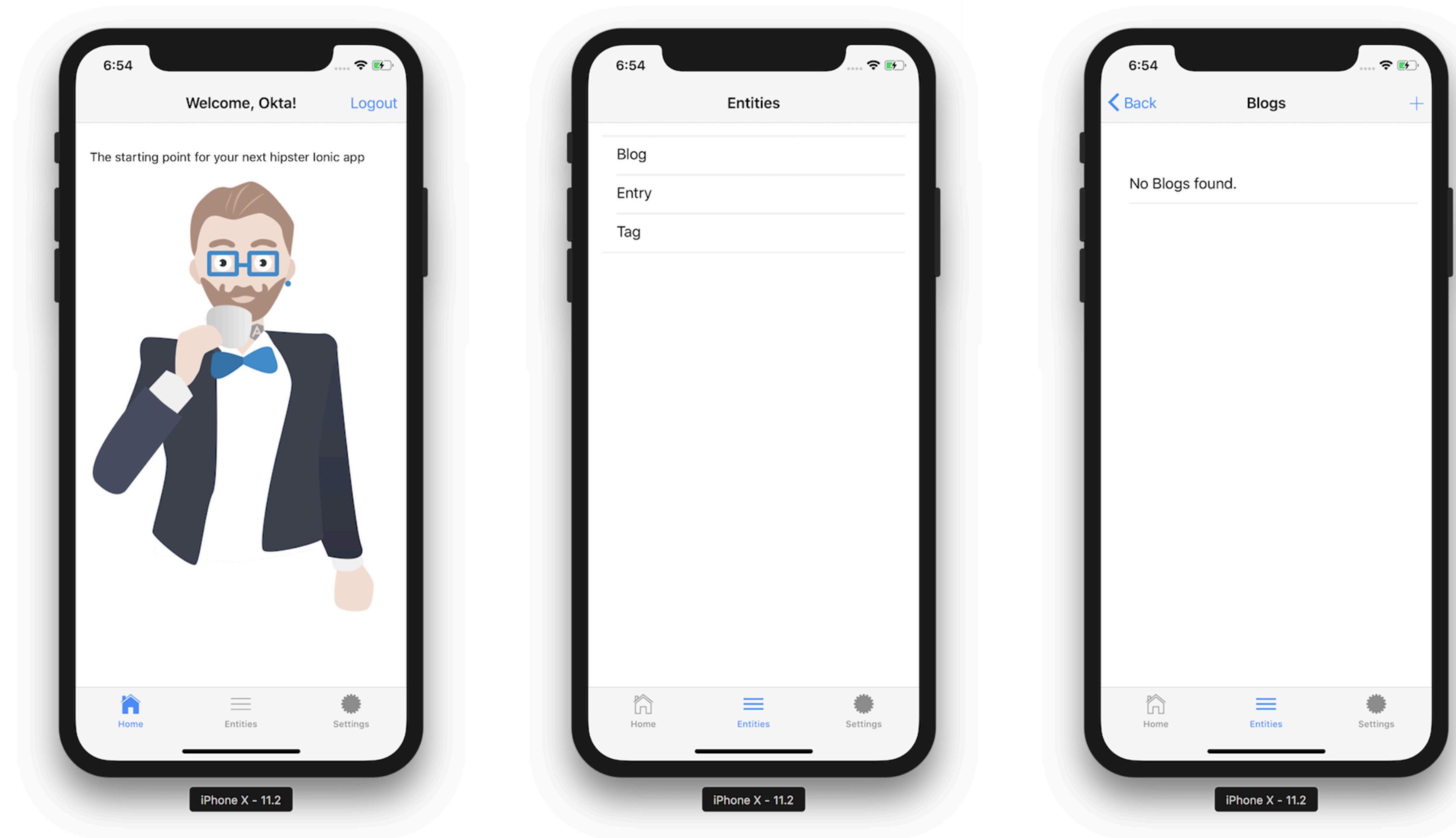


Because Ionic Apps need some JHipster ❤ too!

