

Desafio DevOps

Grupo:

- Gabriela May Canarin – RM 554853
- Guilherme Marcionilo Pedroso do Rosario Silva – RM 557115
- Guilherme Menoti Merli – RM 556144

1. Introdução

O **ESGames API** é uma aplicação corporativa que promove **sustentabilidade e engajamento ambiental** através da **gamificação**. O sistema incentiva colaboradores de grandes empresas a realizarem ações sustentáveis, acumulando pontos, selos e recompensas conforme seus resultados.

Este projeto foi desenvolvido na **FIAP**, como parte do módulo **DevOps e Microsserviços**, e integra práticas completas de **CI/CD, containerização, infraestrutura como código e publicação automatizada** via GitHub Actions e Docker Hub.

2. Arquitetura da Aplicação

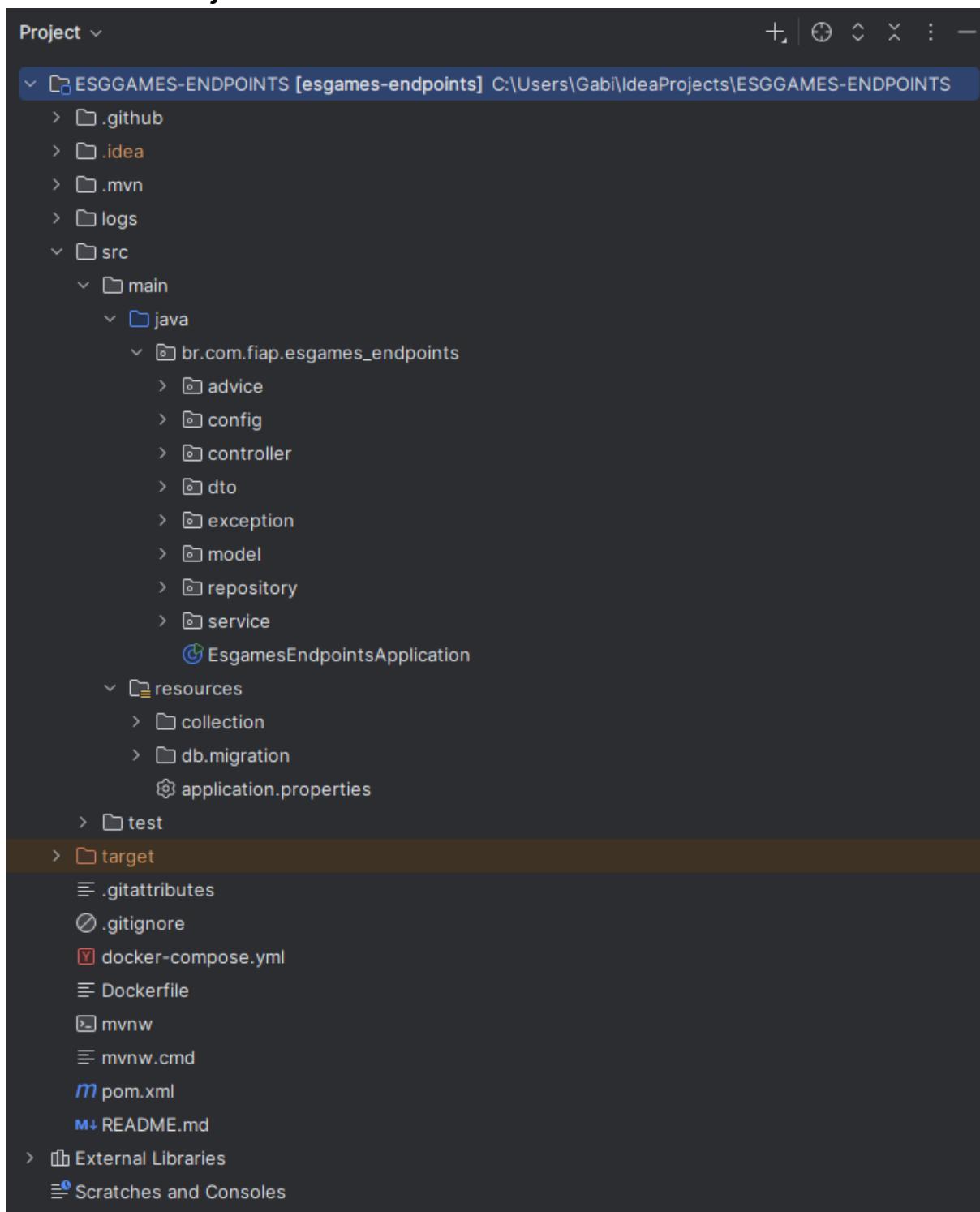
A aplicação foi desenvolvida em **Spring Boot 3.3** e utiliza **MongoDB Atlas** como banco de dados NoSQL.

