

AWS for Java

Developers

#CampusParty2023

About the speaker

Weder Mariano de Sousa

Specialist Senior - GFT



AWS
community
builder

Graduated **Computer Science**

Post Graduate in **Midias UFG**

Post Graduate in **Information Security**

<https://www.linkedin.com/in/wedermarianodesousa/>

<https://github.com/weder96>

<https://twitter.com/weder96>

<https://dev.to/weder96>



GOJava



AWS User Group Goiânia

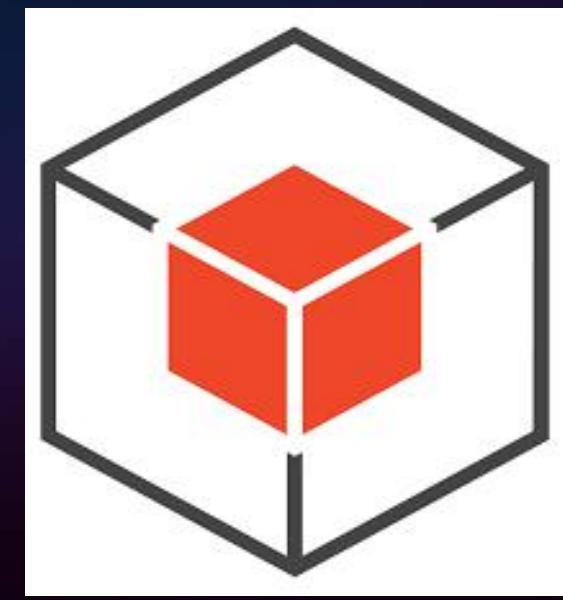


Agenda

1. **SDK for Java**
2. **AWS IDE Toolkits**
3. **AWS CDK for Java**
4. **Amazon Corretto**
5. **EC2 with Springboot**
6. **Beanstalk and Springboot**
7. **Docker and Fargate with Springboot**
8. **Pipelines with Java Springboot**
9. **Serverless - Java and Lambdas with Springboot**
10. **AWS Lambda SnapStart for Spring Developers**



1. SDK for Java



What is SDK for Java ?

The AWS SDK for Java simplifies use of AWS Services by providing a set of libraries that are consistent and familiar for Java developers.

It provides support for API lifecycle consideration such as credential management, retries, data marshaling, and serialization. The AWS SDK for Java also supports higher level abstractions for simplified development.

GitHub to see examples (<https://github.com/awsdocs/aws-doc-sdk-examples/tree/main>)
AWS-focused open source Java libraries.

- Aws Page Main Language Java

https://aws.amazon.com/developer/language/java/?nc1=h_ls

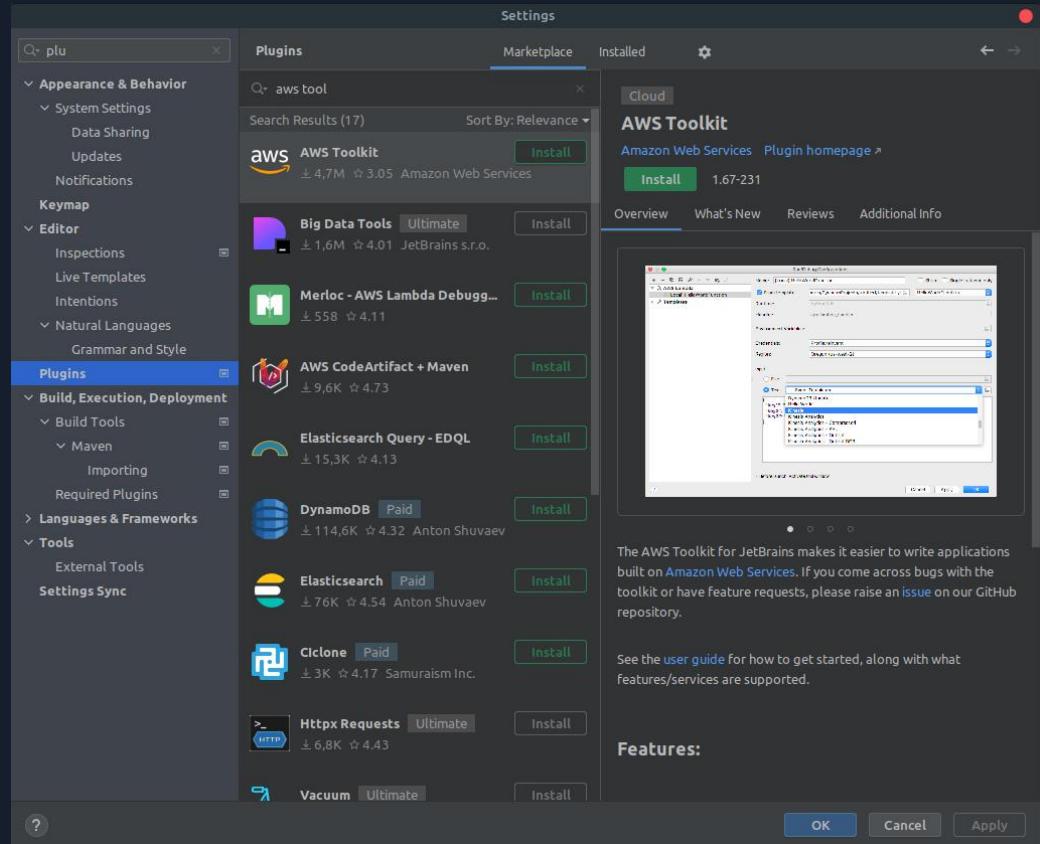


2. AWS IDE Toolkits





What is Toolkits de IDE da AWS ?



<https://aws.amazon.com/intellij/>



What is Toolkits de IDE da AWS ?

Amazon S3 > Buckets > campus-party-2023

campus-party-2023 Info

Objects Properties Permissions Metrics Management Access Points

Objects (3)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to...

C Copy S3 URI Copy URL Download Open Actions

Find objects by prefix

Name	Type	Last modified
AB177.jpg	jpg	May 31, 2023, 15:25:43 (UTC-03:00)
AB188.jpg	jpg	May 31, 2023, 15:25:44 (UTC-03:00)
AB199.jpg	jpg	May 31, 2023, 15:25:45 (UTC-03:00)

Project fleet ~/Documentos/Projetos/tracking/BackEnd/fleet

AWS Toolkit

default us-east-1

Explorer Developer Tools

Name

- AB177.jpg
- AB188.jpg
- AB199.jpg

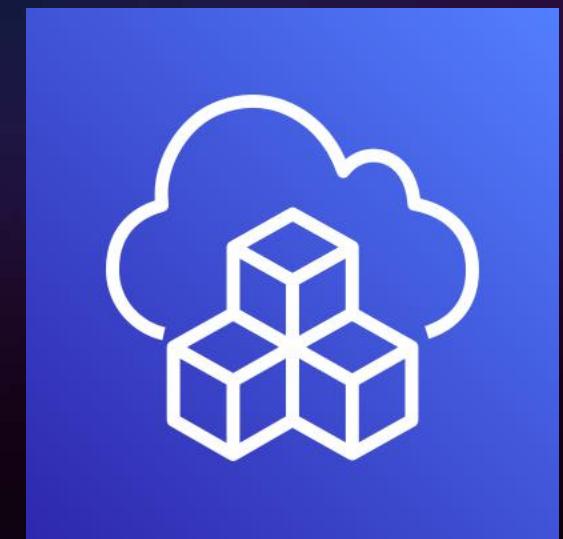
App Runner CloudFormation CloudWatch Logs DynamoDB ECR ECS Lambda S3

Schemas SQS Resources

campus-party-2023

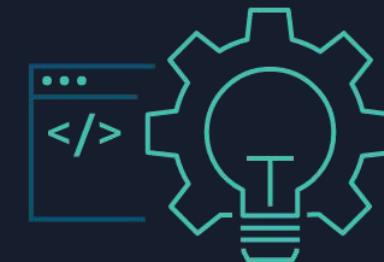
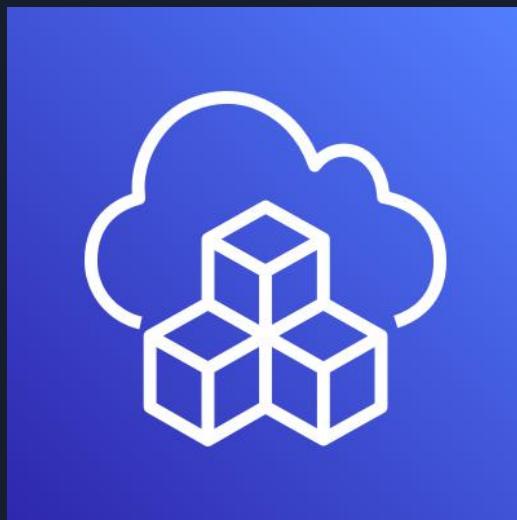
<https://aws.amazon.com/intellij/>

3. AWS CDK for Java



What is AWS CDK?

The AWS Cloud Development Kit (AWS CDK) is an open-source software development framework for defining cloud infrastructure as code with modern programming languages and deploying it through AWS CloudFormation.



<https://aws.amazon.com/cdk/faqs/>
https://aws.amazon.com/cdk/?nc1=h_ls

Difference between v2 and v1

CDK v1

```
"dependencies": {  
  "@aws-cdk/aws-amplify": "^1.125.0",  
  "@aws-cdk/aws-appsync": "^1.125.0",  
  "@aws-cdk/aws-codepipeline": "^1.125.0",  
  "@aws-cdk/aws-cognito": "^1.125.0",  
  "@aws-cdk/aws-lambda": "^1.125.0",  
  "@aws-cdk/aws-sns": "^1.125.0",  
  "@aws-cdk/core": "1.125.0",  
  "@aws-cdk/pipelines": "^1.125.0",  
}
```

```
import { App, Stack } from "@aws-cdk/core";  
const lambda = require("@aws-cdk/aws-lambda");
```

CDK v2

```
"dependencies": {  
  "aws-cdk-lib": "2.0.0",  
  "constructs": "^10.0.0",  
  "@aws-cdk/aws-iot-alpha": "^2.0.0-alpha.0"  
}
```

```
import { App, Stack } from "aws-cdk-lib";  
const s3 = require("aws-cdk-lib/aws-s3");
```

AWS Cloud Development Kit (CDK)

A multi-language software development framework for modeling cloud infrastructure as reusable components

```
package com.wsousa.awsimageupload.service;

import com.amazonaws.AmazonServiceException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.model.*;
import com.amazonaws.util.IOUtils;
import com.wsousa.awsimageupload.domain.BucketName;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.io.IOException;
import java.io.InputStream;
import java.util.List;
import java.util.Map;
import java.util.Optional;

3 usages  ± weder96
@Service
public class FileStore {

    9 usages
    private final AmazonS3 s3;

    ± weder96
    @Autowired
    public FileStore(AmazonS3 s3) { this.s3 = s3; }

}
```