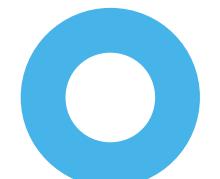
<https://github.com/oktadeveloper/ionic-jhipster-starter>



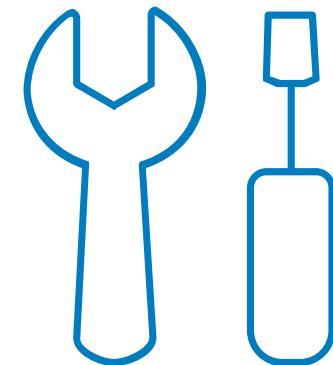
Use Ionic for JHipster to Create Mobile Apps



developer.okta.com/blog/2018/01/30/jhipster-ionic-with-oidc-authentication

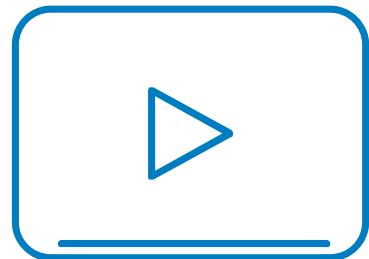


Ignite JHipster



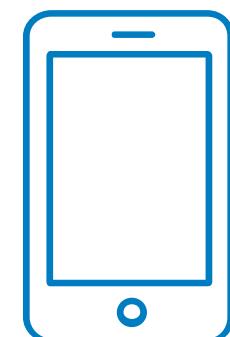
Install Ignite CLI and Ignite JHipster, using npm:

```
npm i -g ignite-cli ignite-jhipster
```



Create an app:

```
ignite new myapp -b ignite-jhipster
```



Ignite!