A estrutura do projeto segue o padrão **MVC (Model-View-Controller)**, organizada da seguinte forma:

esgames-endpoints/

```
└── src/
    ├── main/java/br/com/fiap/esgames_endpoints/
    │   ├── controller/ → Endpoints REST (Missões, Selos, Usuários, Ranking)
    │   ├── service/ → Regras de negócio
    │   ├── repository/ → Persistência (MongoRepository)
    │   └── model/ → Modelos de domínio
    └── resources/
        └── application.properties
└── Dockerfile
└── docker-compose.yml
└── .github/workflows/github-actions-esgame.yml
```

Estrutura do Projeto



The screenshot shows the Project Structure dialog in IntelliJ IDEA. The project root is `ESGGAMES-ENDPOINTS [esgames-endpoints]` located at `C:\Users\Gabi\IdeaProjects\ESGGAMES-ENDPOINTS`. The tree view displays the following structure:

- `.github`
- `.idea`
- `.mvn`
- `logs`
- `src`
 - `main`
 - `java`
 - `br.com.fiap.esgames_endpoints`
 - `advice`
 - `config`
 - `controller`
 - `dto`
 - `exception`
 - `model`
 - `repository`
 - `service`
 - `EsgamesEndpointsApplication` (highlighted)
 - `resources`
 - `collection`
 - `db.migration`
 - `application.properties`
 - `test`
 - `target`
 - `.gitattributes`
 - `.gitignore`
 - `docker-compose.yml`
 - `Dockerfile`
 - `mvnw`
 - `mvnw.cmd`
 - `pom.xml`
 - `README.md`
 - `External Libraries`
 - `Scratches and Consoles`

3. Banco de Dados - MongoDB Atlas

O projeto utiliza o **MongoDB Atlas**, um banco de dados escalável em nuvem. A conexão está configurada no application.properties:

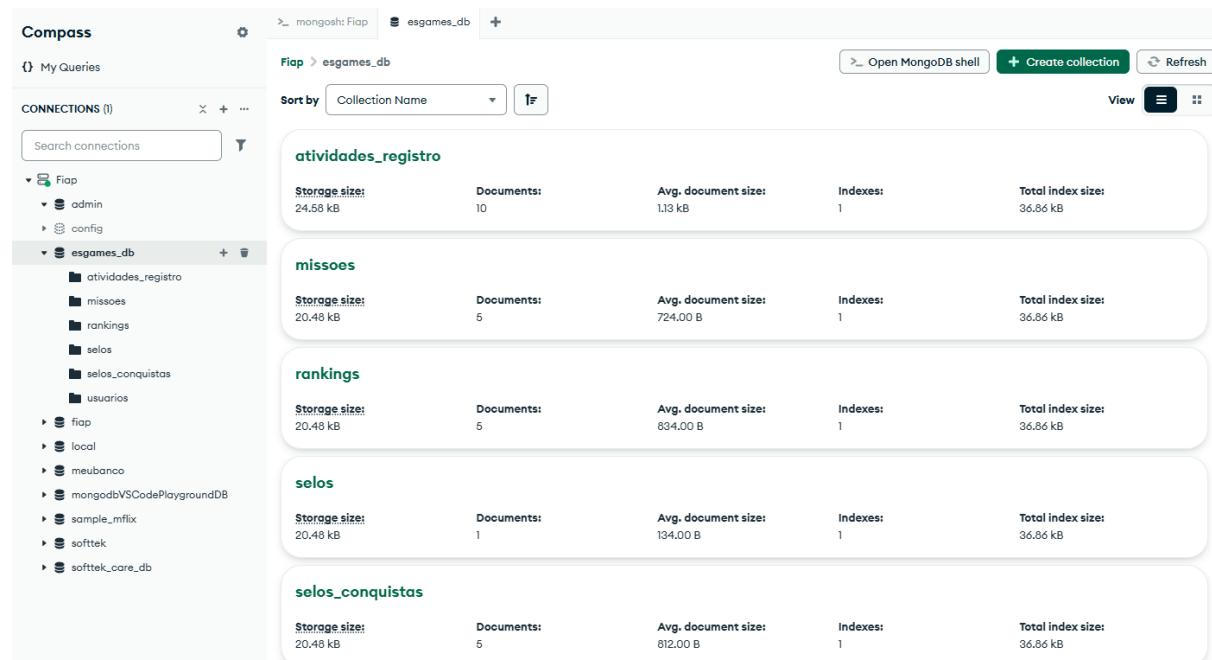
```
spring.data.mongodb.uri=mongodb+srv://gabimay:181114@cluster0.tuwr99s.mongodb.net/esgames_db?retryWrites=true&w=majority
```

```
spring.data.mongodb.database=esgames_db
```

