

PAN ACADEMY

# Desafio final

14 de Dezembro 2021

# Visão Geral

O que você precisa saber

**01** Nosso time

**02** O desafio

**03** Metodologias Ágeis

**04** Tecnologias utilizadas

**05** AWS e estrutura

**06** Escopo do projeto

**07** Serviços da API

**08** Testes

**09** Documentação

# Sobre o Time

A 6Devs com uma equipe multidisciplinar e autogerenciavel que combina a inteligência, tecnologia e expertise humana para fornecer soluções confiáveis.



# Nosso Time



CARLOS BETIOL  
TECH LEAD



[Carlos Betiol](#)



WESLLEY LIMA



[Weslley Lima](#)



TAISIS MARINELO



[Tassis Marinelo](#)



JONAS WILLIAM



[Jonas William](#)



RICHARDSON BRUNO



[Richardson Bruno](#)



CASSIANO AUGUSTO



[Cassiano Augusto](#)



O desafio  
Criar uma  
API para  
gerenciar  
transações da  
nova plataforma  
do BlueBank



# Metodologias e Ferramentas

Usamos um conjunto de ferramentas e  
práticas conduzir e concluir o projeto



## Google Meet



## Quadro Kanban Semana 4 - Data 24 de Novembro

The screenshot shows a Trello Kanban board titled "Squad 7 - Banco BlueBank". The board is divided into four main columns: "To do", "Doing", "Done", and "Revisão e correções Concluidas".

- To do (A fazer):**
  - Implementação final do DTO (Due 27 de nov, assigned to W)
  - Novo endpoint correntista/id/extrato (Due 27 de nov, assigned to JW)
  - Colocacao do digito verificador no payload de post da movimentacao (Due 29 de nov, assigned to JW)
  - Adicionar configurações finais essenciais para a Documentação ao Swagger (Due 30 de nov, assigned to W)
- Doing (Fazendo):**
  - Implementação inicial DTO (Due 20 de nov - 24 de nov, assigned to TM, W)
  - Criação do Service e Controller de Movimentação (Due 24 de nov, assigned to JW)
  - Ajuste nos pacotes de exceptions colocados pelo Wesley (Due 24 de nov, assigned to JW)
  - Adicionar validacao no modelo (Due 26 de nov, assigned to JW)
  - Adicionado restante da documentação do Swagger (Due 1 de dez, assigned to JW)
- Done (Concluído):**
  - EER (Due 10 de nov, assigned to JW, TM, W)
    - Diagrama caso de uso (Due 9 de nov, assigned to JW, TM, W)
  - Endpoint/contas GET/POST/PUT/DELETE (repository, service, controller) (Due 23 de nov, assigned to JW)
- Revisão e correções Concluidas:**
  - Revisão (Due 13 de nov, assigned to JW)
  - Resolvendo o problema do Jpa (Due 19 de nov, assigned to JW, 3/3)
  - Adição de exception (Due 22 de nov, assigned to JW)
  - Ajuste na classe correntista (Due 23 de nov, assigned to JW)
  - Solucionar o problema de correntista sobre endereços e contatos (Due 23 de nov, assigned to JW)
  - Ajustar método dentro de Correntista service (Due 23 de nov, assigned to JW)



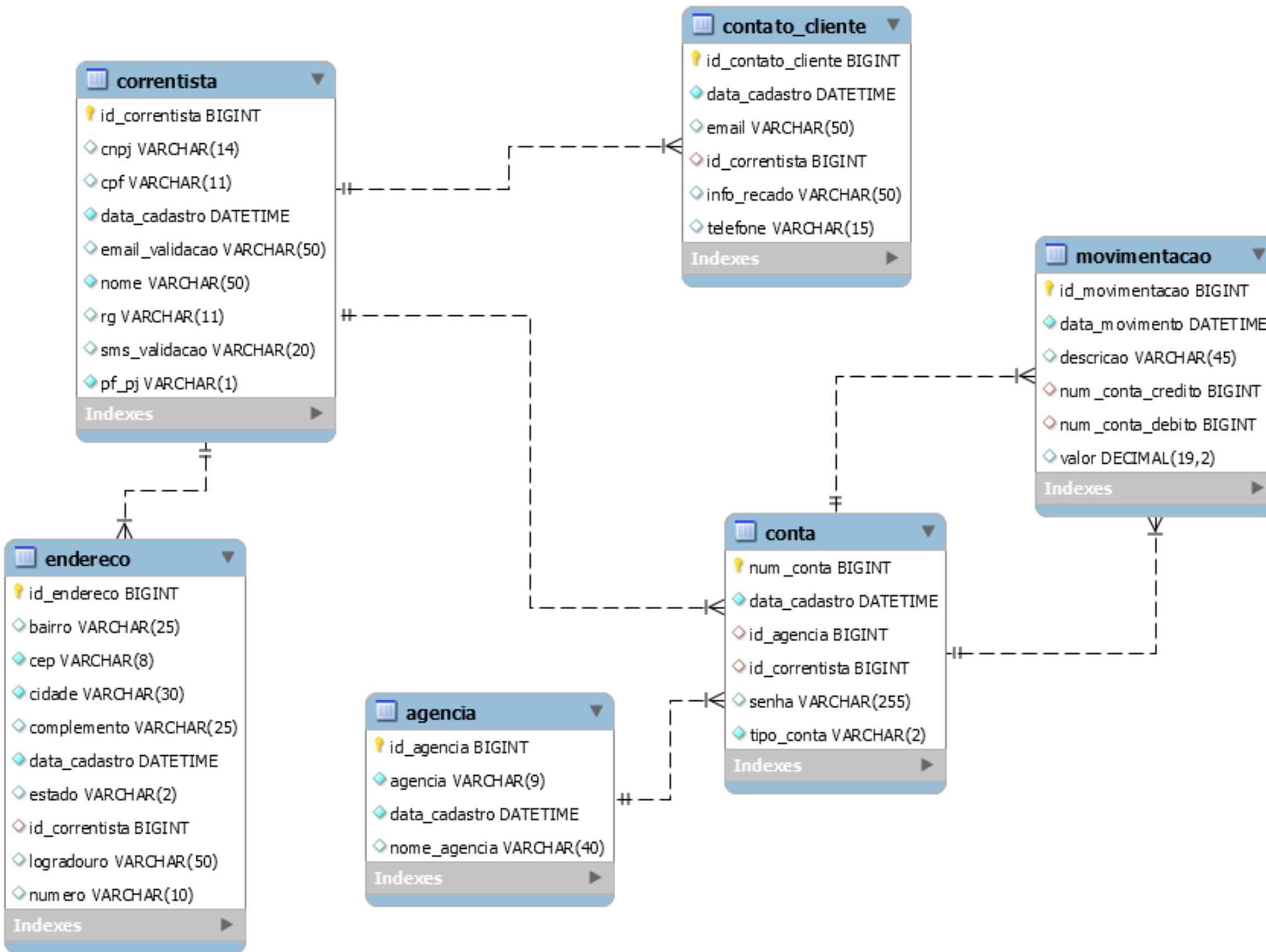
As tarefas foram divididas e acompanhadas conforme prioridade. Criado uma coluna à parte para administrar revisões e correções concluidas