Familiar



Tool Support



Abstraction

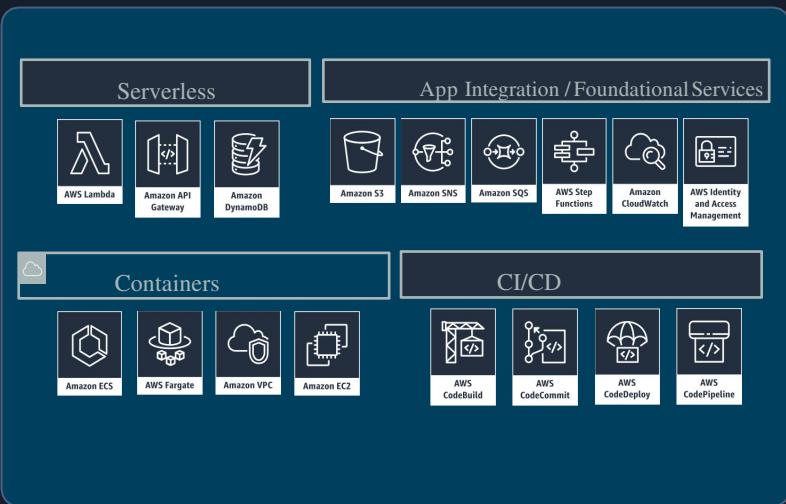


Main Components

App
Stacks
Resources



Core Framework



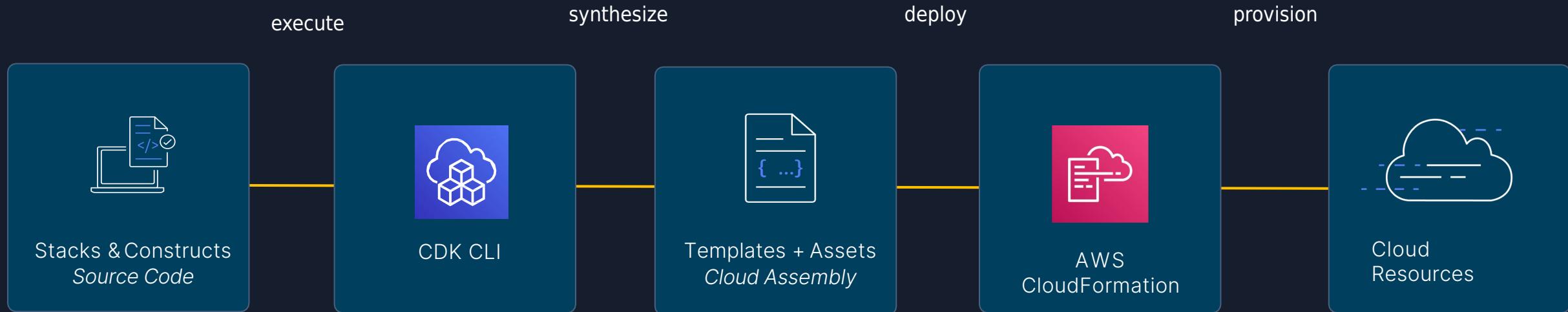
AWS Construct Library

A screenshot of a terminal window titled "zsh" showing the output of the AWS CDK CLI command. The output includes:

- IAM Statement Changes:** A table showing a new IAM policy statement being added to the stack. It includes columns for Resource, Effect, Action, Principal, and Condition.
- Conditions:** A detailed JSON object defining conditions for the IAM policy.
- Resources:** A list of AWS resources being created or modified, including S3 buckets, SQS queues, and Lambda functions.
- Logs:** The full command-line output of the CDK command, including the AWS CLI version and the stack name.

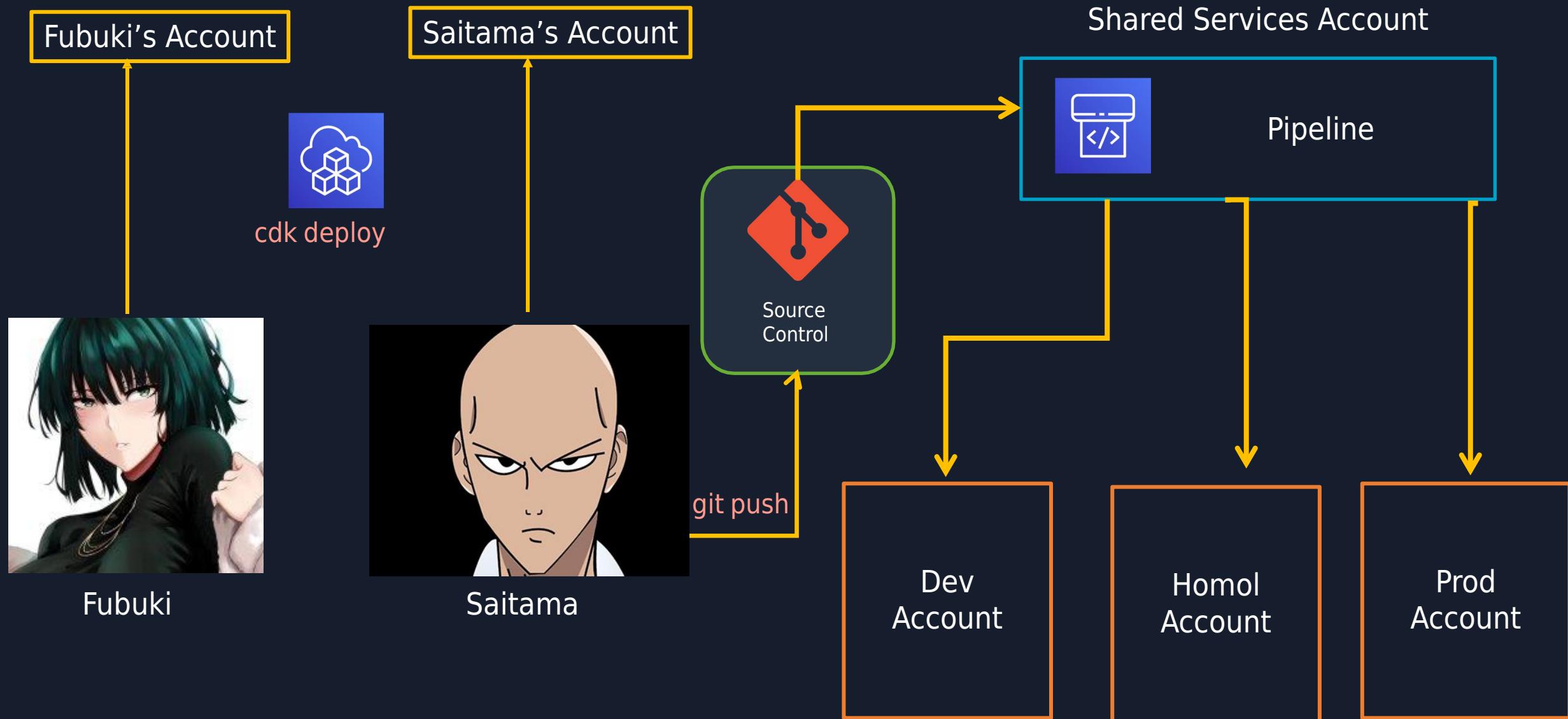
AWS CDK CLI

Development Workflow

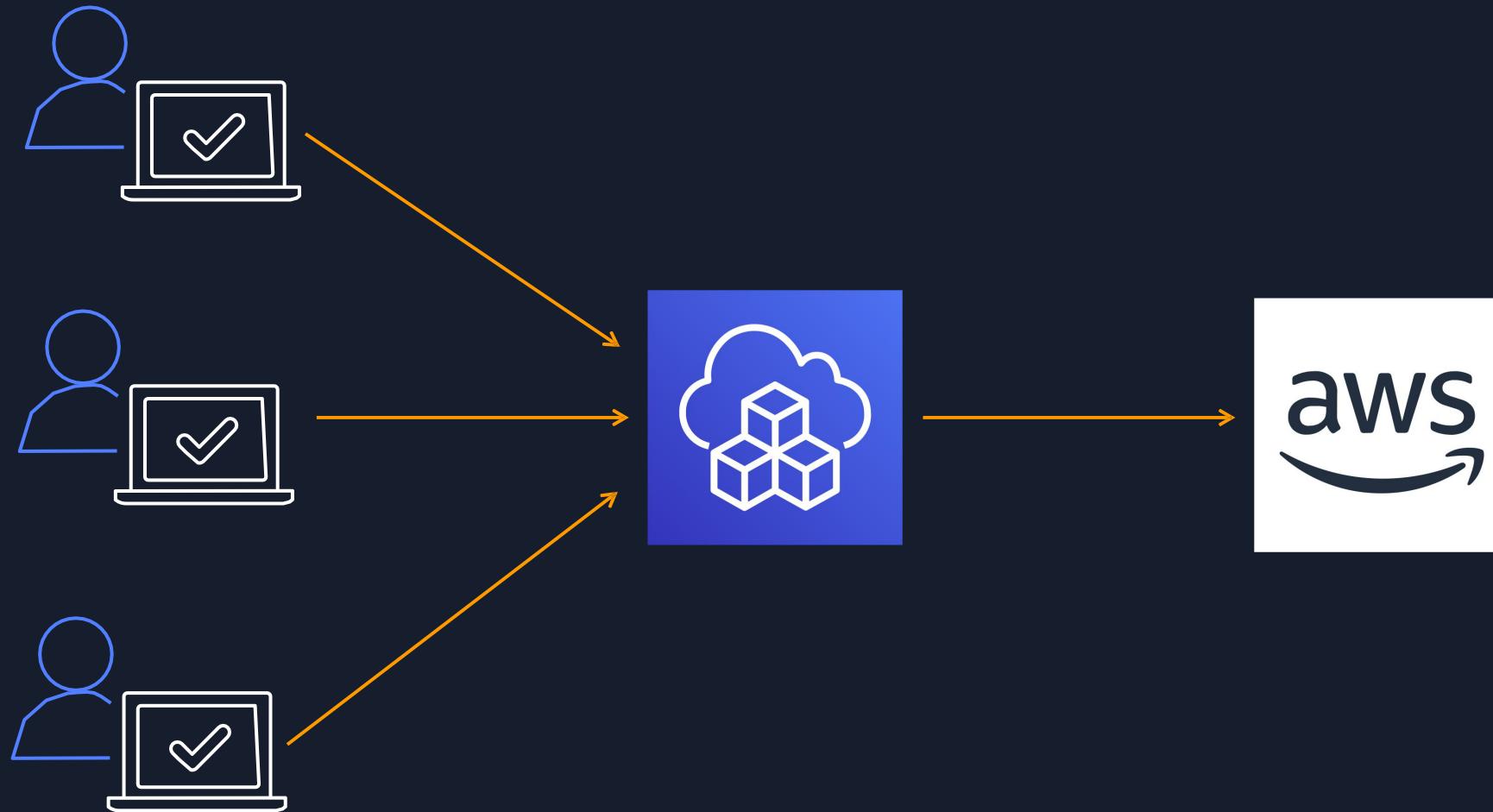


- | | | |
|--|----------------------|---------------------------------|
| | cdk init | // create new project |
| | npm run build | // build project |
| | cdk synth | // create templates and assets |
| | cdk diff | // check what will change |
| | cdk deploy | // push changes to your account |