As principais coleções são:

Coleção	Descrição
usuarios	Dados de autenticação via JWT
missoes	Missões de sustentabilidade
selos	Selos conquistados por desempenho
registros_atividade	Registro de ações dos colaboradores
recompensas	Recompensas trocadas por pontos

MongoDB Compass



The screenshot shows the MongoDB Compass interface connected to the 'mongosh: Fiap' cluster and the 'esgames_db' database. The left sidebar lists connections: Fiap, admin, config, and esgames_db. The esgames_db connection is expanded, showing collections: atividades_registro, missoes, rankings, selos, and selos_conquistas. Each collection card provides storage size, document count, average document size, index count, and total index size.

Collection	Storage size	Documents	Avg. document size	Indexes	Total index size
atividades_registro	24.58 kB	10	1.13 kB	1	36.86 kB
missoes	20.48 kB	5	724.00 B	1	36.86 kB
rankings	20.48 kB	5	834.00 B	1	36.86 kB
selos	20.48 kB	1	134.00 B	1	36.86 kB
selos_conquistas	20.48 kB	5	812.00 B	1	36.86 kB

4. Containerização com Docker e Docker Compose

A aplicação foi totalmente containerizada, permitindo execução em qualquer ambiente com **Docker** instalado.

Dockerfile

```
FROM maven:3.9.9-eclipse-temurin-21 AS build
WORKDIR /app
COPY pom.xml .
RUN mvn dependency:go-offline
COPY src ./src
RUN mvn clean package -DskipTests
```

```
FROM eclipse-temurin:21-jdk-alpine
WORKDIR /app
COPY --from=build /app/target/esgames-endpoints-0.0.2-SNAPSHOT.jar app.jar
EXPOSE 8080
ENTRYPOINT ["java", "-jar", "app.jar"]
```

docker-compose.yml

```
version: "3.9"
services:
  mongo:
    image: mongo:7
    container_name: mongo
    restart: always
    ports:
      - "27017:27017"
    volumes:
      - mongo_data:/data/db
networks:
```

- esgames-network

esgames-api:

build:

context: .

dockerfile: Dockerfile

container_name: esgames-endpoints

depends_on:

- mongo

ports:

- "8080:8080"

environment:

SPRING_PROFILES_ACTIVE: prod

SPRING_DATA_MONGODB_URI:

mongodb+srv://gabimay:181114@cluster0.tuwr99s.mongodb.net/esgames_db

JWT_SECRET: esgamesSecretKey2025!@#

networks:

- esgames-network

volumes:

mongo_data:

networks:

esgames-network:

driver: bridge



Docker Desktop

The screenshot shows the Docker desktop application interface. On the left, a sidebar lists various sections: Ask Gordon (Beta), Containers, Images, Volumes, Kubernetes, Builds, Models (with one entry for esgames-endpoints), MCP Toolkit (Beta), Docker Hub, Docker Scout, and Extensions. The main area displays two containers running:

- mongo** (version 2.0.27_2017.27.2017) with a green status icon.
- esgames-endpoints** (version 8.0.0.800) with a blue status icon.