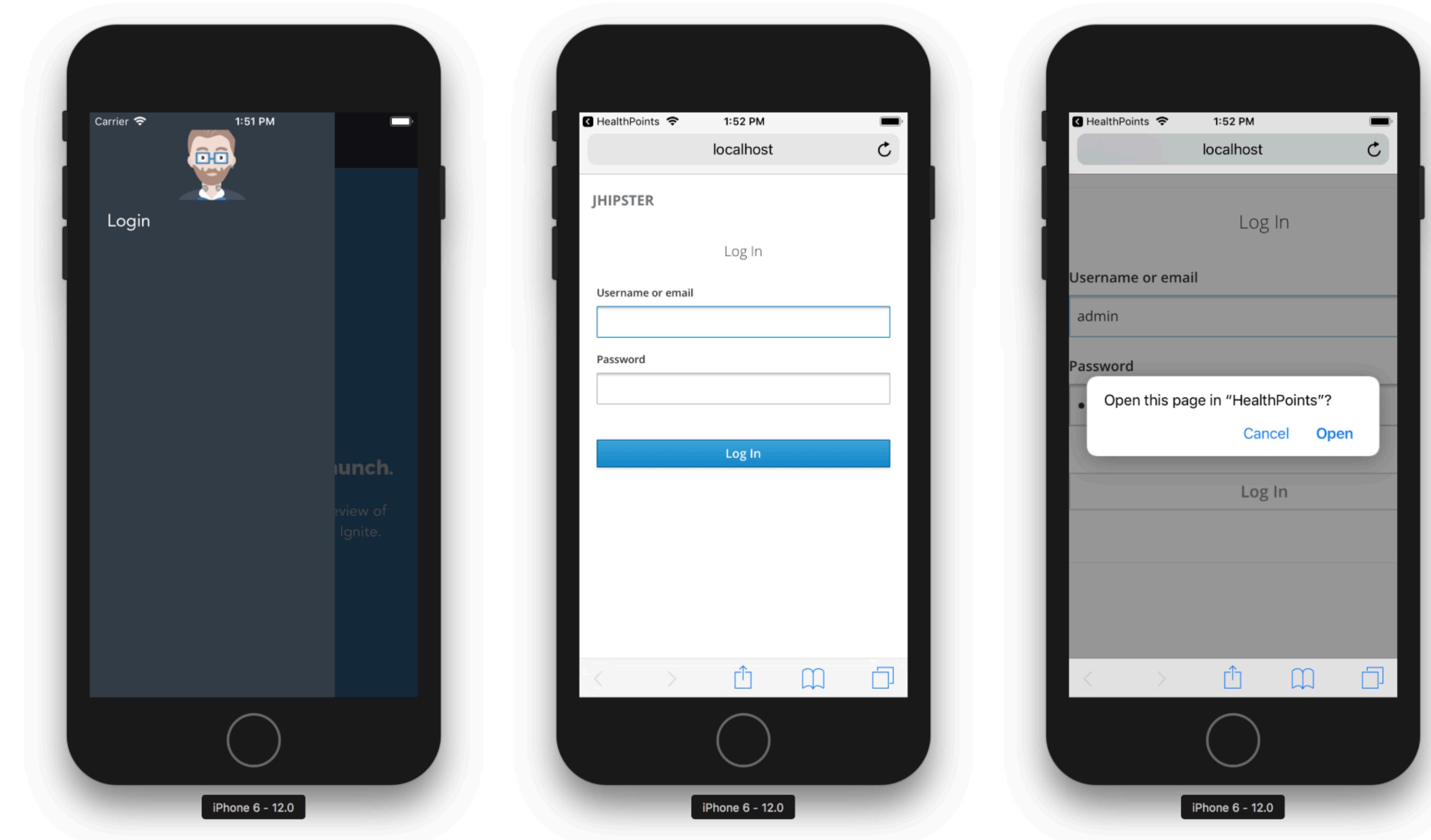


A React Native boilerplate for JHipster apps

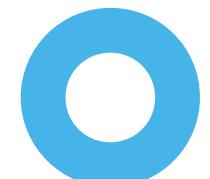
<https://github.com/ruddell/ignite-jhipster>



Build a Mobile App with React Native



developer.okta.com/blog/2018/10/10/react-native-spring-boot-mobile-app



Demo



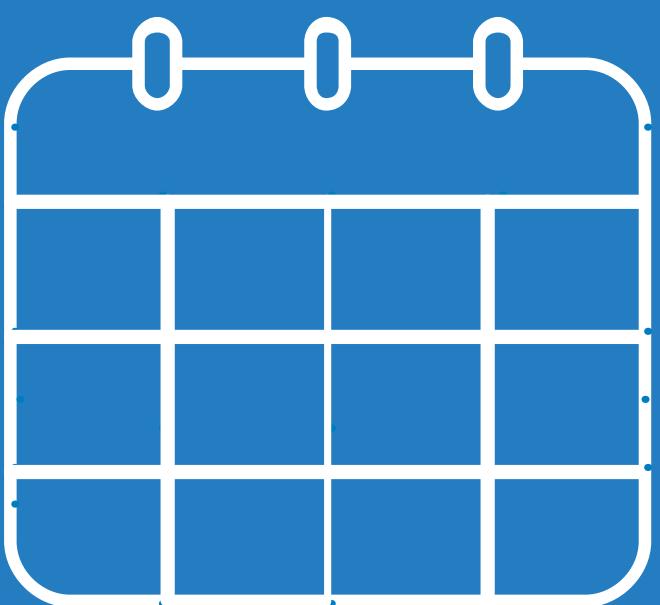
Create an Ionic app for JHipster
Generate entities that exist on gateway

Deploy to emulator

Deploy to phone

[https://github.com/oktadeveloper/
okta-ionic-jhipster-example](https://github.com/oktadeveloper/okta-ionic-jhipster-example)