Best Practice - Deploy to multiple accounts

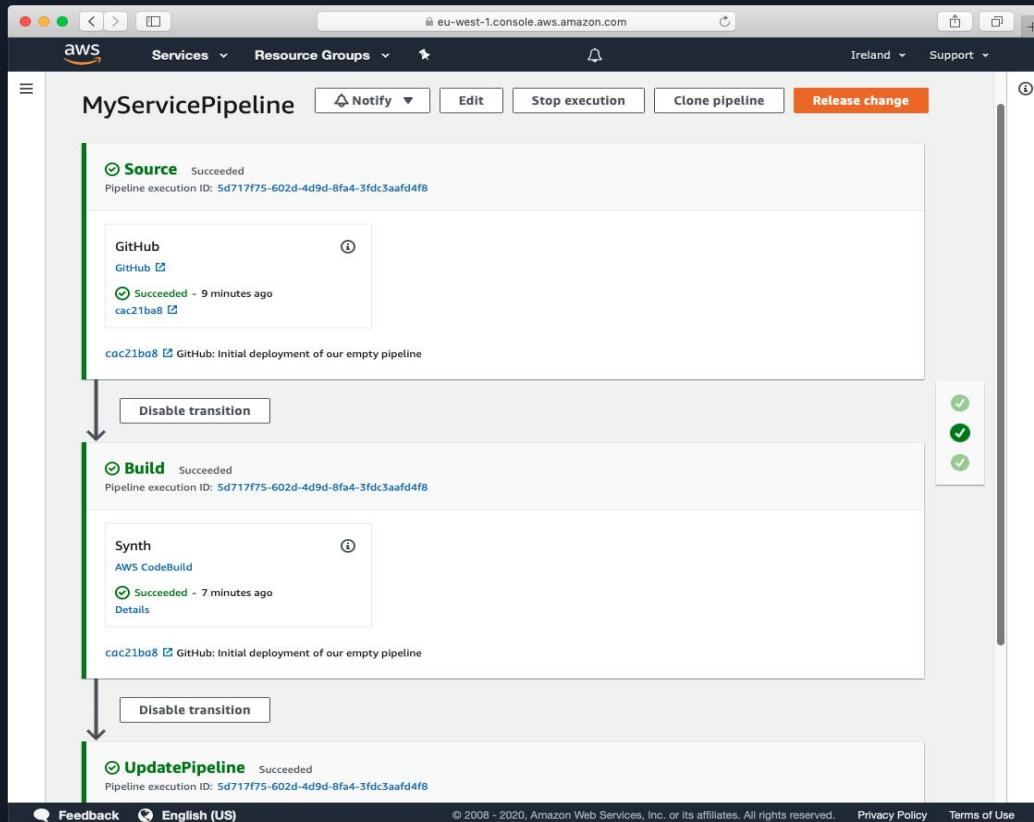


Application Development



CDK Pipelines

Continuous delivery for AWS CDK applications



Model continuous delivery pipelines as part of your infrastructure code.

Easily model cross-account and cross-region pipeline configurations.

Pipelines are self modifying as you push your CDK code to origin.

Demo CDK

Clone S3 AWS with CDK

aws-image-upload-wsousa / src / main / resources / application.properties



weder96

download e upload de arquivos

Code

Blame

7 lines (6 loc) · 168 Bytes

```
1 aws.access_key = ${value}
2 aws.secret_key = ${value}
3 aws.test = ${value}
4 spring.servlet.multipart.max-file-size=10MB
5
6 server.port=9090
7 springdoc.api-docs.path=/api-docs
```

Campus Party 2022

Home / Languages Typescript

Buckets (4) Info Buckets are containers for data stored in S3. Learn more

C ⚡ Copy ARN Empty Delete Create Bucket

Name	AWS Region	Access	Owner	Creation Date
data-lake-crime-results	US East (N. Virginia) us-east-1	Não informado	wmarians	27-04-2022 09:54:41
data-lake-crimes	US East (N. Virginia) us-east-1	Não informado	wmarians	27-04-2022 09:51:08
heroes-marvel	US East (N. Virginia) us-east-1	Não informado	wmarians	10-06-2022 15:36:53
heroes-marvel-iron-men	US East (N. Virginia) us-east-1	Não informado	wmarians	16-06-2022 15:17:48

« < > » Showing 1 to 4 of 4 buckets 10

Upload About: Seja Bem Vindo!!!

Success Info Warn Error

Message Plain Success Message Plain Info Message Plain Warn Message Plain Error

<https://github.com/weder96/aws-image-upload-wsousa>

<https://github.com/weder96/presentationCampusParty2022>

4. Amazon Corretto



What is Amazon Corretto ?

Amazon Corretto is a no-cost, multiplatform, production-ready distribution of the [Open Java Development Kit \(OpenJDK\)](#).

Corretto comes with Long-Term Support(LTS) that will include performance enhancements and security fixes. Amazon runs Corretto internally on thousands of production services and Corretto is certified as compatible with the Java SE standard. With Corretto, you can develop and run Java applications on popular operating systems, including Linux, Windows, and macOS.



Long-Term Support(LTS) ?

Benefits

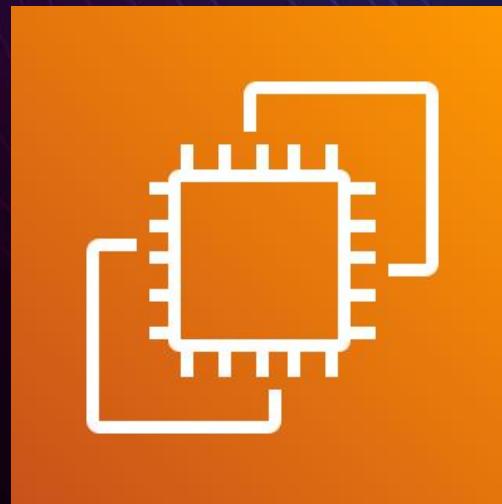


- 1. Backed by Amazon*
- 2. Production Ready*
- 3. Multiplatform Support*
- 4. No Cost*

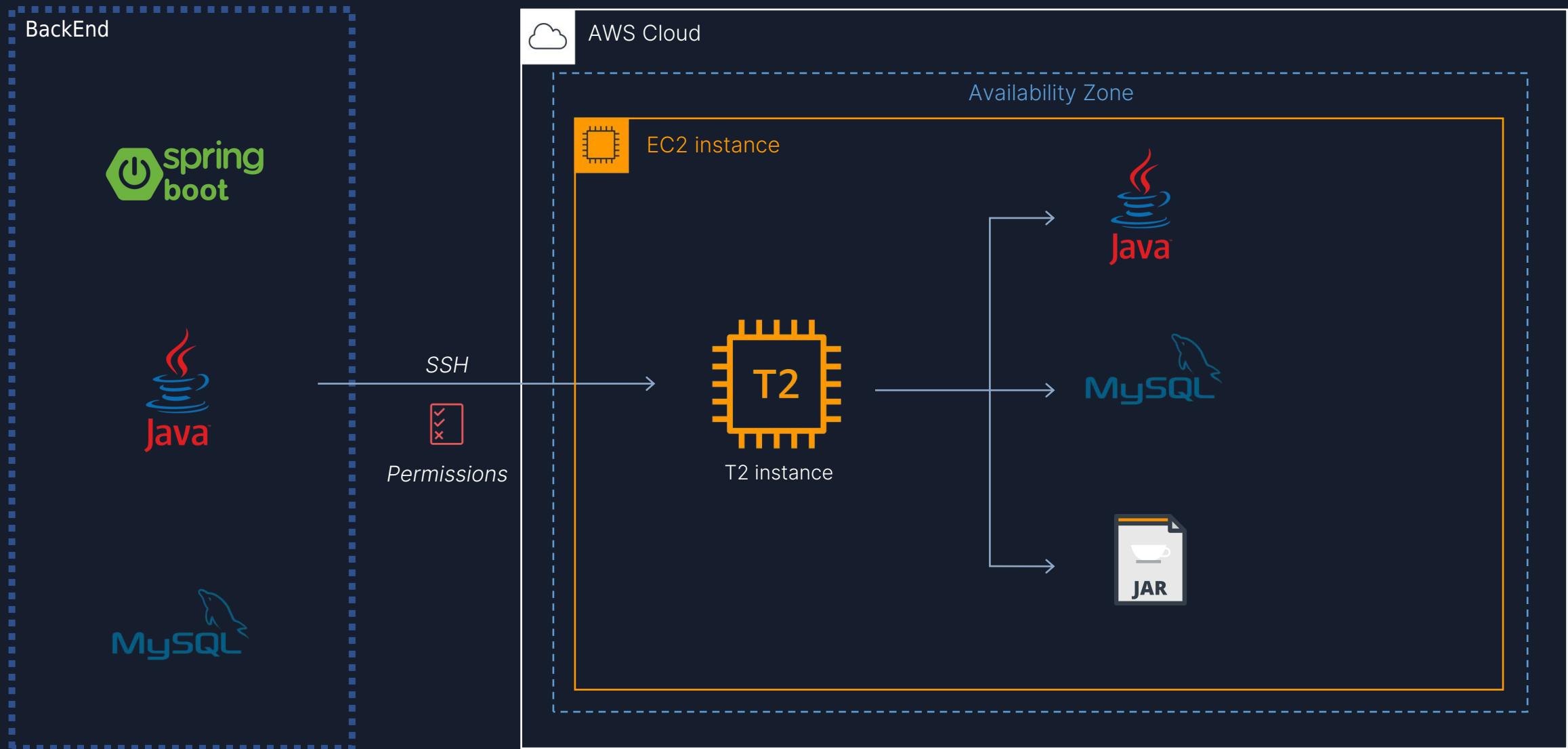


<https://aws.amazon.com/corretto/?filtered-posts.sort-by=item.additionalFields.createdDate&filtered-posts.sort-order=desc>

5. EC2 with Spring boot



Architecture (EC2)



Ec2 Baisc Server Instance

Instances (1) [Info](#)

Find instance by attribute or tag (case-sensitive)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type
<input type="checkbox"/>	ec2-springboot-server	i-0b8ccf9a3563af734	Running	t2.micro

[EC2 Instance Connect](#) [Session Manager](#) [SSH client](#) [EC2 serial console](#)

Instance ID

[i-0b8ccf9a3563af734](#) (ec2-springboot-server)

1. Open an SSH client.

2. Locate your private key file. The key used to launch this instance is `ec2-key-par-ssh.pem`

3. Run this command, if necessary, to ensure your key is not publicly viewable.

`chmod 400 ec2-key-par-ssh.pem`

4. Connect to your instance using its Public DNS:

`ec2-54-197-199-11.compute-1.amazonaws.com`

Example:

`ssh -i "ec2-key-par-ssh.pem" ubuntu@ec2-54-197-199-11.compute-1.amazonaws.com`

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

```
[weder@weder ~]$ cd Downloads/
[weder@weder Downloads]$ chmod 400 ec2-key-pair-ssh.pem
[weder@weder Downloads]$ ssh -i "ec2-key-pair-ssh.pem" ec2-user@ec2-34-203-216-237.compute-1.amazonaws.com
The authenticity of host 'ec2-34-203-216-237.compute-1.amazonaws.com (34.203.216.237)' can't be established.
ECDSA key fingerprint is SHA256:IxLq2/sAKnSCgSW8INrAym6srlb2dEv89uZCddXTUns.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'ec2-34-203-216-237.compute-1.amazonaws.com,34.203.216.237' (ECDSA) to the list of known host
s.