Below the containers, a log viewer shows the following log entries for the esgames-endpoints container:

```
2025-10-06T08:33:24.797+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759718804, "ts_usec":797581, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 139, snapshot max: 139 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:33:24.797+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759718804, "ts_usec":826568, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 140, snapshot max: 140 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:33:24.853+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759718924, "ts_usec":853371, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 141, snapshot max: 141 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:36:24.826+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759718924, "ts_usec":882188, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 142, snapshot max: 142 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:37:24.916+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759718944, "ts_usec":916341, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 144, snapshot max: 144 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:38:24.947+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759711194, "ts_usec":947542, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 144, snapshot max: 144 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:39:24.972+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759711194, "ts_usec":973322, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 145, snapshot max: 145 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}], [{"message": "2025-10-06T08:40:24.996+00:00,"":{ "c":"WTCHOPT", "id":22430, "ctx":"Checkpointer", "ns":"$tredTiger message", "attr":{}}, {"ts_sec":1759711224, "ts_usec":996581, "thread":1, "session_name":"_WT_SESSION.checkpoint", "category":"_WT_VERB_CHECKPOINT_PROGRESS", "cat_epry_id":6, "verbose_level":1, "verbose_level_id":1, "msg":"saving checkpoint snapshot mtn: 146, snapshot max: 146 snapshot count: 0, oldest timestamp: (0, 0), meta checkpoint timestamp: (0, 0) base write gen: 742"}]}]
```

5. Pipeline CI/CD - GitHub Actions

A automação do ciclo de vida da aplicação foi configurada utilizando **GitHub Actions** e **Docker Hub**.

Cada push na branch main executa o workflow que:

1. Realiza o build e testes com Maven;
2. Efetua login seguro no Docker Hub;
3. Constrói e publica a imagem automaticamente.

Pipeline YAML - `github-actions-esgame.yml`

```
name: 🚀 CI/CD - ESGames API
```

```
on:
```

```
push:
```

```
  branches: [ "main" ]
```

```
pull_request:
```

```
  branches: [ "main" ]
```

```
jobs:
```

```
build-and-push:
```

```
  runs-on: ubuntu-latest
```

```
  steps:
```

```
    - name: 📁 Checkout repository
```

```
      uses: actions/checkout@v4
```

```
    - name: 🎨 Setup JDK 21
```

```
      uses: actions/setup-java@v4
```

```
      with:
```

```
        java-version: '21'
```

distribution: 'temurin'

- name: 🖌 Build e Testes

run: mvn -B clean verify

- name: 🔒 Login no Docker Hub

uses: docker/login-action@v3

with:

username: \${{ secrets.DOCKERHUB_USERNAME }}

password: \${{ secrets.DOCKERHUB_TOKEN }}

- name: 🛠 Build & Push Docker image

uses: docker/build-push-action@v5

with:

context: .

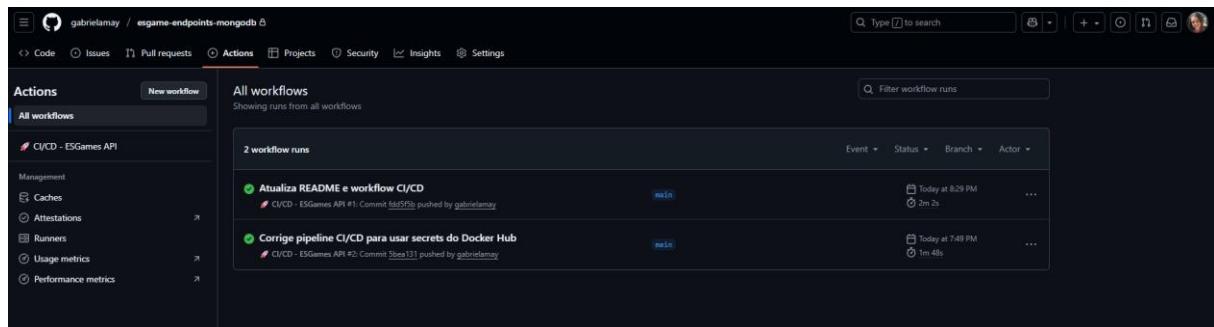
push: true

tags: |

\${{ secrets.DOCKERHUB_USERNAME }}/esgames-api:latest

\${{ secrets.DOCKERHUB_USERNAME }}/esgames-api:\${{ github.run_number }}

GitHub Actions



The screenshot shows the GitHub Actions interface for the repository 'gabrielamay / esgame-endpoints-mongodb'. The 'Actions' tab is selected. On the left, there's a sidebar with 'Management' sections for Caches, Attestations, Runners, Usage metrics, and Performance metrics. The main area displays 'All workflows' with a search bar and filter for workflow runs. Two runs are listed:

- Atualiza README e workflow CI/CD**: Triggered by a CI/CD event, pushed by gabrielamay, status: success, run time: Today at 8:29 PM, duration: 2m 2s.
- Corrige pipeline CI/CD para usar secrets do Docker Hub**: Triggered by a CI/CD event, pushed by gabrielamay, status: success, run time: Today at 7:49 PM, duration: 1m 48s.