Part 5

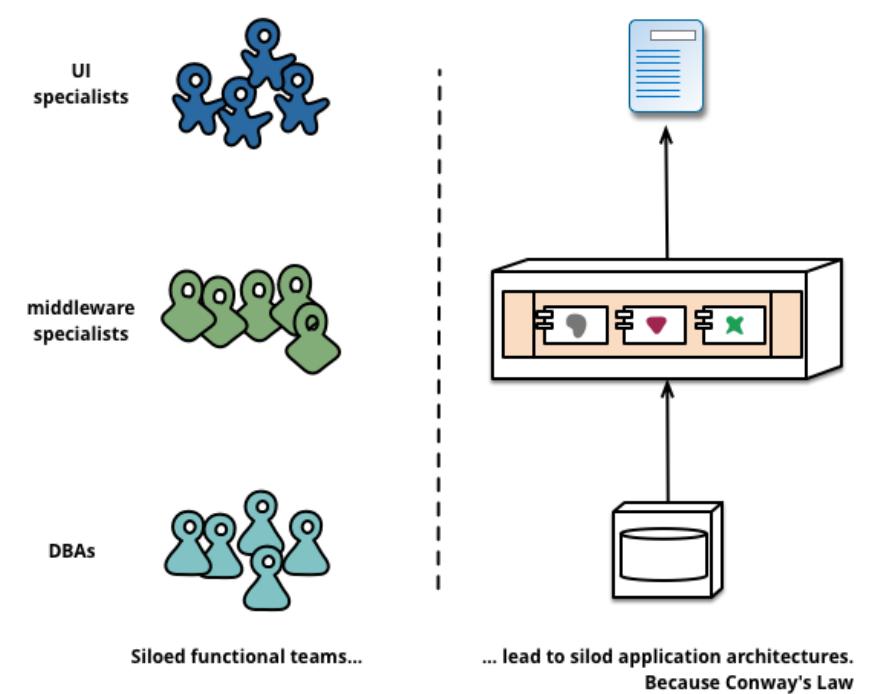


JHipster Roadmap

What You Learned

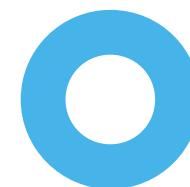
What's Next for JHipster

What You Learned



{okta}

ionic



JHipster Mobile Apps and Microservices on Pluralsight



pluralsight.com/courses/play-by-play-developing-microservices-mobile-apps-jhipster



What's Next for JHipster?