# Tecnologias utilizadas

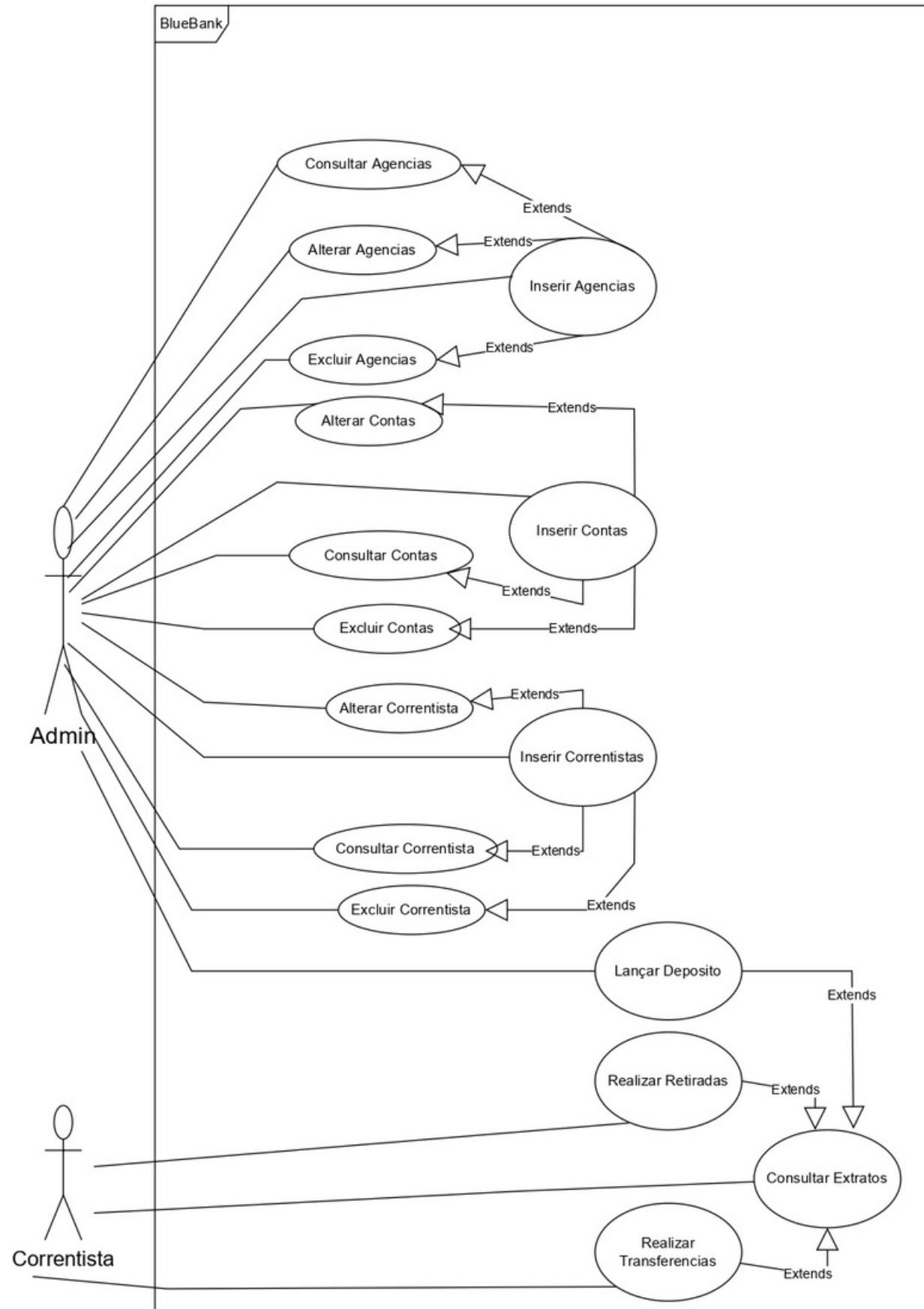
1. Java 11
2. Spring Boot: Spring Data & Spring Security
3. JUnit
4. MAVEN
5. Swagger
6. AWS Lambda
7. ECR - Elastic Container Registry
8. RDS - Relational Database Service
9. ECS - Elastic Container Service
10. FARGATE - computação Serverless
11. CloudWatch
12. ELB - Elastic Load Balancing
13. EC2 - Elastic Compute Cloud
14. SNS - Simple Notification Service
15. Lambda
16. API Gateway
17. AWS: AWS Certificate Manager
18. AWS Systems Manager (gerenciamento de senhas e configurações)



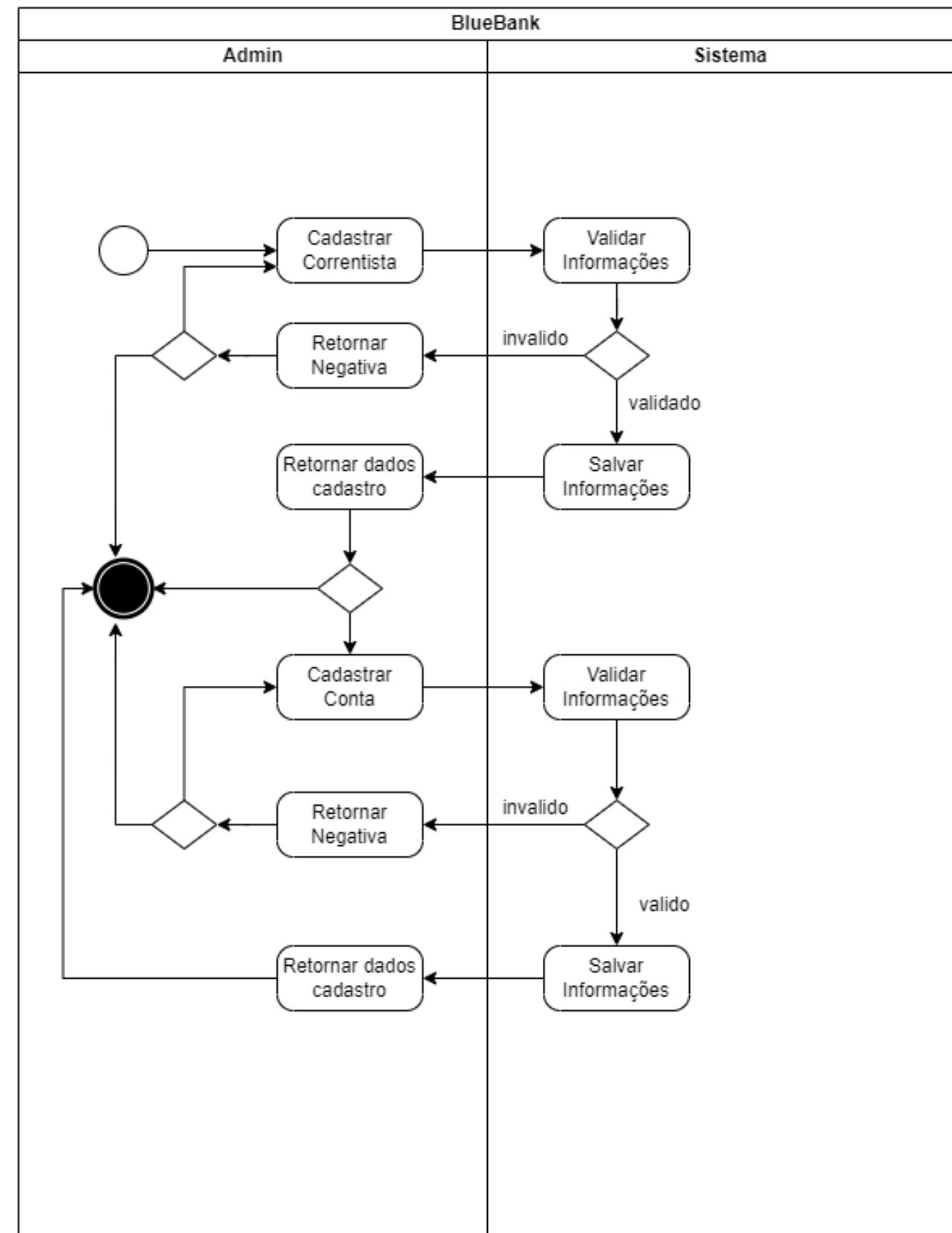
# EER - DIAGRAMA ENTIDADE-RELACIONAMENTO