Actions New workflow

All workflows

Build and deploy Java app to Azure Web App - esgame-prod

main_esgame-prod.xml

18 workflow runs

This workflow has a `workflow_dispatch` event trigger.

Run workflow

Event	Status	Branch	Actor
27 minutes ago	Success	main	Guilherme Marconilo
33 minutes ago	Success	main	Guilherme Marconilo
43 minutes ago	Success	main	Guilherme Marconilo

Actions New workflow

All workflows

Build and deploy Java app to Azure Web App - esgame-staging

main_esgame-staging.xml

20 workflow runs

This workflow has a `workflow_dispatch` event trigger.

Run workflow

Event	Status	Branch	Actor
27 minutes ago	Success	main	Guilherme Marconilo
34 minutes ago	Success	main	Guilherme Marconilo
43 minutes ago	Success	main	Guilherme Marconilo

Docker Hub

[Repositories / esgames-api / Tags](#)

gabrielamay/esgames-api

Last pushed about 1 hour ago • Repository size: 232 MB

Add a description

Add a category

General **Tags** Image Management BETA Collaborators Webhooks Settings

Sort by Newest Filter tags Delete

TAG **1**

Last pushed about 1 hour by [gabrielamay](#)

Digest	OS/ARCH	Last pull	Compressed size
b5a05668da56	linux/amd64	less than 1 day	203.24 MB

TAG **latest**

Last pushed about 1 hour by [gabrielamay](#)

Digest	OS/ARCH	Last pull	Compressed size
b5a05668da56	linux/amd64	less than 1 day	203.24 MB

6. Deploy no Azure App Service (Staging e Produção)

A aplicação foi implantada em dois ambientes Azure App Service, automatizando o deploy via GitHub Actions.