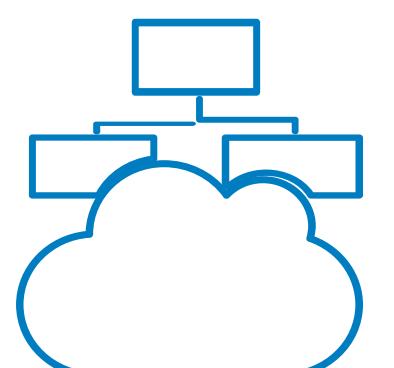
Blueprint Support



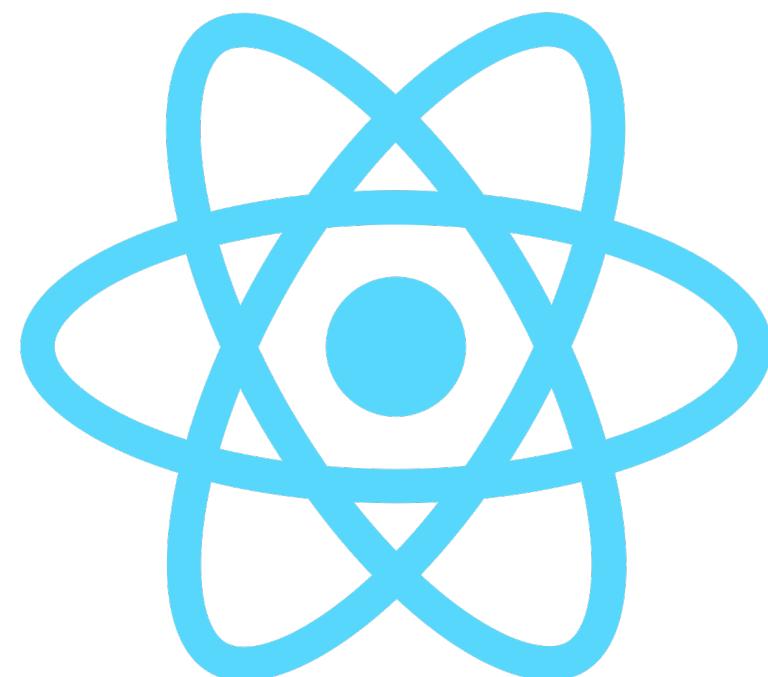
Spring WebFlux Support



start.jhipster.tech



React Support



 **JHipster v5.0**
June 2018



The JHipster Mini-Book



InfoQ
ENTERPRISE SOFTWARE
DEVELOPMENT SERIES

Written with Asciidoctor

Quick and to the point, 150 pages

v5.0 due out November 2018!

Developed a Real World App:

www.21-points.com

Free Download from InfoQ:

infoq.com/minibooks/jhipster-mini-book

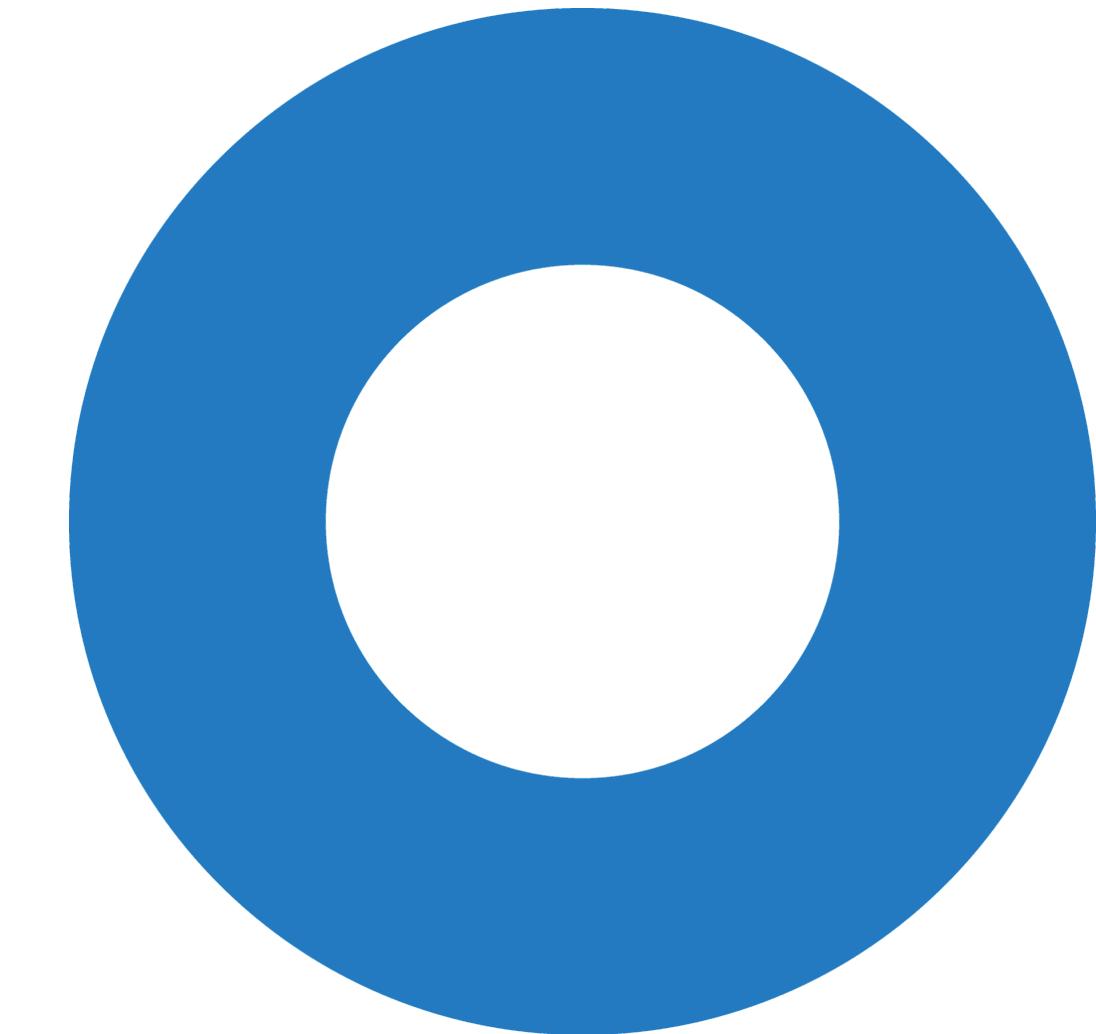
Learn More



Spring Boot
spring.io/guides



JHipster
www.jhipster.tech



Okta APIs
developer.okta.com



An aerial photograph of a city, likely Dubai, showing a dense urban area with numerous skyscrapers and a complex network of elevated highways and overpasses. The perspective is from a high vantage point, looking down at the city's infrastructure.

developer.okta.com/blog

@oktadev

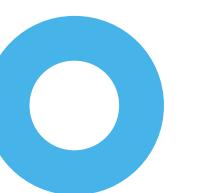
React PWA with JHipster 5

The screenshot shows a web browser window for a "gallery" application. The title bar indicates the page is at `localhost:9000/#/entity/photo`. The header includes the "JHipster" logo and navigation links for "Home", "Entities", "Administration", "EN", and "Account". A red banner on the left says "Gallery 0.0.1-SNAPSHOT". The main content area is titled "Photos" and features three large thumbnail images: one of four people on a ski lift, one of a family with a baby, and one of several colorful Volkswagen campervans. A blue button in the top right corner says "+ Create a new Photo". Below the thumbnails is a table listing three photo entries:

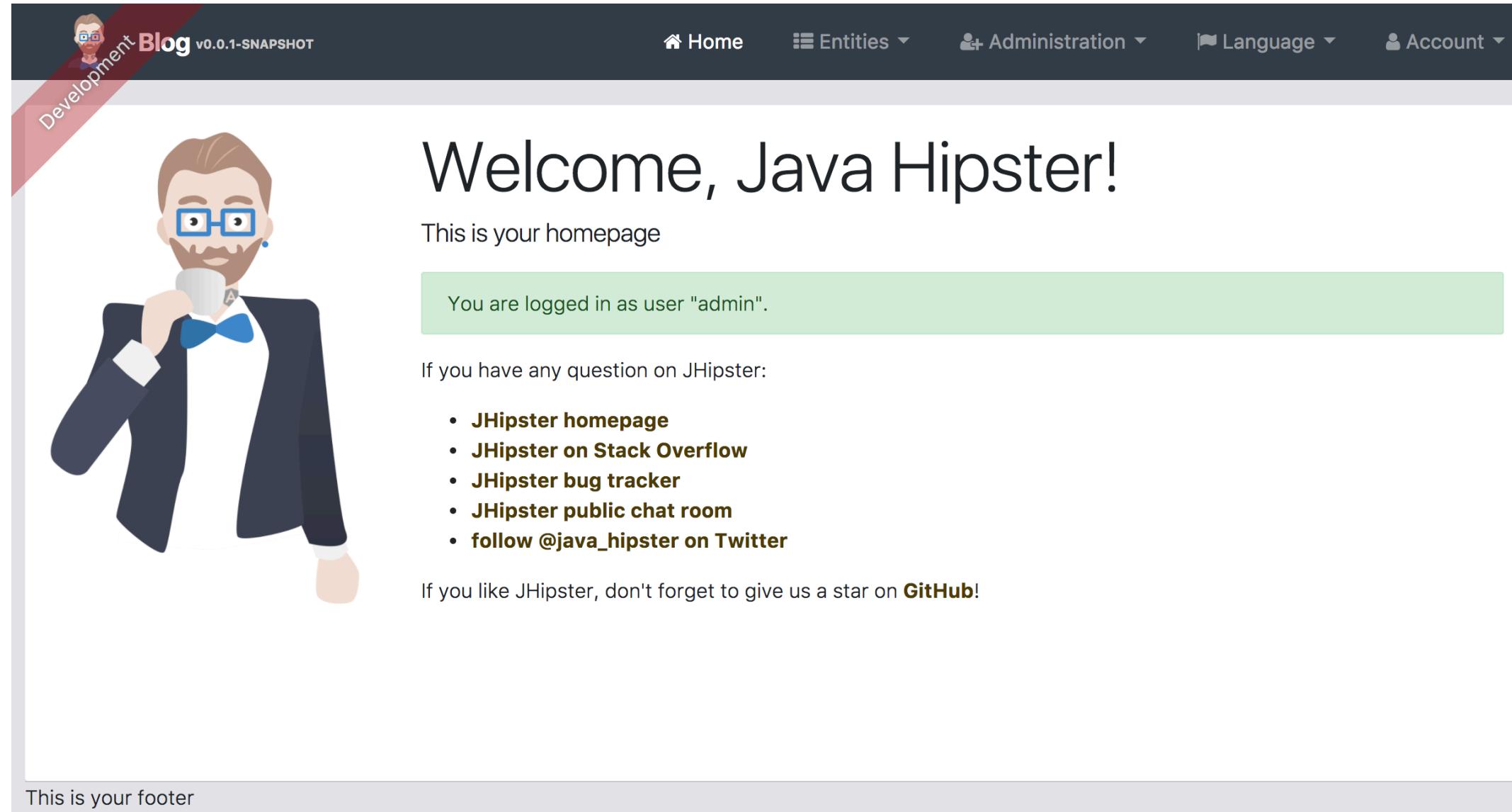
ID	Title	Description	Image	Height	Width	Taken	Uploaded	Album	
1001	Skiing			1539	2048	21/06/18 16:46	21/06/18 16:46		View Edit Delete
1002	James Ward			1536	2048	21/06/18 16:46	21/06/18 16:46		View Edit Delete
1003	VWs			1536	2048	21/06/18 16:46	21/06/18 16:46		View Edit Delete