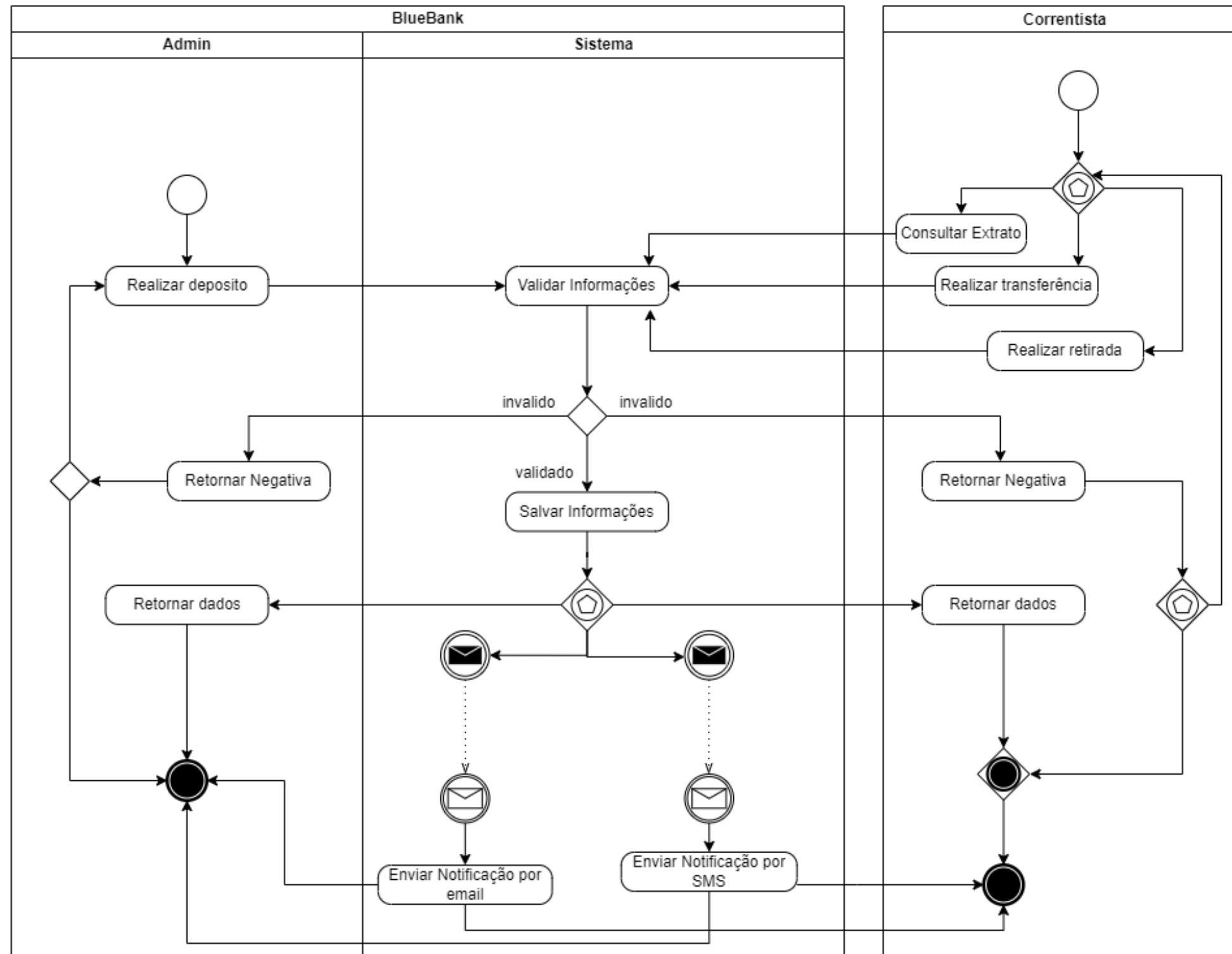
# DER - DIAGRAMA DE CASO DE USO



# BPMN - DIAGRAMA DE FLUXO CONTA



# BPMN - DIAGRAMA DE FLUXO MOVIMENTAÇÃO



# Escopo do Projeto

## Funcionalidades da API

- Gerenciamento de agências (CRUD completo)

- Gerenciamento de clientes (CRUD completo)

- Gerenciamento de contas bancárias (CRUD completo)

- Opção para realizar depósito, saque, transferência e gerar extratos

- Sistema de autenticação, rastreamento de IP e notificação via e-mail

# 30 funções disponíveis na API



The screenshot shows a list of API endpoints for 'BlueBank AWS'. The endpoints are categorized under 'BlueBank AWS' and include various HTTP methods (GET, POST, PUT, DELETE) and resource names. A large circular icon with a pen nib is visible on the right side of the screen.

- + ⌂ ⌂ ⌂
- BlueBank AWS
  - GET API hostcheck
  - GET Agencias Lista
  - GET Agencia por Id
  - DEL Agencia delete
  - PUT Agencia update
  - POST Agencia create
  - GET Correntistas Lista
  - GET Correntistas por Id
  - POST Correntistas create
  - PUT Correntistas update
  - DEL Correntistas delete
  - GET Correntistas Lista Enderecos
  - POST Correntistas add endereço
  - DEL Correntistas delete endereço
  - GET Correntistas Lista Contatos
  - POST Correntistas add contato
  - DEL Correntistas delete contato
  - GET Contas Lista
  - GET Contas Extrato
  - GET Contas Saldo
  - GET Contas por Id
  - POST Contas create
  - PUT Contas update
  - DEL Contas delete
  - GET Movimentacoes por Id
- Open Scratch Pad Overview

Postman

File Edit View Help

Home Workspaces Reports Explore

Search Postman

Working locally in Scratch Pad. Switch to a Workspace

Scratch Pad

New Import + ...

No Environment

Collections

+ =

BlueBank AWS

APIs

GET API hostcheck

GET Agencias Lista

GET Agencia por Id

Agencia delete

PUT Agencia update

POST Agencia create

Environment

Mock Servers

Monitors

History

Open Scratch Pad Overview

Find and Replace Console

Runner Trash

Digite aqui para pesquisar

23°C Limpo 21:10  
12/12/2021

The screenshot shows the Postman application window. The main title bar says "Postman". The menu bar includes "File", "Edit", "View", and "Help". Below the menu is a navigation bar with "Home", "Workspaces", "Reports", and "Explore". A search bar "Search Postman" is at the top right. A yellow banner across the top says "Working locally in Scratch Pad. Switch to a Workspace". The left sidebar has sections for "Collections", "APIs", "Environments", "Mock Servers", "Monitors", and "History". The "Collections" section is expanded, showing a collection named "BlueBank AWS" which contains several API endpoints: "GET API hostcheck", "GET Agencias Lista", "GET Agencia por Id", "Agencia delete", "PUT Agencia update", and "POST Agencia create". The "APIs" section also lists "GET Correntistas Lista", "GET Correntistas por Id", "POST Correntistas create", "PUT Correntistas update", "DELETE Correntistas delete", "GET Correntistas Lista Enderecos", "POST Correntistas add endereco", "DELETE Correntistas delete endereco", "GET Correntistas Lista Contatos", "POST Correntistas add contato", "DELETE Correntistas delete contato", "GET Contas Lista", "GET Contas Extrato", "GET Contas Saldo", and "GET Contas por Id". At the bottom, there are buttons for "Find and Replace" and "Console", and a toolbar with icons for "Runner" and "Trash". The status bar at the bottom right shows the date and time: "23°C Limpo 21:10 12/12/2021".