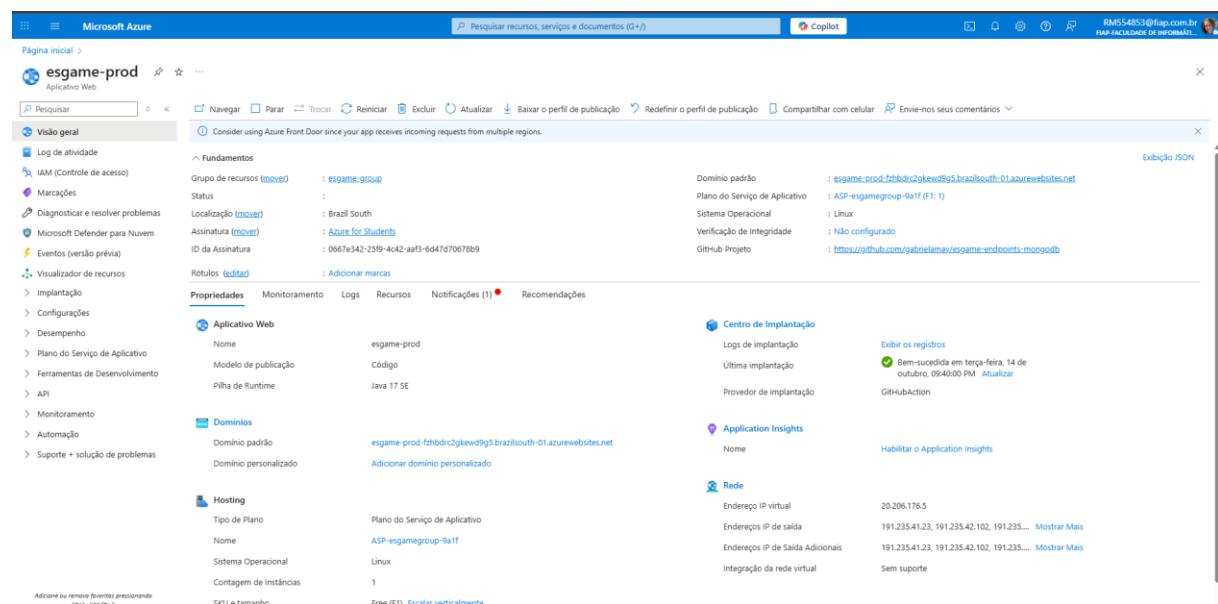
◆ Ambiente de Produção

- **Nome do App:** esgame-prod
- **Região:** Brazil South
- **Stack:** Java 17 SE
- **Repositório:** GitHub conectado via GitHub Actions
- **URL:**

 <https://esgame-prod-fzhbdrc2gkewd9g5.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/>

-  <https://esgame-staging-esbjf5fdaxf0enf8.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/>

💡 Print 6 – Azure App Service “esgame-prod” (status: Em execução)



The screenshot shows the Microsoft Azure portal interface for the 'esgame-prod' application service. The left sidebar lists navigation options: Página inicial, esgame-prod (selected), Visão geral, Log de atividade, IAM (Controle de acesso), Marcações, Diagnosticar e resolver problemas, Microsoft Defender para Nuvem, Eventos (versão prévia), Visualizador de recursos, Implantação, Configurações, Desempenho, Plano do Serviço de Aplicativo, Ferramentas de Desenvolvimento, API, Monitoramento, Automação, Suporte + solução de problemas, and Adicione ou remova favoritos pressionando Shift + Entrar.

The main content area shows the 'Visão geral' (General) blade for the 'esgame-prod' app service. It includes sections for Fundamentos (Basics), Aplicativo Web (Web App), Domínios (Domains), and Hosting. The 'Aplicativo Web' section shows the app's name as 'esgame-prod', deployment model as 'Código' (Code), and runtime stack as 'Java 17 SE'. The 'Domínios' section shows the default domain as 'esgame-prod-fzhbdrc2gkewd9g5.brazilsouth-01.azurewebsites.net' and a custom domain 'Dominio personalizado' (Custom domain) as 'Adicionar domínio personalizado'. The 'Hosting' section shows the plan as 'Plano do Serviço de Aplicativo' (App Service Plan) 'ASP-esgamegroup-9af1'.

On the right side, there are cards for 'Centro de Implantação' (Deployment Center), 'Application Insights' (Application Insights), and 'Rede' (Networking). The 'Centro de Implantação' card shows the last deployment was successful on Friday, October 14, at 09:40:00 PM. The 'Application Insights' card shows the status as 'Habilitar o Application Insights' (Enable Application Insights). The 'Rede' card shows network statistics: IP virtual (20.206.176.5), IP de saída (191.235.41.23, 191.235.42.102, 191.235...), and IP de Saída Adicionais (191.235.41.23, 191.235.42.102, 191.235...).

esgame-prod-fzhbdr2gkewd9g5.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/Missões/listarMissões

Parameters

No parameters

Responses

Curl

```
curl -X 'GET' \
  'https://esgame-prod-fzhbdr2gkewd9g5.brazilsouth-01.azurewebsites.net/api/missões' \
  -H 'accept: application/json'
```

Request URL

<https://esgame-prod-fzhbdr2gkewd9g5.brazilsouth-01.azurewebsites.net/api/missões>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "nome": "Reciclagem de Eletrônicos", "descricao": "Coleta e descarte adequado de equipamentos eletrônicos obsoletos" }, { "nome": "Economia de Energia - Ar Condicionado", "descricao": "Configurar temperatura ideal do ar condicionado (23°C) durante 8 horas" }, { "nome": "Transporte Sustentável", "descricao": "Organizar carona solidária ou usar transporte público" }, { "nome": "Redução de Papel", "descricao": "Digitalizar documentos e eliminar impressões desnecessárias" }, { "nome": "Horta Corporativa", "descricao": "Plantar e cuidar de mudas na horta da empresa" }]</pre> <p>Download</p> <p>Response headers</p> <pre>content-type: application/json date: Wed,15 Oct 2025 00:42:26 GMT transfer-encoding: chunked</pre>

esgame-prod-fzhbdr2gkewd9g5.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/Missões/listarMissões