```

```
[weder@weder campus (main)]$ scp -i ~/Downloads/ec2-key-pair-ssh.pem target/campus-0.0.1-SNAPSHOT.jar ubuntu@ec2-54-197-199-11.compute-1.amazonaws.com:~  
campus-0.0.1-SNAPSHOT.jar                                         100%   48MB   2.9MB/s   00:16  
[weder@weder campus (main)]$
```

```
>_ ubuntu@ip-172-31-88-3:~ +  
ubuntu@ip-172-31-88-3:~$ pwd  
/home/ubuntu campus (main)S scp -r ~/Downloads/ec2-k  
ubuntu@ip-172-31-88-3:~$ ls campus-0.0.1-SNAPSHOT.jar  
ubuntu@ip-172-31-88-3:~$
```

Ec2 Basic Server Instance - Deploy Application Java and Mysql

```
ubuntu@ip-172-31-88-3:~$ ls
campus-0.0.1-SNAPSHOT.jar
ubuntu@ip-172-31-88-3:~$ java -version
openjdk version "17.0.7" 2023-04-18
OpenJDK Runtime Environment (build 17.0.7+7-Ubuntu-0ubuntu122.04.2)
OpenJDK 64-Bit Server VM (build 17.0.7+7-Ubuntu-0ubuntu122.04.2, mixed mode, sharing)
ubuntu@ip-172-31-88-3:~$
```

POST ▼ ec2-54-197-199-11.compute-1.amazonaws.com:8080/api/agenda/v1/save

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON ▼

```
1 [{}]
2   "community": "Join Community",
3   "description": " teste ",
4   "schedule": "10:30 as 11:30",
5   "speaker": "Weder Sousa",
6   "title": "Aws for Java Developers"
7 ]
```

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize **JSON** ▼ Copy

```
1 []
2   "recordCount": 1,
3   "data": [
4     {
5       "id": 1,
6       "title": "Aws for Java Developers",
7       "speaker": "Weder Sousa",
8       "description": " teste ",
9       "community": "Join Community",
10      "schedule": "10:30 as 11:30"
11    }
12  ]
```

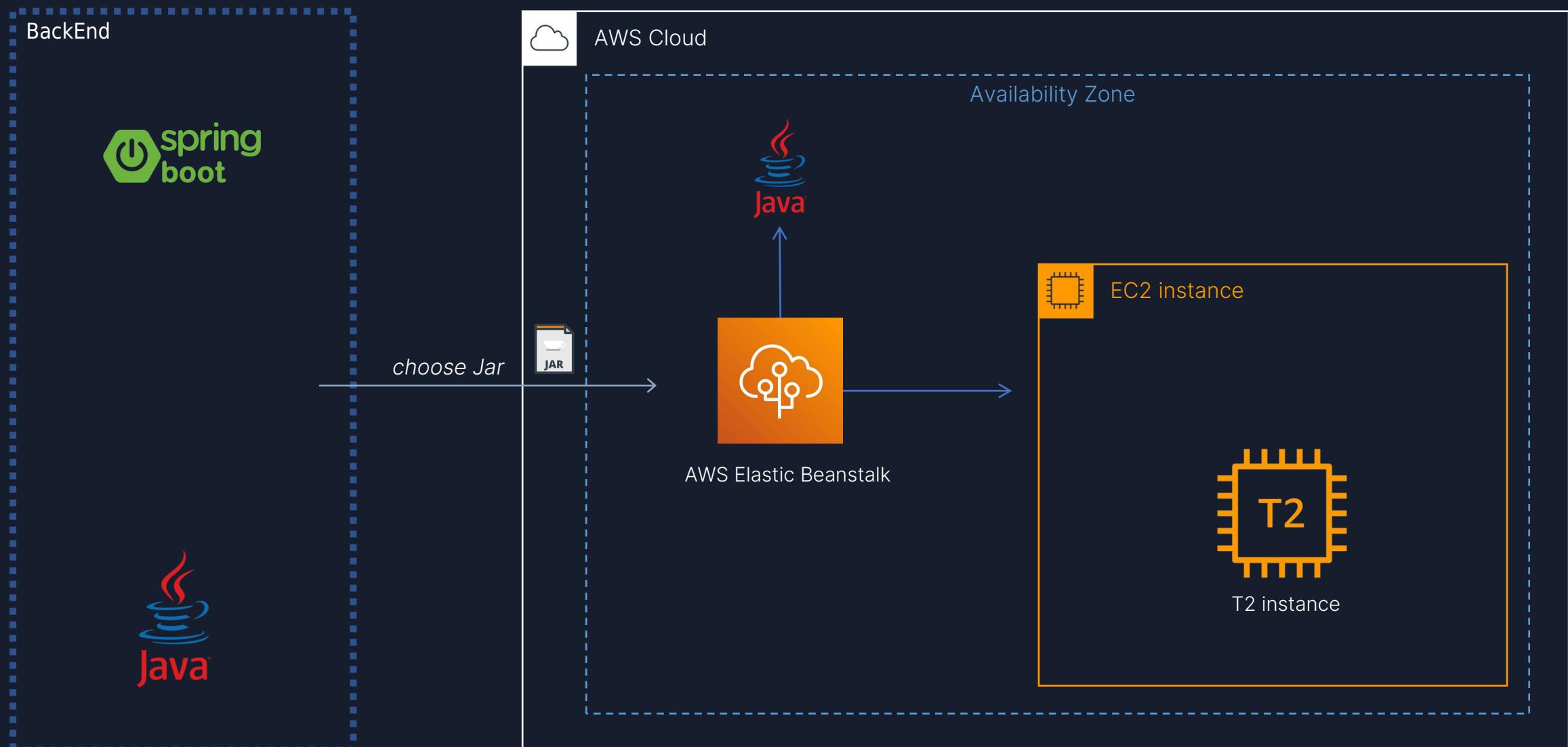
```
GET ec2-54-197-199-11.compute-1.amazonaws.com:8080/api/agenda/v1

Params Authorization Headers (6) Body Pre-request Script Tests Settings
Query Params
KEY VALUE
Body Cookies Headers (8) Test Results
Pretty Raw Preview Visualize JSON ↻
1 [
2   "recordCount": 1,
3   "data": [
4     {
5       "id": 1,
6       "title": "Aws for Java Developers",
7       "speaker": "Weder Sousa",
8       "description": " teste ",
9       "community": "Join Community",
10      "schedule": "10:30 as 11:30"
11    }
12  ]
13 ]
```

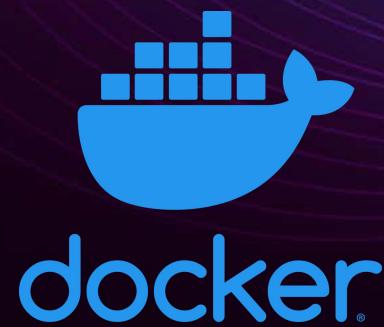
6. Beanstalk and Spring boot



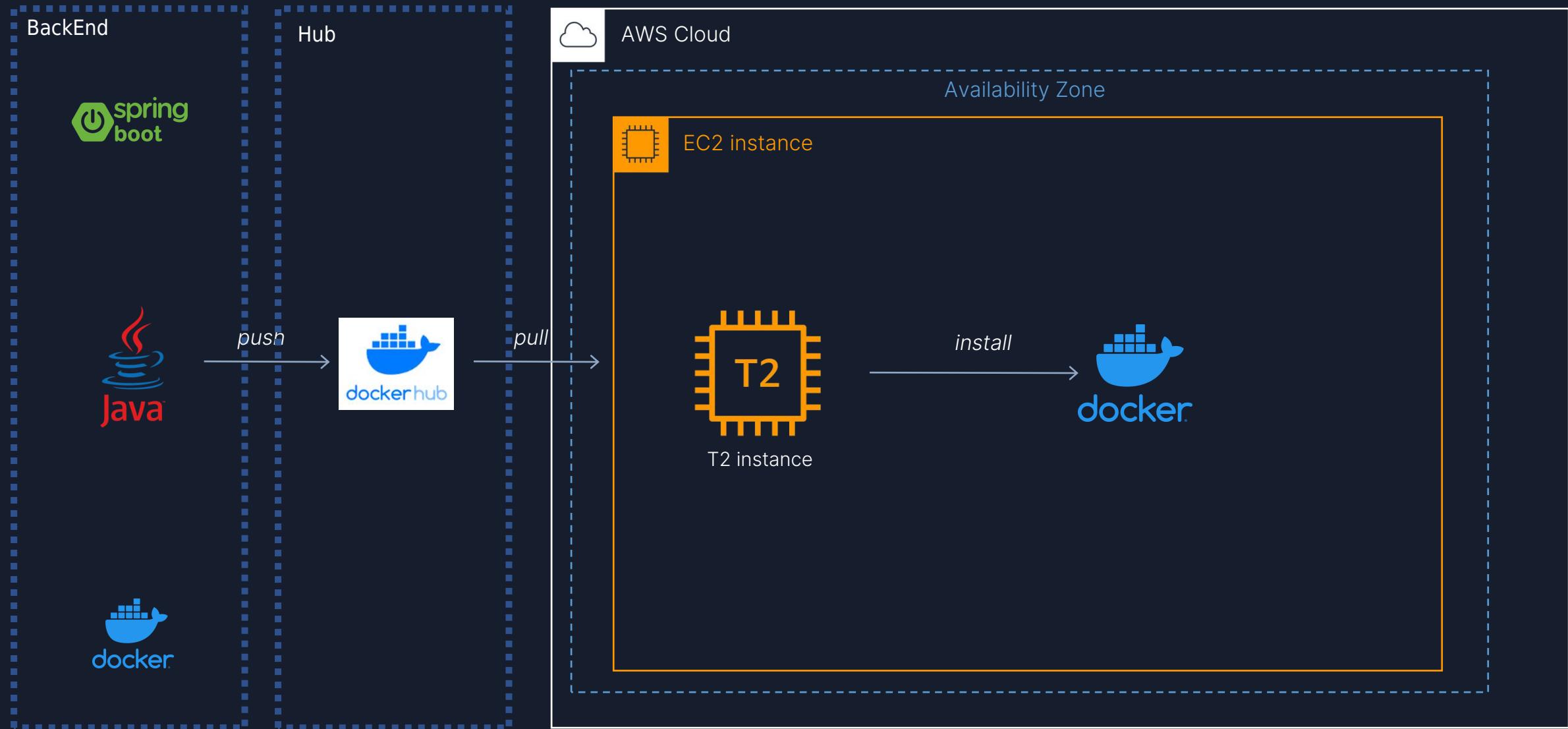
Architecture (Beanstalk)



7. Docker and Fargate with Spring boot



Architecture (Docker Hub)



Create Account Docker Hub(Repository) and Login

The screenshot shows the Docker Hub interface for a repository named 'campus-docker'. The top navigation bar includes 'Explore', 'Repositories', 'Organizations', and 'Help'. Below the navigation, the repository path 'weder96 > Repositories > campus-docker > General' is shown. The 'General' tab is selected, followed by 'Tags', 'Builds', 'Collaborators', 'Webhooks', and 'Settings'. The main content area displays the repository name 'weder96 / campus-docker', a 'Description' section containing 'test image springboot and docker Hub campus-docker', and a timestamp 'Last pushed: a few seconds ago'. A 'Tags' section indicates there are 2 tags: '0.0.2-SNAPSHOT' and '0.0.1-SNAPSHOT', both of which are listed as 'Image' type tags. There are also links to 'See all' and 'Go to Advanced Image Management'.