TEST

TEST

The screenshot shows a Java development environment with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Project Explorer:** Shows the project structure for "Banco-BlueBank".
- Code Editor:** Displays the content of the file `ContaEndPointIT.java`. The code is a Spring Boot integration test using RestAssured to test endpoints for accounts.
- Run/Debug:** Shows 0/0 runs, 0 errors, and 0 failures.

```
1 package com.banco.bluebank.service;
2
3
4 import static org.mockito.ArgumentMatchers.isNull;
5
6
7 @RunWith(SpringRunner.class)
8 @SpringBootTest(webEnvironment = SpringBootTest.WebEnvironment.RANDOM_PORT)
9 @TestPropertySource("/application-test.properties")
10 public class ContaEndPointIT {
11
12     @LocalServerPort
13     private int Webport;
14
15     @Before
16     public void iniciar() {
17         RestAssured.port = Webport;
18     }
19
20
21     @Test
22     public void deveRetornar200AoListarContas() {
23
24         RestAssured.enableLoggingOfRequestAndResponseIfValidationFails();
25
26         RestAssured.given().basePath("/contas").accept(MediaType.JSON)
27             .when().get().then().statusCode(HttpStatus.OK.value());
28     }
29
30
31     @Test
32     public void deveRetornar404AoListarContasComErroNoPath() {
33
34         RestAssured.enableLoggingOfRequestAndResponseIfValidationFails();
35
36         RestAssured.given().basePath("/conta").accept(MediaType.JSON)
37             .when().get().then().statusCode(HttpStatus.NOT_FOUND.value());
38     }
39
40
41     @Test
42     public void deveRetornar200AoBuscarPorIdContas() {
43
44         RestAssured.enableLoggingOfRequestAndResponseIfValidationFails();
45
46         RestAssured.given().basePath("/contas").pathParam("id", 42)
47             .accept(MediaType.JSON).when().get("/{id}")
48
49
50 }
```

```
graph TD; A((Initiation  
Phase 1)) --> B((Planning  
Phase 2)); B --> C((Execution  
Phase 3)); C --> D((Closure and  
Analysis  
Phase 4)); D --> A
```

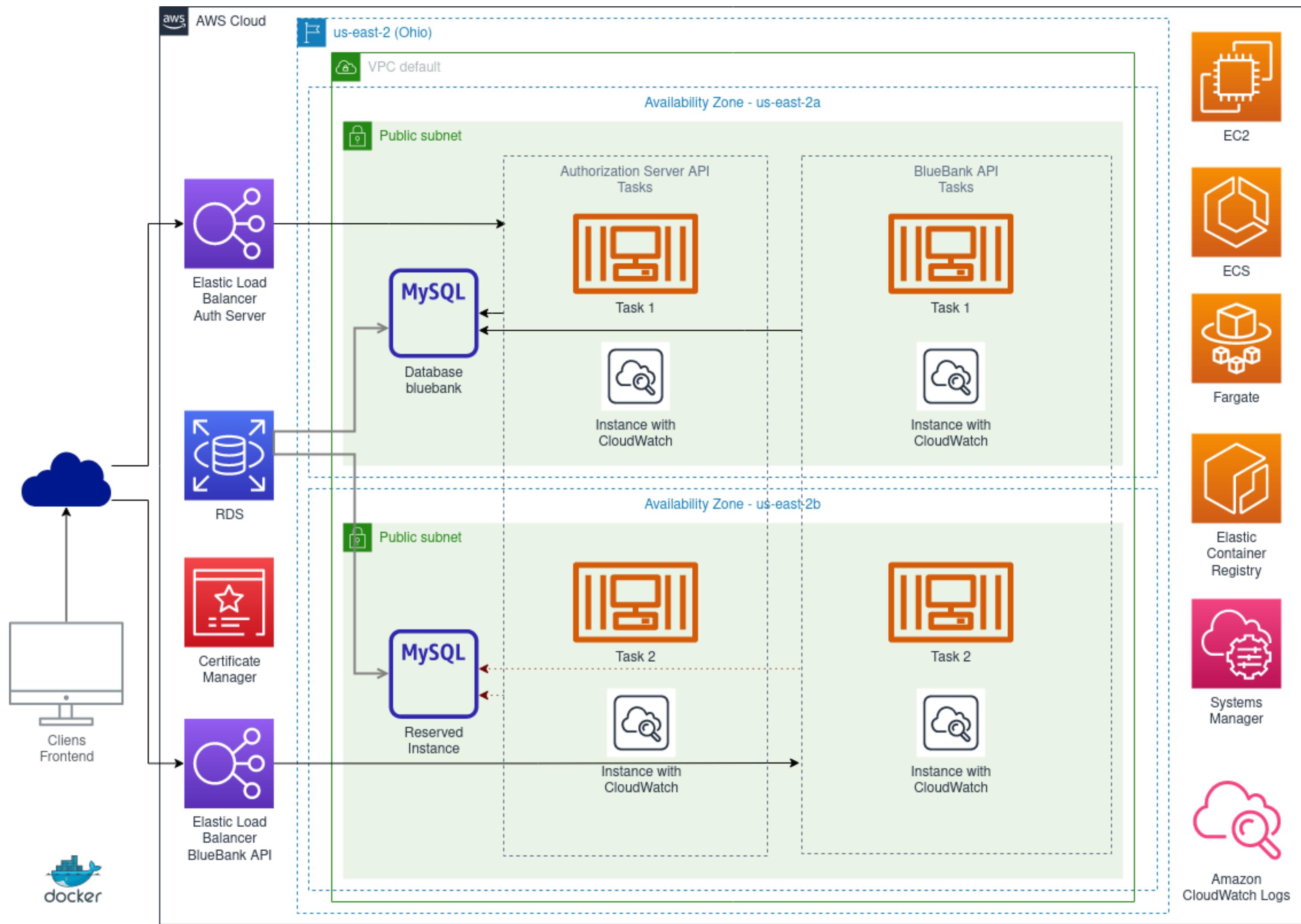
Initiation  
Phase 1

Planning  
Phase 2

Execution  
Phase 3

Closure and  
Analysis  
Phase 4

# AWS



# RDS

The screenshot shows the AWS RDS console interface. On the left, there's a sidebar with various navigation options like Painel, Bancos de dados, Editor de consultas, etc. The main area is titled 'Bancos de dados' and shows a table with one row highlighted in blue. The table columns include 'Identificador de banco de dados', 'Função', 'Região e AZ', 'Tamanho', 'Status', and 'Multi-AZ'. The highlighted row contains the values: 'bluebank', 'Instância', 'us-east-2a', 'db.t2.micro', 'Disponível', and 'Sim'. A red box highlights this row. A red arrow points from this row to a callout bubble containing the text: 'Instância de database mysql no RDS configurado para multi-AZ com uma réplica em espera em Zona de Disponibilidade diferente'. Below this, another red box highlights a row in a 'Reserved instances' table. This table has columns: 'Reservation Id', 'Product Desc', 'Class', and 'Offering Type'. The highlighted row contains: 'bluebank-database-replica', 'mysql', 'db.t2.micro', and 'No Upfront'. A red arrow points from this row to the same callout bubble.