Responses

Curl

```
curl -X 'GET' \
  'https://esgame-prod-fzhbdr2gkewd9g5.brazilsouth-01.azurewebsites.net/api/missões' \
  -H 'accept: application/json'
```

Request URL

<https://esgame-prod-fzhbdr2gkewd9g5.brazilsouth-01.azurewebsites.net/api/missões>

Server response

Code	Details						
200	<p>Response body</p> <pre>[{ "nome": "Reciclagem de Eletrônicos", "descricao": "Coleta e descarte adequado de equipamentos eletrônicos obsoletos" }, { "nome": "Economia de Energia - Ar Condicionado", "descricao": "Configurar temperatura ideal do ar condicionado (23°C) durante 8 horas" }, { "nome": "Transporte Sustentável", "descricao": "Organizar carona solidária ou usar transporte público" }, { "nome": "Redução de Papel", "descricao": "Digitalizar documentos e eliminar impressões desnecessárias" }, { "nome": "Horta Corporativa", "descricao": "Plantar e cuidar de mudas na horta da empresa" }]</pre> <p>Download</p> <p>Response headers</p> <pre>content-type: application/json date: Wed,15 Oct 2025 00:42:26 GMT transfer-encoding: chunked</pre> <p>Responses</p> <table border="1"> <thead> <tr> <th>Code</th> <th>Description</th> <th>Links</th> </tr> </thead> <tbody> <tr> <td>200</td> <td></td> <td></td> </tr> </tbody> </table>	Code	Description	Links	200		
Code	Description	Links					
200							

◆ Ambiente de Staging

- Nome do App: esgame-staging
- Região: Brazil South
- Stack: Java 17 SE
- Pipeline: GitHub Actions
- URL:
🔗 <https://esgame-staging-esbjf5fdaxf0enf8.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/>

🖨 Print 7 – Azure App Service “esgame-staging” (status: Em execução)

portal.azure.com/#@fiap.com.br/resource/subscriptions/0667e342-25f9-4c42-aa53-6d47d7067b69/resourceGroups/esgame-group/providers/Microsoft.Web/sites/esgame-staging/appServices

esgame-staging Aplicativo Web

Página inicial >

Visão geral

- Log de atividade
- IAM (Controle de acesso)
- Marcações
- Diagnosticar e resolver problemas
- Microsoft Defender para Nuvem
- Eventos (versão prévia)
- Visualizador de recursos
- Implantação
- Configurações
- Desempenho
- Plano do Serviço de Aplicativo
- Ferramentas de Desenvolvimento
- API
- Monitoramento
- Automação
- Supo + solução de problemas

Fundamentos

Grupo de recursos (moved)	: esgame-group
Status	: Em execução
Localização (moved)	: Brazil South
Assinatura (moved)	: Azure for Students
ID da Assinatura	: 0667e342-25f9-4c42-aa53-6d47d7067b69
Rótulos (editar)	: Adicionar marcas

Propriedades

Aplicativo Web	Centro de Implantação
Nome	: esgame-staging
Modelo de publicação	Código
Pilha de Runtime	: Java 17 SE
Dominios	Application Insights
Domínio padrão	: esgame-staging-estb5fdaxf0enf8.brazilsouth-01.azurewebsites.net
Domínio personalizado	: Adicionar domínio personalizado
Hosting	Rede
Tipo de Plano	Plano do Serviço de Aplicativo
Nome	: ASP-esgamegroup-9a1f
Sistema Operacional	: Linux
Contagem de Instâncias	: 1
SKU e tamanho	: Free (F1) Escalar verticalmente

Exibição JSON

<https://github.com/gabrielaev/esgame-endpoints-mongodb>

<https://esgame-staging-estb5fdaxf0enf8.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/>

Swagger /api/api-docs **Explore**

ESGames API 0.0.2 OAS 3.0

API RESTful do projeto ESGames - Sistema de gamificação para práticas ESG

Contact FIAP
Apache 2.0

Servers
[https://esgame-staging-estb5fdaxf0enf8.brazilsouth-01.azurewebsites.net/api - Servidor de Staging](https://esgame-staging-estb5fdaxf0enf8.brazilsouth-01.azurewebsites.net/api)