A footer message "This is your footer" is visible at the bottom of the page.

developer.okta.com/blog/2018/06/25/react-spring-boot-photo-gallery-pwa



Action: Try JHipster!



The screenshot shows the JHipster homepage. At the top, there's a navigation bar with links for Home, Entities, Administration, Language, and Account. A red banner on the left says "Development Blog v0.0.1-SNAPSHOT". The main content area features a cartoon character of a man with glasses and a bow tie, holding a coffee cup. The text "Welcome, Java Hipster!" is prominently displayed. Below it, a message says "This is your homepage". A green box contains the text "You are logged in as user 'admin'.". Further down, there's a list of links for getting help with JHipster, followed by a note about starring the GitHub repository. The footer contains the text "This is your footer".



The screenshot shows the 21-Points Health homepage. At the top, there's a navigation bar with links for Home, Language, Account, and About. The title "21-Points Health v6.0.1" is at the top left, accompanied by a heart icon. The main content area features a large heart with an ECG line. The text "Welcome!" is displayed, followed by a message about tracking health and improving life. A blue button says "To get started, please ➔ Sign In". Below it, another blue button says "Learn more about 21-Points Health". There's also a link to the JHipster Mini-Book. The footer contains the text "Don't have an account yet? Register a new account".

JHipster 5 Demo

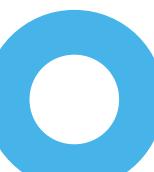
github.com/mraible/jhipster5-demo

youtu.be/-VQ_SVkaXbs

21-Points Health

github.com/mraible/21-points

infoq.com/minibooks/jhipster-mini-book



Use the Source, Luke!



github.com/oktadeveloper/okta-jhipster-microservices-oauth-example



{okta}

developer.okta.com

Thanks!

Keep in Touch

raibledesigns.com

@mraible

Presentations

speakerdeck.com/mraible

Code

github.com/oktadeveloper



developer.okta.com