Tag	OS	Type	Pulled	Pushed
0.0.2-SNAPSHOT	🐧	Image	---	a minute ago
0.0.1-SNAPSHOT	🐧	Image	---	9 minutes ago

```
[weder@weder ~ ]$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: weder96
Password:
WARNING! Your password will be stored unencrypted in /home/weder/.docker/config.json.
Configure a credential helper to remove this warning. See 'docker login --help'.
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
Type          Pulled          Pushed          Login Succeeded
[...]
Available with Pro, Team and Business subscriptions. Read more about automated builds.
```

Push Project Spring boot to Docker Hub

```
FROM openjdk:17-oracle
ADD target/campus-docker.jar campus-docker.jar
EXPOSE 8080
ENTRYPOINT ["java", "-jar", "campus-docker.jar"]
```

```
[weder@weder campus_docker (main)]$ mvn dockerfile:push
```

```
[INFO] The push refers to repository [docker.io/weder96/campus-docker]
[INFO] Image 1b6995919e95: Preparing
[INFO] Image dc9fa3d8b576: Preparing
[INFO] Image 27ee19dc88f2: Preparing
[INFO] Image c8dd97366670: Preparing
[INFO] Image 1b6995919e95: Pushing
[INFO] Image 27ee19dc88f2: Mounted from library/openjdk
[INFO] Image dc9fa3d8b576: Mounted from library/openjdk
[INFO] Image c8dd97366670: Mounted from library/openjdk
[INFO] Image 1b6995919e95: Pushed
[INFO] 0.0.2-SNAPSHOT: digest: sha256:d2b2ff145ef5eb300a644baa2708ee19d72239d9fb73d4b2a52d61b1adf8c2d0 size: 1166
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 31.795 s
[INFO] Finished at: 2023-06-05T20:44:56-03:00
[INFO] -----
```

```
<plugin>
    <groupId>com.spotify</groupId>
    <artifactId>dockerfile-maven-plugin</artifactId>
    <version>1.4.13</version>
    <executions>
        <execution>
            <id>default</id>
            <goals>
                <goal>build</goal>
                <goal>push</goal>
            </goals>
        </execution>
    </executions>
    <configuration>
        <useMavenSettingsForAuth>true</useMavenSettingsForAuth>
        <repository>weder96/campus-docker</repository>
        <tag>${project.version}</tag>
        <buildArgs>
            <JAR_FILE>${project.build.finalName}.jar</JAR_FILE>
        </buildArgs>
    </configuration>
</plugin>
</plugins>
<finalName>campus-docker</finalName>
```

https://github.com/weder96/campusParty3/tree/main/backEnd/campus_docker

Pull Project Docker Hub

```
[weder@weder ~]$ docker pull weder96/campus-docker:0.0.2-SNAPSHOT
0.0.2-SNAPSHOT: Pulling from weder96/campus-docker
Digest: sha256:d2b2ff145ef5eb300a644baa2708ee19d72239d9fb73d4b2a52d61b1adf8c2d0
Status: Image is up to date for weder96/campus-docker:0.0.2-SNAPSHOT
docker.io/weder96/campus-docker:0.0.2-SNAPSHOT
```

```
[weder@weder ~]$ docker run -p 8080:8080 weder96/campus-docker:0.0.2-SNAPSHOT
```

Test Project Spring boot (localhost)

```
> + [weder@weder ~ ]$ curl http://192.168.1.110:8080/api/agenda/v1/health {"recordCount":1,"data":"Loading..."}[weder@weder ~ ]$
```

The screenshot shows the Postman application interface. At the top, there is a header bar with a 'GET' method dropdown, a URL input field containing 'http://192.168.1.110:8080/api/agenda/v1/health', and several other tabs like 'Authorization', 'Headers (6)', 'Body', 'Pre-request Script', 'Tests', and 'Settings'. Below the header, there is a 'Params' section and a 'Query Params' table with one row. Underneath these sections, there are tabs for 'Body', 'Cookies', 'Headers (8)', and 'Test Results', with 'Body' being the active tab. At the bottom of the body section, there are buttons for 'Pretty', 'Raw', 'Preview', 'Visualize', and a dropdown menu set to 'JSON'. The main content area displays the JSON response from the curl command above. The JSON structure is as follows:

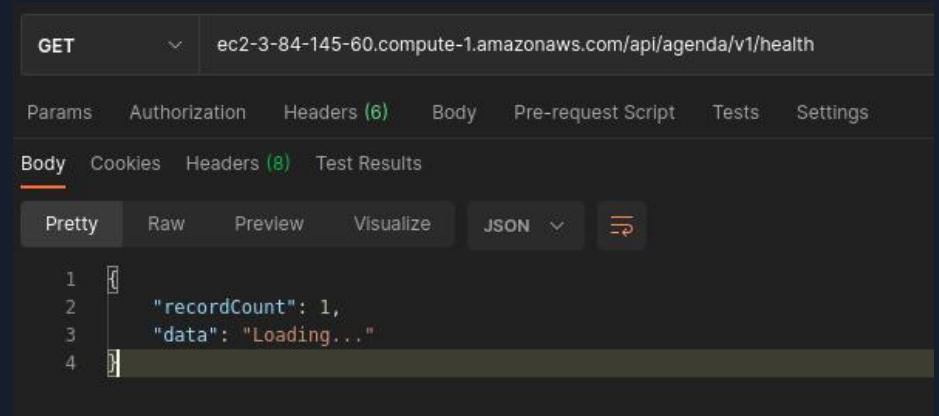
```
1 [ {  
2   "recordCount": 1,  
3   "data": "Loading..."  
4 } ]
```

Test Project Spring boot (EC2 Instance)

```
ubuntu@ip-172-31-82-253:~$ docker pull weder96/campus-docker:0.0.2-SNAPSHOT
0.0.2-SNAPSHOT: Pulling from weder96/campus-docker
38a980f2cc8a: Pull complete
de849f1cfbe6: Pull complete
a7203ca35e75: Pull complete
ee2a9eb88acf: Pull complete
Digest: sha256:d2b2ff145ef5eb300a644baa2708ee19d72239d9fb73d4b2a52d61b1adf8c2d0
Status: Downloaded newer image for weder96/campus-docker:0.0.2-SNAPSHOT
docker.io/weder96/campus-docker:0.0.2-SNAPSHOT
```

```
ubuntu@ip-172-31-82-253:~$ docker run -p 80:8080 weder96/campus-docker:0.0.2-SNAPSHOT
:: Spring Boot ::   (v3.0.6)

2023-06-06T14:10:58.699Z  INFO 1 --- [           main] com.wsousa.campus.CampusApplication      : Starting CampusApplication v0.0.2-SNAPSHOT using
Java 17.0.2 with PID 1 (/campus-docker.jar started by root in /)
2023-06-06T14:10:58.711Z  INFO 1 --- [           main] com.wsousa.campus.CampusApplication      : No active profile set, falling back to 1 default
profile: "default"
2023-06-06T14:11:01.363Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2023-06-06T14:11:01.393Z  INFO 1 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2023-06-06T14:11:01.393Z  INFO 1 --- [           main] o.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/10.1.8]
2023-06-06T14:11:01.703Z  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationConte
xt
2023-06-06T14:11:01.709Z  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization compl
eted in 2866 ms
2023-06-06T14:11:03.442Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with cont
ext path ''
2023-06-06T14:11:03.468Z  INFO 1 --- [           main] com.wsousa.campus.CampusApplication      : Started CampusApplication in 5.862 seconds (proc
ess running for 7222)
```



```
GET ec2-3-84-145-60.compute-1.amazonaws.com/api/agenda/v1/health
```

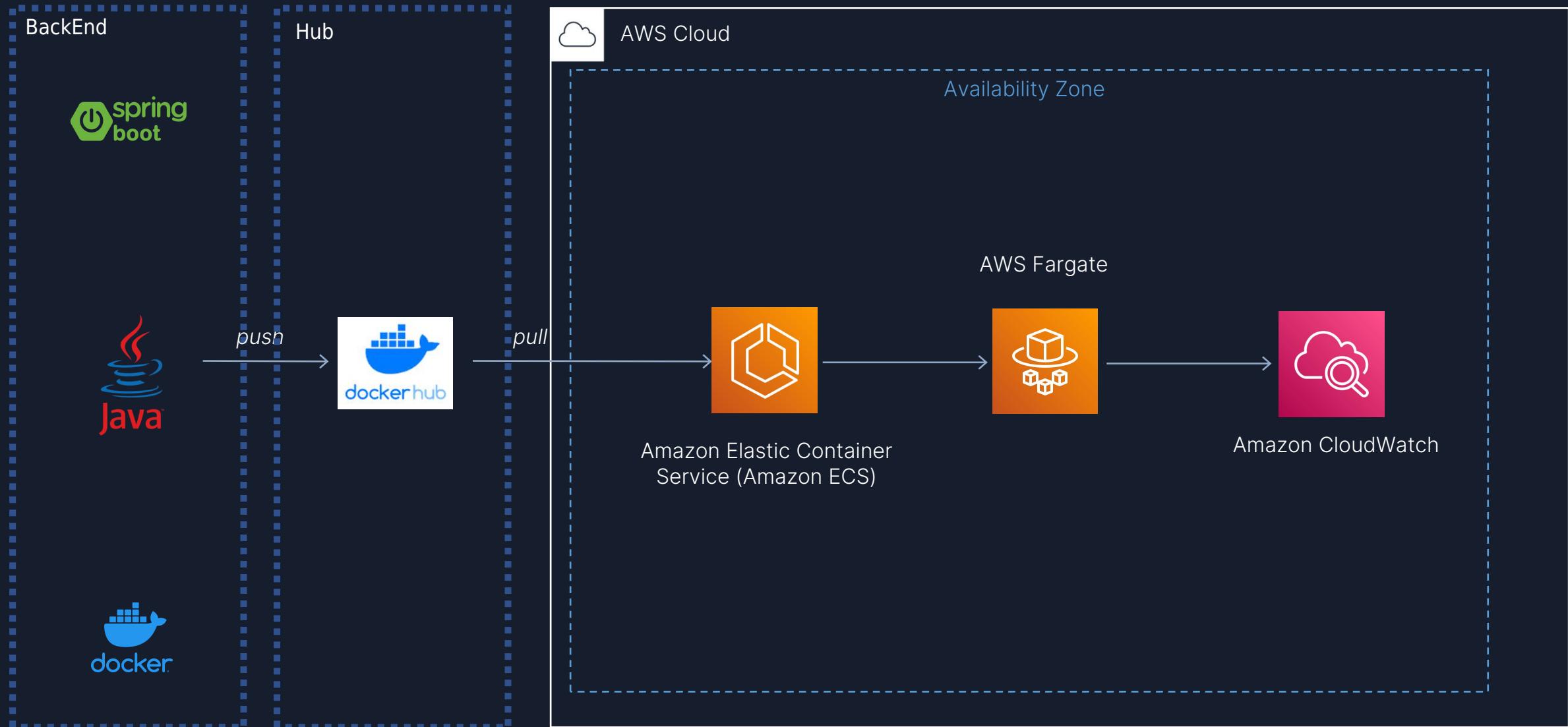
Params Authorization Headers (6) Body Pre-request Script Tests Settings

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

```
1 []
2   "recordCount": 1,
3   "data": "Loading..."
4 ]
```

Architecture (Docker Hub and Fargate)



ECS and Fargate

Create Cluster

The screenshot shows the AWS Elastic Container Service (ECS) Clusters page. At the top, there is a breadcrumb navigation: Amazon Elastic Container Service > Clusters. Below the header, there is a search bar labeled "Search clusters". On the right side of the header, there are three buttons: a "Create cluster" button (highlighted in orange), a "Copy" button, and a "Delete" button. Below the header, there is a table with the following columns: Cluster, Services, Tasks, Registered container instances, CloudWatch monitoring, and Capacity provider strategy. The table contains one row for the cluster "cls-demo-springboot". The "Tasks" column shows "0 Pending | 1 Running". The "Registered container instances" column shows "0". The "CloudWatch monitoring" column has a checked checkbox labeled "Default". The "Capacity provider strategy" column shows "No default found".