**Bancos de dados**

Identificador de banco de dados	Função	Região e AZ	Tamanho	Status	Multi-AZ
bluebank	Instância	us-east-2a	db.t2.micro	Disponível	Sim

**Reserved instances (1)**

Reservation Id	Product Desc	Class	Offering Type
bluebank-database-replica	mysql	db.t2.micro	No Upfront

Instância de database mysql no RDS configurado para multi-AZ com uma réplica em espera em Zona de Disponibilidade diferente

# ECS - FARGATE

**Fargate serverless rodam sobre instâncias EC2 totalmente gerenciadas pela AWS**

**Duas instancias do Authorization Server em diferentes Zonas de Disponibilidade**

**Duas instancias da API em diferentes Zonas de Disponibilidade**

**Uso de Serverless FARGATE**

Cluster ARN: arn:aws:ecs:us-east-2:653929532121:cluster/bluebank-cluster

Status: ACTIVE

Registered container instances: 0

Pending tasks count: 0 Fargate, 0 EC2, 0 External

Running tasks count: 4 Fargate, 0 EC2, 0 External

Active service count: 2 Fargate, 0 EC2, 0 External

Draining service count: 0 Fargate, 0 EC2, 0 External

Last updated on December 12, 2021 4:34:10 PM (0m ago)

Task	Task definition	Contain...	Last status	Desired status...	Started at	Started By	Group	Launch type	Platform
5264f4d76f104...	bluebank-auth-taskdef:1	-	RUNNING	RUNNING	2021-12-11 10:...	ecs-svc/06487...	service:blueba...	FARGATE	1.4.0
5467c2ef1b72...	bluebank-auth-taskdef:1	-	RUNNING	RUNNING	2021-12-11 10:...	ecs-svc/06487...	service:blueba...	FARGATE	1.4.0
58b6f91796ce...	bluebank-api-taskdef:1	-	RUNNING	RUNNING	2021-12-12 15:...	ecs-svc/18930...	service:blueba...	FARGATE	1.4.0
e1e807aced60...	bluebank-api-taskdef:1	-	RUNNING	RUNNING	2021-12-12 15:...	ecs-svc/18930...	service:blueba...	FARGATE	1.4.0

# SYSTEM MANAGER

The screenshot shows the AWS Systems Manager Parameter Store interface. The top navigation bar includes the AWS logo, a 'Services' dropdown, a search bar with placeholder text 'Search for services, features, blogs, docs, and more' and a keyboard shortcut '[Alt+S]', and a three-line menu icon.

The main navigation path is 'AWS Systems Manager > Repositório de parâmetros'. Below this, there are three tabs: 'Meus parâmetros' (selected), 'Parâmetros públicos', and 'Configurações'.

The main content area is titled 'Meus parâmetros' and contains a search bar. A red arrow points from the text 'Uso de armazenamento de parâmetros do AWS Systems Manager para variáveis de ambiente com informações sensíveis' to the table below.

A red box highlights the first seven rows of the table, which list environment variables for the '/prod/bluebank-api-service/' namespace:

<input type="checkbox"/>	Nome	Nível	Tipo
<input type="checkbox"/>	/prod/bluebank-api-service/AUTHORIZATION_PASSWORD	Padrão	SecureString
<input type="checkbox"/>	/prod/bluebank-api-service/DATABASE	Padrão	String
<input type="checkbox"/>	/prod/bluebank-api-service/DATABASE_HOST	Padrão	String
<input type="checkbox"/>	/prod/bluebank-api-service/DATABASE_PASSWORD	Padrão	SecureString
<input type="checkbox"/>	/prod/bluebank-api-service/DATABASE_USER	Padrão	String
<input type="checkbox"/>	/prod/bluebank-api-service/EMAIL_PASSWORD	Padrão	SecureString
<input type="checkbox"/>	/prod/bluebank-api-service/KEYSTORE_PASSWORD	Padrão	SecureString

The text 'Uso de armazenamento de parâmetros do AWS Systems Manager para variáveis de ambiente com informações sensíveis' is highlighted with a red oval and has a red arrow pointing to the table.

# ECR

aws | Services | Search for services, features, blogs, docs, and more [Alt+S] | Ohio | Carlos Be

Amazon Container Services X

Amazon ECS

Clusters

Task definitions

Amazon EKS

Clusters

Amazon ECR

**Repositories**

Private registry

Public registry

Public gallery

Amazon ECR > Repositories

Private Public

Private repositories (2)

Find repositories

C View push commands Delete Edit Create repository < 1 > ⚙

Repository name	URI	Created at	Tag immutability	Scan frequency	Encryption type	Pull through cache
bluebank-api	653929532121.dkr.ecr.us-east-2.amazonaws.com/bluebank-api	10 de dezembro de 2021, 21:33:58 (UTC-03)	Disabled	Scan on push	AES-256	Inactive
bluebank-auth-server	653929532121.dkr.ecr.us-east-2.amazonaws.com/bluebank-auth-server	10 de dezembro de 2021, 21:57:06 (UTC-03)	Disabled	Scan on push	AES-256	Inactive

Repositórios de containers utilizados para o deploy

Configurados para entrega contínua através do register level scan filter

Repository name	URI	Created at	Tag immutability	Scan frequency	Encryption type	Pull through cache
bluebank-api	653929532121.dkr.ecr.us-east-2.amazonaws.com/bluebank-api	10 de dezembro de 2021, 21:33:58 (UTC-03)	Disabled	Scan on push	AES-256	Inactive
bluebank-auth-server	653929532121.dkr.ecr.us-east-2.amazonaws.com/bluebank-auth-server	10 de dezembro de 2021, 21:57:06 (UTC-03)	Disabled	Scan on push	AES-256	Inactive

# ECR

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

Amazon Container Services X

Amazon ECS

Clusters

Task definitions

Amazon EKS

Clusters

Amazon ECR

Repositories

Images

Permissions

Lifecycle Policy

Tags

Private registry

Public registry

Public gallery

Amazon ECR > Repositories > bluebank-api

## bluebank-api

### Images (5)

Find Images

<input type="checkbox"/>	Image tag	Pushed at	Size (MB)	Image URI	Digest
<input type="checkbox"/>	latest	12 de dezembro de 2021, 15:40:24 (UTC-03)	271.61	<input type="checkbox"/> Copy URI	<input type="checkbox"/> sha256:cc4d704677a1ce5...
<input type="checkbox"/>	<untagged>	12 de dezembro de 2021, 14:53:03 (UTC-03)	271.61	<input type="checkbox"/> Copy URI	<input type="checkbox"/> sha256:23b7168711f40c...
<input type="checkbox"/>	<untagged>	11 de dezembro de 2021, 10:51:25 (UTC-03)	272.96	<input type="checkbox"/> Copy URI	<input type="checkbox"/> sha256:2154bd94f60f97...
<input type="checkbox"/>	<untagged>	11 de dezembro de 2021, 09:36:48 (UTC-03)	272.96	<input type="checkbox"/> Copy URI	<input type="checkbox"/> sha256:6df7ded919ebe6...
<input type="checkbox"/>	<untagged>	10 de dezembro de 2021, 21:55:44 (UTC-03)	272.96	<input type="checkbox"/> Copy URI	<input type="checkbox"/> sha256:bd8269eabf3d54...

Algumas imagens de containers utilizadas em deployments

# CLOUD WATCH

The screenshot shows the AWS CloudWatch Logs interface. On the left, a sidebar menu is open under the 'Logs' section, listing 'Grupos de logs', 'Log Insights', 'Métricas', 'Eventos', 'Monitoramento de aplicações', 'Insights', 'Configurações', and 'Conceitos básicos'. The main pane displays log events for a specific log group: /ecs/bluebank-api-taskdef > ecs/bluebank-api-container/58b6f91796ce47feac3774b36c18c36f. The interface includes a filter bar at the top with options like 'View as text', 'Ações', and 'Create Metric Filter', and a timestamp range selector from 'Clear' to 'Custom'. The log table has columns for 'Timestamp' and 'Mensagem'. Most log entries are truncated, showing only the timestamp and a series of underscores and slashes. One entry is fully visible: '2021-12-12T15:48:42.671-03:00 :: Spring Boot :: (v2.5.6)'. Another entry shows the application starting: '2021-12-12 18:48:42.994 INFO 1 --- [main] com.banco.bluebank.BluebankApplication : Starting BluebankApplication vlates...'. A red box highlights this starting log entry, and a red callout bubble with the text 'Visualização dos logs da aplicação no CloudWatch' points to it.

CloudWatch

Search for services, features, blogs, docs, and more [Alt+S]

Favoritos

Painéis

Alarmes 0

Logs

Grupos de logs

Log Insights

Métricas

Eventos

Monitoramento de aplicações

Insights

Configurações

Conceitos básicos

CloudWatch > Log groups > /ecs/bluebank-api-taskdef > ecs/bluebank-api-container/58b6f91796ce47feac3774b36c18c36f

Eventos de log

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

View as text C Ações Create Metric Filter

Filtrar eventos Clear 1m 30m 1h 12h Custom Custom

Timestamp Mensagem

Não há eventos mais antigos no momento. Tentar novamente

2021-12-12T15:48:42.671-03:00 :: Spring Boot :: (v2.5.6)

2021-12-12 18:48:42.994 INFO 1 --- [main] com.banco.bluebank.BluebankApplication : Starting BluebankApplication vlates...

2021-12-12 18:48:42.994 INFO 1 --- [main] com.banco.bluebank.BluebankApplication : Starting BluebankApplication vlates...

2021-12-12T15:48:43.076-03:00 2021-12-12 18:48:43.069 INFO 1 --- [main] com.banco.bluebank.BluebankApplication : No active profile set, falling back...

2021-12-12T15:48:49.380-03:00 2021-12-12 18:48:49.380 INFO 1 --- [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA rep...

2021-12-12T15:48:49.778-03:00 2021-12-12 18:48:49.773 INFO 1 --- [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository s...

2021-12-12T15:48:52.380-03:00 2021-12-12 18:48:52.380 INFO 1 --- [main] trationDelegate\$BeanPostProcessorChecker : Bean 'org.springframework.security...

2021-12-12T15:48:52.391-03:00 2021-12-12 18:48:52.391 INFO 1 --- [main] trationDelegate\$BeanPostProcessorChecker : Bean 'methodSecurityMetadataSourc...

2021-12-12T15:48:54.091-03:00 2021-12-12 18:48:54.090 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8...

2021-12-12T15:48:54.177-03:00 2021-12-12 18:48:54.176 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]

Copiar

Visualização dos logs da aplicação no CloudWatch

# ELB

The screenshot shows the AWS EC2 Load Balancer configuration interface. On the left, there's a sidebar with navigation links for EC2 services like Painel EC2, EC2 Global View, and various instance-related options. The main area displays a table of existing load balancers, with the 'bluebank-lb' entry selected. A red box highlights the 'Zonas de disponibilidade' (Availability Zones) column, showing two entries: 'us-east-2a, us-east-2b'. A large red oval surrounds this entire section with the annotation: 'Um Load Balancer para cada aplicação em Zonas de disponibilidades diferentes' (One Load Balancer for each application in different availability zones).

**Criar Load Balancer** Actions

Nome	Nome do DNS	Estado	ID da VPC	Zonas de disponibilidade	Tipo	Criado em
bluebank-auth-lb	bluebank-auth-lb-296919576...	Ativo	vpc-03db30f93bd094fb	us-east-2a, us-east-2b	application	11 de dezembro d
bluebank-lb	bluebank-lb-281632386.us-e...	Ativo	vpc-03db30f93bd094fb	us-east-2a, us-east-2b	application	11 de dezembro d

**Load balancer: bluebank-lb**

Descrição **Listeners** Monitoramento Serviços integrados Tags

Ouvintes escutam solicitações de conexão usando seu protocolo e sua porta. Você pode adicionar, remover ou atualizar ouvintes e regras de ouvintes.

Para visualizar e editar atributos de ouvinte, selecione o ouvinte e escolha Editar.

**Adicionar listener** Editar Excluir

ID do listener	Política de segurança	Certificado SSL	Regras
HTTPS : 443	ELBSecurityPolicy-2016-08	Padrão: 46ff6d88-6b10-4fa7-b6c0-3b472f08af77 (ACM)	Padrão: retornando com resposta fixa 503
arn...be5883ed37f2f017		Visualizar/editar certificados	Visualizar/editar regras

**Annotations:**

- A red box highlights the 'Zonas de disponibilidade' column in the table, with a callout: 'Um Load Balancer para cada aplicação em Zonas de disponibilidades diferentes'.
- A red box highlights the 'HTTPS : 443' row in the 'Listeners' table, with a callout: 'Escutando na porta 443 (https) e cada um com seu target group de apontamento para as instâncias multi-AZ ativas'.

# CERTIFICATE

The screenshot shows the AWS Certificate Manager console with the following details:

**Top Bar:** AWS logo, Services menu, Search bar (Search for services, features, blogs, docs, and more), [Alt+S] key shortcut, and a navigation bar with Ohio selected.

**Breadcrumb:** AWS Certificate Manager > Certificados > 46ff6d88-6b10-4fa7-b6c0-3b472f08af77

**Page Title:** 46ff6d88-6b10-4fa7-b6c0-3b472f08af77

**Status do certificado:**

- Identificador: 46ff6d88-6b10-4fa7-b6c0-3b472f08af77
- ARN: arn:aws:acm:us-east-2:653929532121:certificate/46ff6d88-6b10-4fa7-b6c0-3b472f08af77
- Tipo: Emitido pela Amazon

**Callout:** A red oval highlights the status information, which is also copied into a callout box. The text in the callout box reads: "Certificado emitido e apontado para os Load Balancers das duas aplicações permitindo o uso do protocolo HTTPS".

**Status Callout Content:** Status: ✓ Emitido. Status detalhado: O certificado foi emitido em dezembro 11, 2021, 12:03:02 (UTC-03:00).

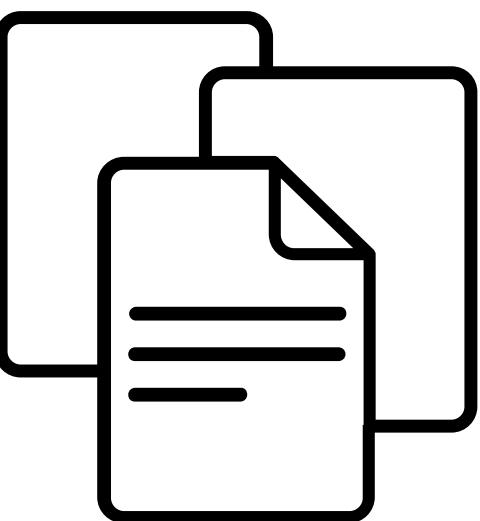
**Domínios (1):**

Domínio	Status	Status da renovação	Proprietários registrados
*.6devs.com.br	<span style="color: green;">✓</span> Êxito	-	admin@6devs.com.br, administrator@6devs.com.br, carlos@realisto.com.br, hostmaster@6devs.com.br, postmaster@6devs.com.br, webmaster@6devs.com.br

**Callout to Domains:** A red arrow points from the status information in the first callout to the status column of the domain table.