Create Task Definitions

The screenshot shows the AWS Elastic Container Service (ECS) Task definitions page. At the top, there is a breadcrumb navigation: Amazon Elastic Container Service > Task definitions. Below the header, there is a search bar labeled "Filter task definitions by property or value" and a status filter "Status of last revision = ACTIVE" with a clear filters button. On the right side of the header, there are four buttons: "Deploy" (with a dropdown arrow), "Create new revision" (with a dropdown arrow), and "Create new task definition" (highlighted in orange). Below the header, there is a table with the following columns: Task definition and Status of last revision. The table contains one row for the task definition "demo-springboot-ecs-fargate". The "Status of last revision" column shows "ACTIVE".

Create Task Definitions

Task definition configuration

Task definition family [Info](#)
Specify a unique task definition family name.
springboot-ecs-fargate-demo
Up to 255 letters (uppercase and lowercase), numbers, hyphens, and underscores are allowed.

Container - 1 [Info](#)

Container details
Specify a name, container image, and whether the container should be marked as essential. Each task definition must have at least one essential container.

Name	Image URI	Essential container
campus3	docker.io/weder96/campus-docker:0.0.3	<input checked="" type="checkbox"/> Yes

Private registry [Info](#)
Store credentials in Secrets Manager, and then use the credentials to reference images in private registries.
 Private registry authentication

Port mappings [Info](#)

Add port mappings to allow the container to access ports on the host to send or receive traffic. Any changes to port mappings configuration impacts the associated service connect settings.

Container port	Protocol	Port name	App protocol	
80	TCP	campus3-80-tcp	HTTP	<input type="button" value="Remove"/>
8080	TCP	campus3-8080-tcp	HTTP	<input type="button" value="Remove"/>

[Add more port mappings](#)

weder96 Repositories campus-docker General

General Tags Builds Collaborators Webhooks Settings

weder96 / campus-docker

Description
test image springboot and docker Hub campus-docker [Edit](#)

Last pushed: 2 days ago

Tags

This repository contains 2 tag(s).

Tag	OS	Type	Pulled	Pushed
0.0.3		Image	2 days ago	2 days ago
0.0.2-SNAPSHOT		Image	2 days ago	3 days ago

[See all](#) [Go to Advanced Image Management](#)

Linked Task and Cluster

Amazon Elastic Container Service > Clusters > cls-demo-springboot > Tasks > b21424b43a7944ffadd3ba82d6cec21 > Configuration

b21424b43a7944ffadd3ba82d6cec21

Configuration Logs Networking Tags

Task overview

ARN
 b21424b43a7944ffadd3ba82d6cec21

Last status
 Running

Desired status
 Running

Started/Created at
6/6/2023, 17:33:08 UTC
6/6/2023, 17:32:18 UTC

Configuration

Operating system/Architecture
Linux/X86_64

CPU | Memory
1 vCPU | 3 GB

Platform version
1.4.0

Capacity provider
-

Launch type
FARGATE

Task definition
demo-springboot-ecs-fargate:1

Task group
family:demo-springboot-ecs-fargate

ENI ID
eni-03497174f6cb11c79 

Network mode
awsvpc

Subnet ID
subnet-0390fff04f7c33786 

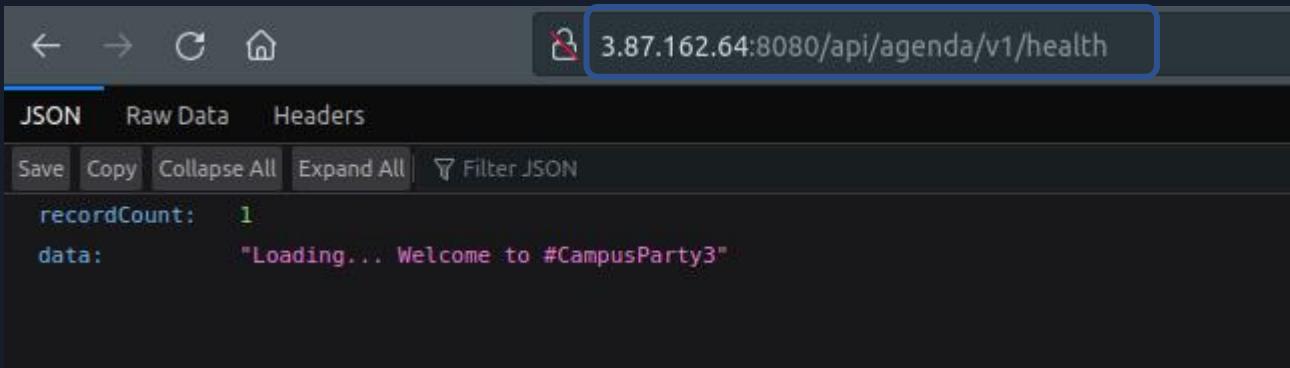
Public IP
 3.87.162.64 | open address 

Private IP


MAC address


Log Events - Cloud Watch

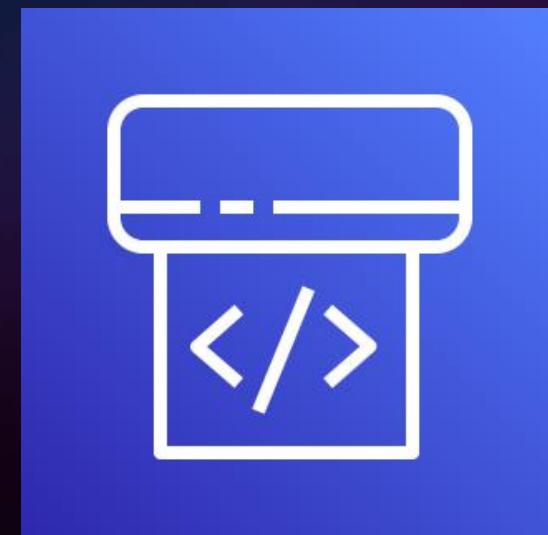
Call API by Public IP:Port



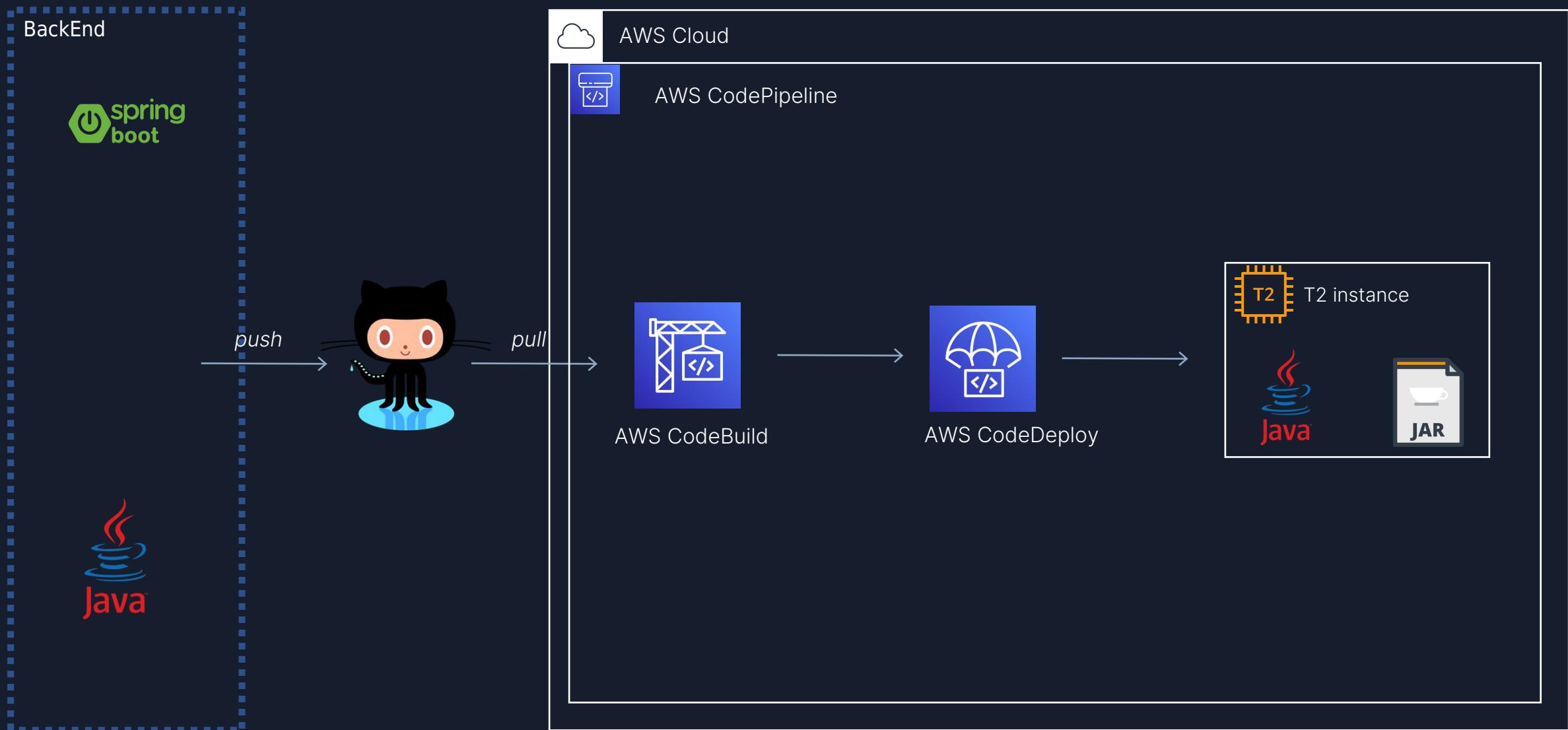
A screenshot of a browser window displaying a JSON response. The URL in the address bar is `3.87.162.64:8080/api/agenda/v1/health`. The browser interface includes standard navigation buttons (back, forward, home) and a toolbar with options like JSON, Raw Data, Headers, Save, Copy, Collapse All, Expand All, and Filter JSON.

```
recordCount: 1
data: "Loading... Welcome to #CampusParty3"
```