**Bottom Section:** Detalhes (Details) button.

# Por que documentar código fonte?



- 1. Facilita com futuras evoluções;**
- 2. Facilita o entendimento ao desenvolver;**
- 3. Minimiza retrabalho;**
- 4. Armazena as informações da Regra de Negócio;**
- 5. E entre outros!**



# Como documentar?



# Usa JavaDoc!

```
/**  
 *  
 * @CopyNote Entidade Correntista  
 * @version v1  
 */  
  
@Entity  
@Table(name = "correntista")  
public class Correntista implements Serializable {  
  
    private static final long serialVersionUID = 1L;  
    /**  
     * @serial Gerar ID  
     */  
    @Id  
    @GeneratedValue(strategy = GenerationType.IDENTITY)  
    @Column(name = "id_correntista")  
    private Long id;  
    /**  
     * Atributo nome  
     */  
    @Column(length = 50, nullable = false)  
    @NotBlank(message = "Nome do correntista deve ser preenchido")  
    @Size(min = 2, max = 50, message = "Nome do correntista deve ter entre 2 e 50 letras")  
    private String nome;
```



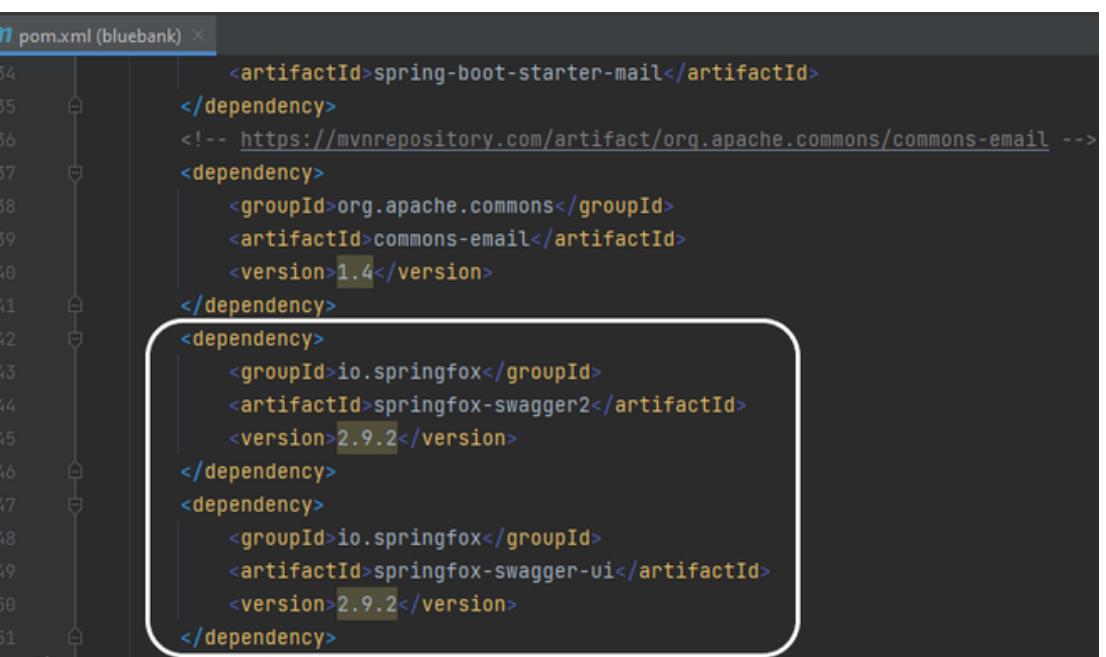
JavaDoc

JavaDoc

JavaDoc

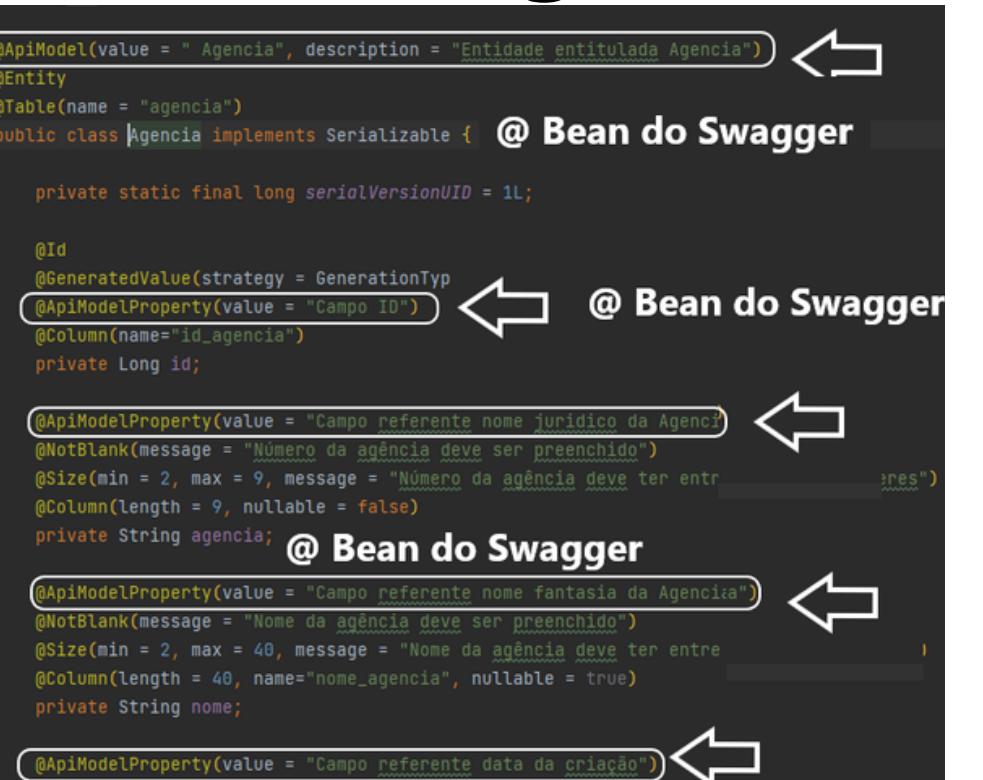
# Por que não o Swagger?

## Dependency pom.xml



```
<dependency>
    <artifactId>spring-boot-starter-mail</artifactId>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.apache.commons/commons-email -->
    <dependency>
        <groupId>org.apache.commons</groupId>
        <artifactId>commons-email</artifactId>
        <version>1.4</version>
    </dependency>
    <dependency>
        <groupId>io.springfox</groupId>
        <artifactId>springfox-swagger2</artifactId>
        <version>2.9.2</version>
    </dependency>
    <dependency>
        <groupId>io.springfox</groupId>
        <artifactId>springfox-swagger-ui</artifactId>
        <version>2.9.2</version>
    </dependency>
```

## Entidade Agencia



```
@ApiModel(value = "Agencia", description = "Entidade entitulada Agencia")
@Entity
@Table(name = "agencia")
public class Agencia implements Serializable { @ Bean do Swagger

    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @ApiModelProperty(value = "Campo ID")
    @Column(name="id_agencia")
    private Long id;

    @ApiModelProperty(value = "Campo referente nome jurídico da Agencia")
    @NotBlank(message = "Número da agência deve ser preenchido")
    @Size(min = 2, max = 9, message = "Número da agência deve ter entre")
    @Column(length = 9, nullable = false)
    private String agencia; @ Bean do Swagger

    @ApiModelProperty(value = "Campo referente nome fantasia da Agencia")
    @NotBlank(message = "Nome da agência deve ser preenchido")
    @Size(min = 2, max = 40, message = "Nome da agência deve ter entre")
    @Column(length = 40, name="nome_agencia", nullable = true)
    private String nome;

    @ApiModelProperty(value = "Campo referente data da criação")
```

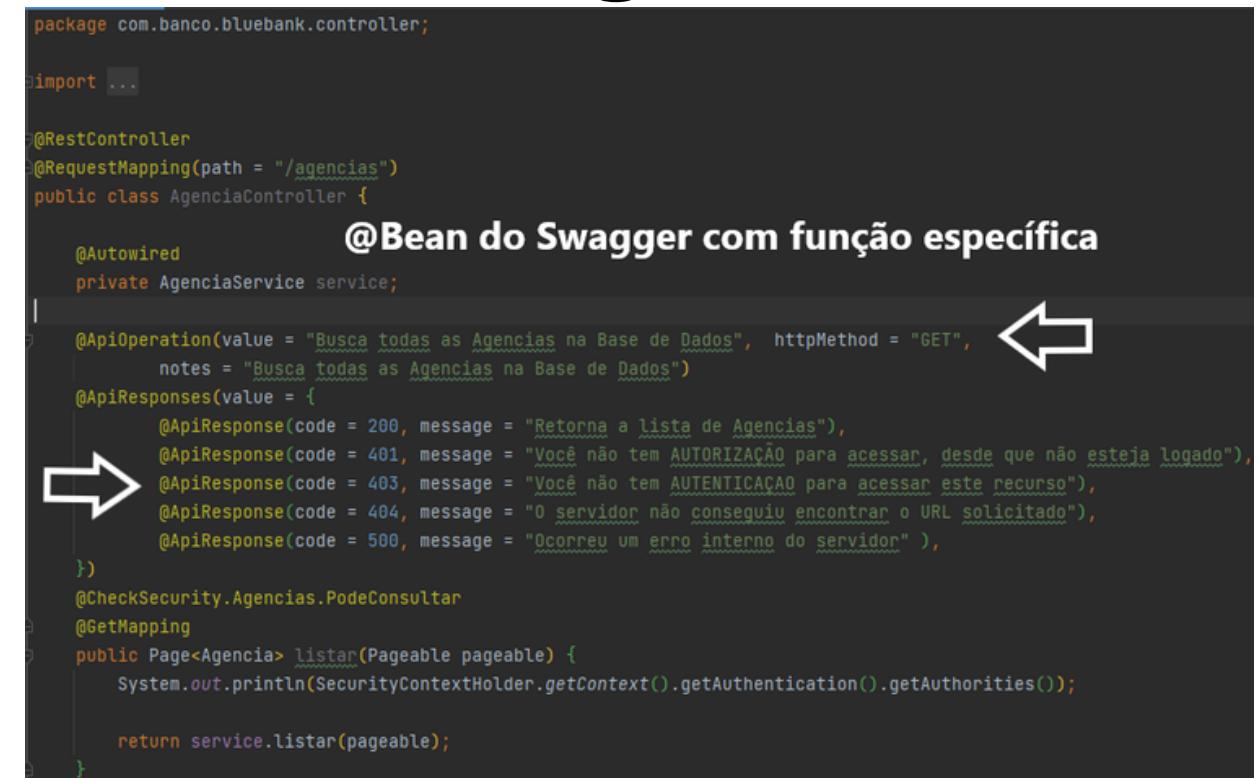
## Configurações do Swagger



```
@Bean
public Docket apiDocket() {
    return new Docket(DocumentationType.SWAGGER_2)
        .select()
        .apis(RequestHandlerSelectors.basePackage("com.banco.bluebank"))
        .paths(PathSelectors.any())
        .build()
        .ignoredParameterTypes(OAuth2ResourceServerConfigurer.JwtConfigurer.class).globalOperationParameters(
            Arrays.asList(new ParameterBuilder().name("Authorization")
                .description("Header para token JWT")
                .modelRef(new ModelRef(type: "string").parameterType("header")
                    .required(false).build()).select().build().securitySchemes(
                        Arrays.asList(new ApiKey(name: "Token Access",
                            HttpHeaders.AUTHORIZATION, In.HEADER.name())))
            ).Docket
        .apiInfo(moreInfo());
}

private List<ResponseMessage> responseMessageForGET() {
    return new ArrayList<ResponseMessage>() {
        add(new ResponseMessageBuilder()
            .code(500)
            .message("500 message")
            .responseModel(new ModelRef(type: "Error"))
            .build());
        add(new ResponseMessageBuilder()
            .code(403)
            .message("403 message")
            .responseModel(new ModelRef(type: "Error"))
            .build());
    };
}
```

## Controller -> AgenciaController



```
package com.banco.bluebank.controller;

import ...

@RestController
@RequestMapping(path = "/agencias")
public class AgenciaController {

    @Autowired
    private AgenciaService service;

    @ApiOperation(value = "Busca todas as Agencias na Base de Dados", httpMethod = "GET",
        notes = "Busca todas as Agencias na Base de Dados")
    @ApiResponses(value = {
        @ApiResponse(code = 200, message = "Retorna a lista de Agencias"),
        @ApiResponse(code = 401, message = "Você não tem AUTORIZAÇÃO para acessar, desde que não esteja logado"),
        @ApiResponse(code = 403, message = "Você não tem AUTENTICAÇÃO para acessar este recurso"),
        @ApiResponse(code = 404, message = "O servidor não conseguiu encontrar o URL solicitado"),
        @ApiResponse(code = 500, message = "Ocorreu um erro interno do servidor")
    })
    @CheckSecurity.Agencias.PodeConsultar
    @GetMapping
    public Page<Agencia> listar(Pageable pageable) {
        System.out.println(SecurityContextHolder.getContext().getAuthentication().getAuthorities());
        return service.listar(pageable);
    }
}
```

# Quais são os resultados?

## Cliente documentada *Interface* de fácil entendimento

Banco BlueBank é projeto final referente ao de treinamento com BACK-END em Java ofertado pelo Banco Pan cuja Parceria da Gama Academy!

API and SDK Documentation

Version: Essa API está na primeira Versão

Essa API faz todas transações bancária conforme solicitação da Gama cujos Professores: Jenifer Plácido, Ana Verônica cuja Staff à mentoria e Jonathan Ferreira cujo Staff referente ao assunto: SPRING BOOT

**AgenciaController**

**atualizarUsingPUT**

Faz atualização global da Agencia específico pelo ID

Atualiza agencia via ID

**PUT**

/agencias/{id}

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

curl -X PUT "https://localhost:8080/agencias/{id}"

**Parameters**

Path parameters

Name	Description
id*	Long (int64) id Required

Header parameters

Name	Description
Authorization	String Lembre-se de adicionar HTTP

swagger

Select a spec: default

Banco BlueBank é projeto final referente ao de treinamento em BACK-END com Java ofertado pelo Banco Pan cuja Parceria da Gama Academy!

[ Base URL: https://bluebank.6devs.com.br/ ]  
https://bluebank.6devs.com.br/v2/api-docs

Essa API faz todas transações bancária conforme solicitação da Gama cujos Professores: Jenifer Plácido, Ana Verônica cuja Staff à mentoria e Jonathan Ferreira cujo Staff referente ao assunto: SPRING BOOT

Squad 6Devs - Website  
Send email to Squad 6Devs  
Apache License Version 2.0

**Interface amigável a todos, ou seja; NÃO TECNICO & TECNICO!**

Authorize

agencia-controller Agencia Controller

basic-error-controller Basic Error Controller

conta-controller Conta Controller

correntista-controller Correntista Controller

health-check-controller Health Check Controller

host-check-controller Host Check Controller

movimentacao-controller Movimentacao Controller

Models



# Obrigado!

Sinta-se à vontade para nos abordar se tiver alguma dúvida.