8. Pipelines com Java Spring boot



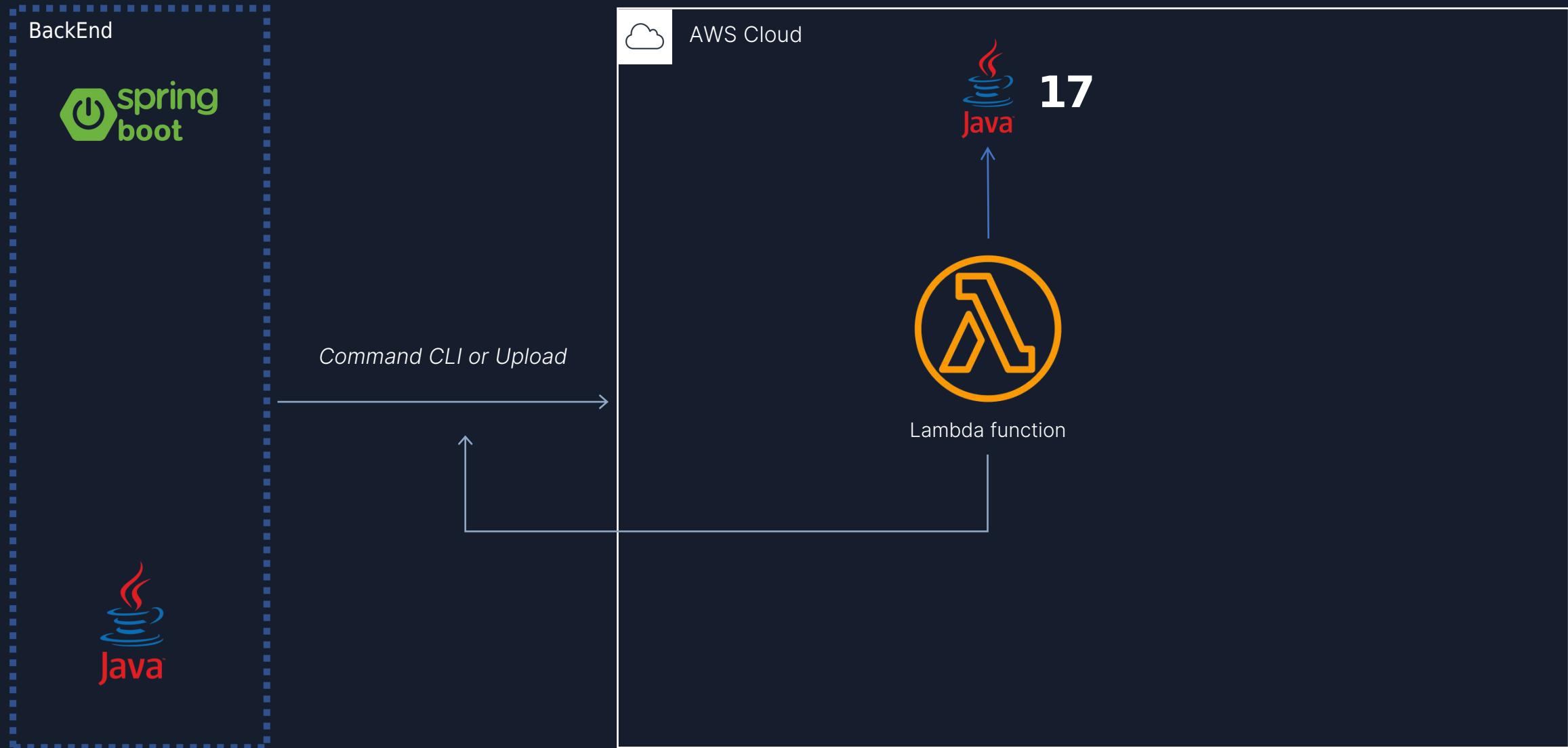
Architecture (Codepipeline)



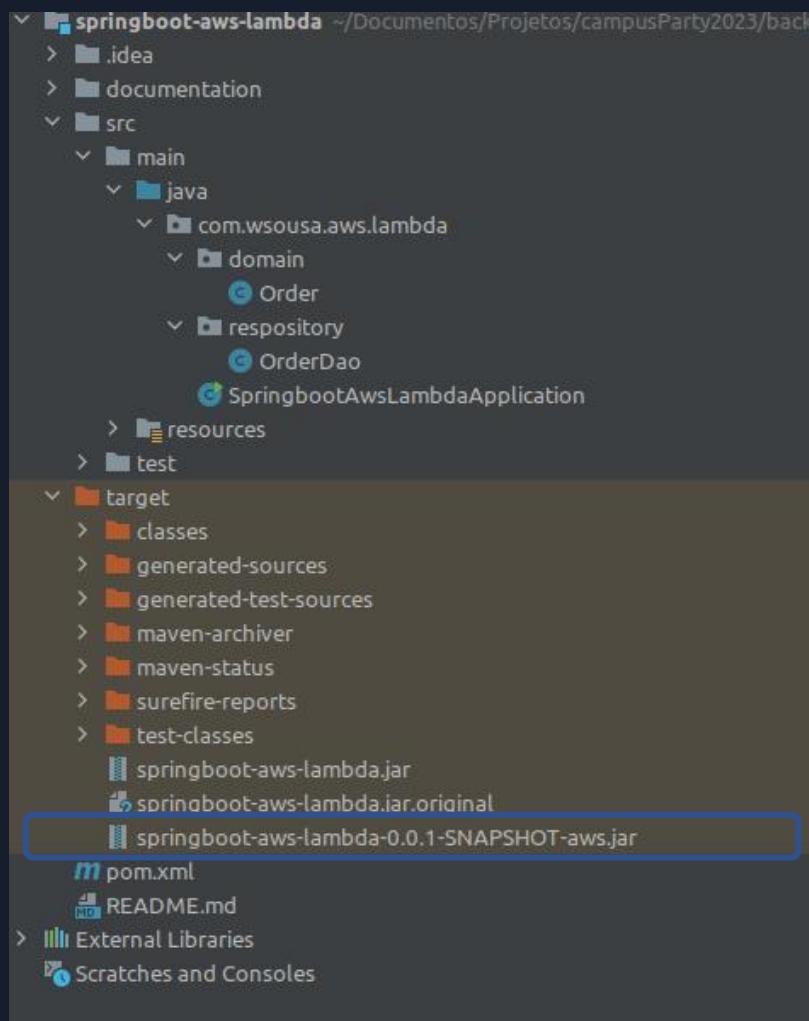
9. Serverless - Java and Lambda



Architecture (Serverless)



Springboot Lambda Demo



<https://github.com/weder96/springboot-aws-lambda>

<https://docs.spring.io/spring-cloud-function/docs/current/reference/html/aws.html>

```
<properties>
    <java.version>17</java.version>
    <spring-cloud.version>2021.0.7</spring-cloud.version>
    <wrapper.version>1.0.29.RELEASE</wrapper.version>
    <aws.java.sdk.version>2.20.80</aws.java.sdk.version>
</properties>
```

```
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-deploy-plugin</artifactId>
    <configuration>
        <skip>true</skip>
    </configuration>
</plugin>
<plugin>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-maven-plugin</artifactId>
    <dependencies>
        <dependency>
            <groupId>org.springframework.boot.experimental</groupId>
            <artifactId>spring-boot-thin-layout</artifactId>
            <version>${wrapper.version}</version>
        </dependency>
    </dependencies>
</plugin>
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-shade-plugin</artifactId>
    <version>3.2.4</version>
    <configuration>
        <createDependencyReducedPom>false</createDependencyReducedPom>
        <shadedArtifactAttached>true</shadedArtifactAttached>
        <shadedClassifierName>aws</shadedClassifierName>
    </configuration>
</plugin>
</plugins>
<finalName>springboot-aws-lambda</finalName>
```

Springboot Lambda Demo

Nome	Tamanho	Tipo
classes	2 itens	Pasta
generated-sources	1 item	Pasta
generated-test-sources	1 item	Pasta
maven-archiver	1 item	Pasta
maven-status	1 item	Pasta
surefire-reports	2 itens	Pasta
test-classes	1 item	Pasta
springboot-aws-lambda.jar	15,1 kB	Arquivo
springboot-aws-lambda.jar.original	7,3 kB	Arquivo
springboot-aws-lambda-0.0.1-SNAPSHOT-aws.jar	15,2 MB	Arquivo

Create Lambda Function Java 17

Lambda > Functions

Functions (1)

Last fetched 10 seconds ago

Actions ▾ Create function

Filter by tags and attributes or search by keyword

Function name	Description	Package type	Runtime	Last modified
weder-function	-	Zip	Java 17	3 hours ago

<https://github.com/weder96/springboot-aws-lambda>

Springboot Lambda Demo

#Upload Jar generate command line mvn clean Install

The screenshot shows the AWS Lambda function configuration page for a function named "weder-function". The "Code source" section indicates that the deployment package is too large for inline code editing. The "Code properties" section shows a package size of 14.5 MB and a SHA256 hash. The "Runtime settings" section shows Java 17 as the runtime, org.springframework.cloud.function.adapter.aws.FunctionInvoker::handleRequest as the handler, and x86_64 as the architecture. There are "Edit" and "Edit runtime management configuration" buttons.

#Runtime Settings

Edit runtime settings

The dialog box shows the "Runtime settings" configuration. It includes fields for "Runtime" (set to Java 17), "Handler" (set to org.springframework.cloud.function.adapter.aws.FunctionInvoker::handleRequest), and "Architecture" (set to x86_64). A "Save" button is at the bottom right.

Runtime settings

Runtime

Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Java 17

Handler

org.springframework.cloud.function.adapter.aws.FunctionInvoker::handleRequest

Architecture

Choose the instruction set architecture you want for your function code.

x86_64

arm64

Cancel

Save

Springboot Lambda Demo

#Environment variables

Configuration	Aliases	Versions
Environment variables (1)		
The environment variables below are encrypted at rest with the default Lambda service key.		
Key		Value
FUNCTION_NAME		orders

#Test

Code | **Test** | Monitor | Configuration | Aliases | Versions

Executing function: succeeded (logs [\[2\]](#))
▶ Details

Test event [Info](#) Delete Save Test

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save changes.

Test event action

Create new event Edit saved event

Event name

testEvent ▾ C

Event JSON Format JSON

```
1 "Book"
```

<https://github.com/weder96/springboot-aws-lambda>

Springboot Lambda Demo

#Result Test Invoke

 Executing function: succeeded (logs 

▼ Details

The area below shows the last 4 KB of the execution log.

```
[  
 {  
 "id": 102,  
 "type": "Book",  
 "title": "Programming Aws Lambda: Build and Deploy Serverless Applications with Java",  
 "price": 69,  
 "quantity": 2  
 },  
 {  
 "id": 103,
```

Summary

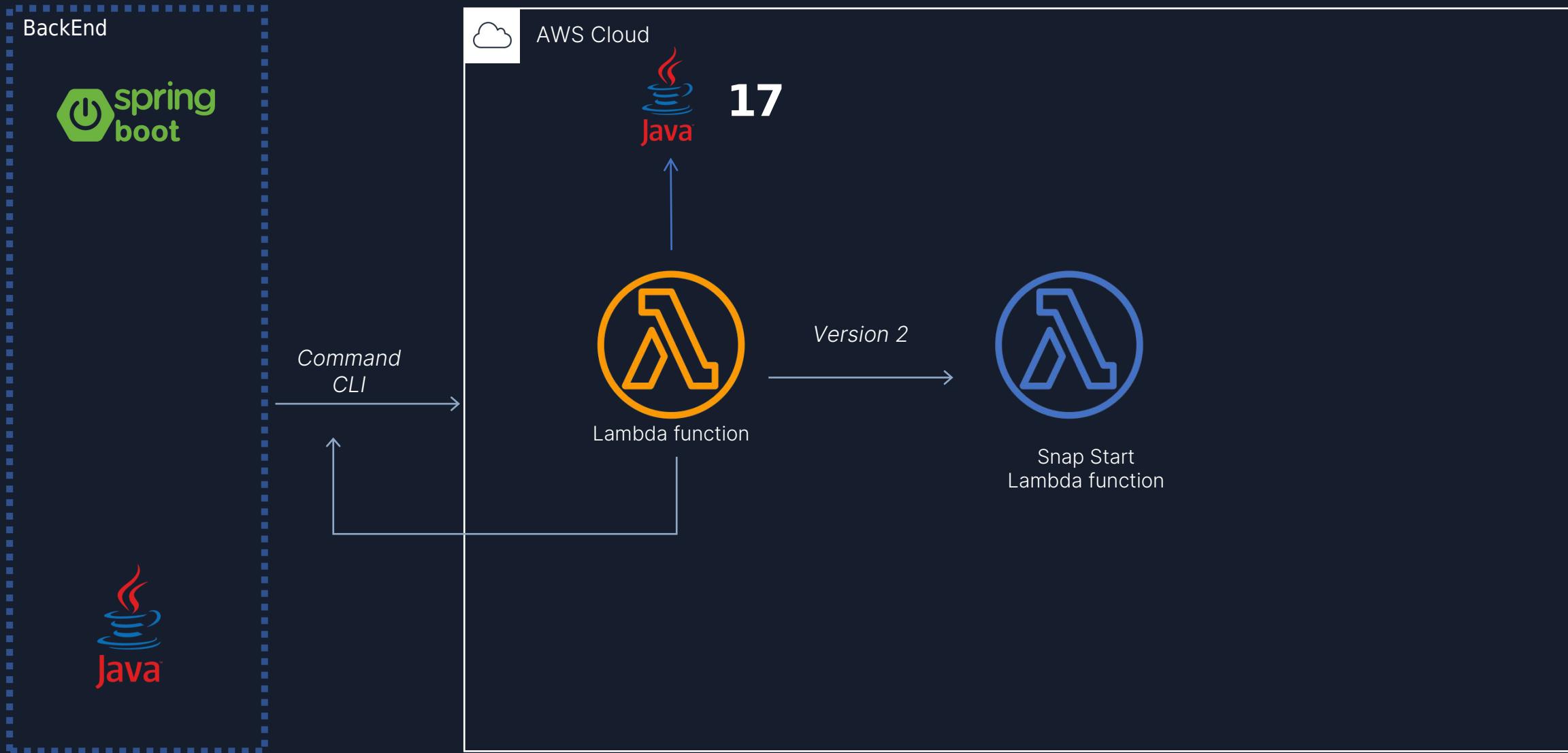
Code SHA-256	Request ID
EBBqgKt0bUkveKJmWd5bUObuDBu7BPCgkd4mV11X6go=	6fe2da13-5dbc-4729-9915-f2a64887f339
Init duration	Duration
1843.54 ms	496.38 ms
Billed duration	Resources configured
497 ms	512 MB
Max memory used	
122 MB	

Log output

10. AWS Lambda SnapStart for Spring Developers



Architecture (Serverless)



Reducing Cold Start

SNAPSTART

Init during deployment



(optional)

Resume

Invoke

Invoke

first request →

Resume Snapshot

Post Snapshot Hook

(optional)

Code execution

Code execution

AWS Lambda SnapStart + Spring Cloud Function

Executing function: succeeded (logs 

▼ Details

The area below shows the last 4 KB of the execution log.

```
"3ytraPsupmaC"
```

Summary

Code SHA-256
aPV4xuwzGv9fGcBotJFF8/RvB5aeK4t7I8uQVT5sDs=

Init duration 2776.64 ms	Request ID 29283a7a-6c03-4981-8fe4-03678bd45610
Billed duration 283 ms	Duration 282.82 ms
Max memory used 144 MB	Resources configured 512 MB

Log output

The section below shows the logging calls in your code. [Click here](#) to view the corresponding CloudWatch log group.

```
"3ytraPsupmaC"
```

Executing function: succeeded (logs 

▼ Details

The area below shows the last 4 KB of the execution log.

```
"3ytraPsupmaC"
```

Summary

Code SHA-256
aPV4xuwzGv9fGcBotJFF8/RvB5aeK4t7I8uQVT5sDs=

Duration 2.46 ms	Request ID 65eb0061-edb3-4a49-a88f-389ee218e831
Resources configured 512 MB	Billed duration 3 ms
	Max memory used 145 MB

Log output

The section below shows the logging calls in your code. [Click here](#) to view the corresponding CloudWatch log group.

```
START RequestId: 65eb0061-edb3-4a49-a88f-389ee218e831 Version: $LATEST
2023-06-08 15:39:02.607 INFO 8 --- [      main] o.s.c.f.adapter.aws.AWSLambdaUtils    : Received: "CampusParty3"
END RequestId: 65eb0061-edb3-4a49-a88f-389ee218e831
REPORT RequestId: 65eb0061-edb3-4a49-a88f-389ee218e831 Duration: 2.46 ms Billed Duration: 3 ms Memory Size: 512 MB Max Memory Used: 145 MB
```

<https://github.com/weder96/snapstart>

AWS Lambda SnapStart + Spring Cloud Function

Edit basic settings

Basic settings [Info](#)

Description - optional

Memory [Info](#)
Your function is allocated CPU proportional to the memory configured.
512 MB

Set memory to between 128 MB and 10240 MB.

Ephemeral storage [Info](#)

You can configure up to 10 GB of ephemeral storage (/tmp) for your function. [View pricing](#) ▾
512 MB
Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.

SnapStart [Info](#)

Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the [SnapStart compatibility considerations](#) ▾

None

Supported runtimes: Java 11, Java 17.

Timeout

0 min 15 sec

Execution role

Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#) ▾

Use an existing role

Create a new role from AWS policy templates

Existing role

Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

service-role/snapstartFunction-role-vbvefy3

Cancel

Save

Code | Test | Monitor | Configuration | Aliases | **Versions**

General configuration

Triggers

Permissions

Destinations

Function URL

General configuration [Info](#)

Description

-

Timeout

0 min 15 sec

Memory

512 MB

[SnapStart](#) [Info](#)
[PublishedVersions](#)

Ephemeral storage
512 MB

Publish new version from \$LATEST

Publishing a new version saves a snapshot of the code and the configuration of the \$LATEST version. You need to deploy code changes in \$LATEST before you can create a new version. Any triggers you added to the function are not saved to the new version. You can't change the new version's code. Choose Publish to confirm. To change SnapStart, navigate to \$LATEST and choose the Configuration tab. On the General configuration pane, choose Edit.

Version description - optional

SnapStart

PublishedVersions

Cancel

Publish

AWS Lambda SnapStart + Spring Cloud Function

The screenshot displays the AWS Lambda console interface for a function named "snapstartFunction".

Function Overview: Shows the function name "snapstartFunction:1", 0 layers, and a success message: "Successfully created version 1 of function snapstartFunction."

Configuration: General configuration tab shows 512 MB of memory, 0 min 15 sec timeout, and On SnapStart optimization status.

Test: Executing function: succeeded (logs) - The logs show the output: "*3ytraPsupmaC*".

Summary: Includes details like Code SHA-256, Request ID, Duration (544.02 ms), and Resources configured (512 MB).

Metrics: Shows Billed restore duration (141 ms), Duration (544.02 ms), Billed duration (686 ms), and Max memory used (116 MB).

<https://github.com/weder96/snapstart>

Want to Try?

Workshop Java on AWS Lambda



<https://s12d.com/java-workshop>

aws workshop studio

Java on AWS Lambda

Java on AWS Lambda

Welcome Builders!

The Java programming language has adapted to a changing technology landscape for more than 25 years. It plays an essential role building sustainable software architectures and applications. The combination of efficiency and portability encourages businesses to build their applications using Java.

The introduction of [AWS Lambda](#) has changed the desired characteristics of programming languages. Each Lambda execution environment only processes a single invocation at a time, the lifetime of a function is short, and resources are constrained. In this model, the start-up time of an application and the memory footprint is critical.

The Java ecosystem is adapting and introduced several new projects to foster innovation. [Project Leyden](#) will address startup time, memory footprint and peak performance by introducing a concept of static images to the Java Platform, and to the JDK. The changes to the release cycle mean that innovation will be delivered faster. Open source projects like [GraalVM](#) are already able to create static images using their [native-image](#) tool.

Learning Objectives

Resources

<https://cdkworkshop.com>

<https://github.com/aws-samples/aws-cdk-examples>

<https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

<https://aws.amazon.com/pt/developer/language/java/>

<https://docs.aws.amazon.com/toolkit-for-jetbrains/latest/userguide/setup-toolkit.html>

<https://aws.amazon.com/pt/intellij/>

https://docs.aws.amazon.com/code-library/latest/ug/java_2_code_examples.html

https://docs.aws.amazon.com/pt_br/prescriptive-guidance/latest/patterns/deploy-a-ci-cd-pipeline-for-java-microservices-on-amazon-ecs.html

<https://docs.aws.amazon.com/lambda/latest/dg/lambda-java.html>

<https://aws.amazon.com/pt/blogs/compute/java-17-runtime-now-available-on-aws-lambda/>

<https://www.slideshare.net/AmazonWebServices/java-on-aws>

<https://www.jrebel.com/blog/aws-java-application-setup>

<https://www.slideshare.net/VadymKazulkin/adopting-java-for-the-serverless-world-at-jax-2022>

<https://towardsaws.com/deploy-spring-boot-application-to-aws-ec2-using-docker-f359e7ad2026>

<https://aws.amazon.com/pt/blogs/developer/stepfunctions-fluent-api/>

<https://aws.amazon.com/blogs/compute/java-17-runtime-now-available-on-aws-lambda/>

<https://docs.aws.amazon.com/lambda/latest/dg/snapstart.html>

About the speaker

Q & A

Weder Mariano de Sousa

Specialist Senior - GFT

Graduated **Computer Science**

Post Graduate in **Midias UFG**

Post Graduate in **Information Security**



<https://www.linkedin.com/in/wedermarianodesousa/>

<https://github.com/weder96>

<https://twitter.com/weder96>

<https://dev.to/weder96>



GOJava



AWS User Group Goiânia



THANK YOU