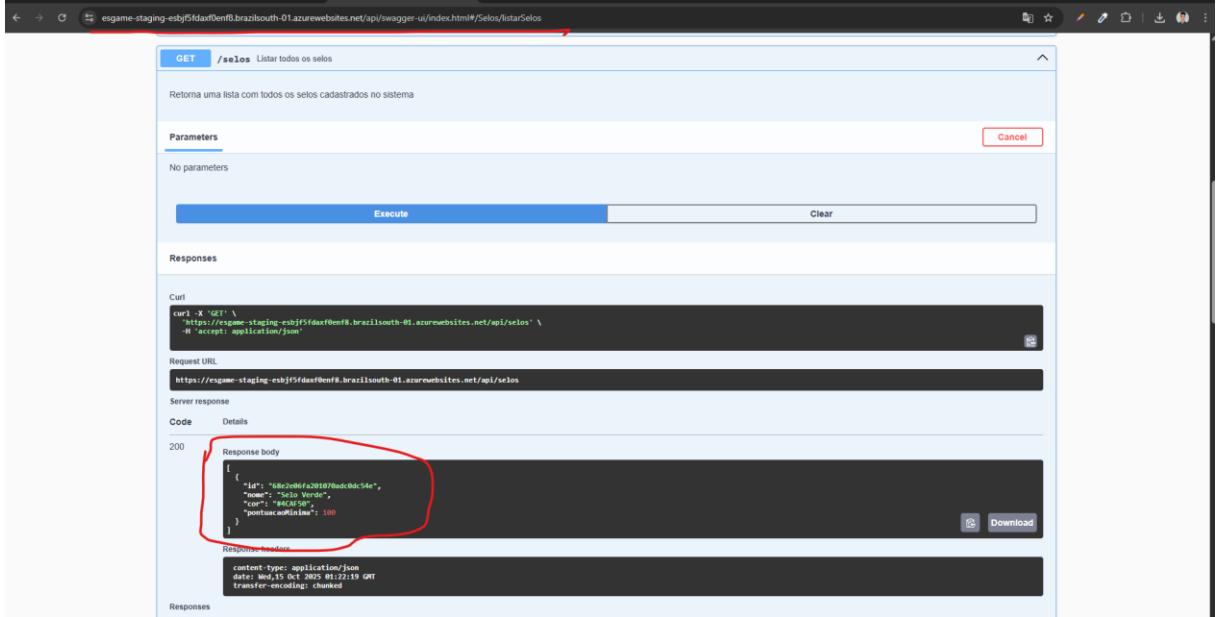
Missões Endpoints para gerenciamento de missões do sistema ESGames

Ranking Endpoints para consulta e gerenciamento de rankings do sistema ESGames

GET /ranking/setor	Listar ranking por setor
GET /ranking/individual	Listar ranking individual
POST /ranking/registro-atividade	Registrar atividade

Selos Endpoints para gerenciamento de selos e conquistas do sistema ESGames

DELETE /selos/{id}	Deletar selo
GET /selos/{id}	Buscar selo por ID



The screenshot shows the Swagger UI interface for a REST API. The URL in the address bar is `esgame-staging-esbj5fdax0enf8.brazilsouth-01.azurewebsites.net/api/swagger-ui/index.html#/Selos/listarSelos`. The main panel displays a GET request for the endpoint `/selos`, which lists all registered colors. The response body is highlighted with a red box and contains the following JSON data:

```
[{"id": "6ac3ebfa3b1078dc0d54e", "name": "Selo Verde", "color": "#00A358", "percentage": 100}]
```

◆ Workflow de Deploy no Azure

```
name: Build and deploy JAR app to Azure Web App - esgame-prod
```

```
on:
```

```
  push:
```

```
    branches:
```

```
      - main
```

```
  workflow_dispatch:
```

```
jobs:
```

```
  build:
```

```
    runs-on: ubuntu-latest
```

```
    steps:
```

```
      - uses: actions/checkout@v4
```

```
      - uses: actions/setup-java@v4
```

```
        with:
```

```
          java-version: '17'
```

```
          distribution: 'microsoft'
```

```
        - run: mvn -B clean package spring-boot:repackage -DskipTests -Djacoco.skip=true
```

```
        - uses: actions/upload-artifact@v4
```

```
          with:
```

```
            name: java-app
```

```
            path: '${{ github.workspace }}/target/*.jar'
```

```
  deploy:
```

```
    runs-on: ubuntu-latest
```

```
    needs: build
```

```
    steps:
```

```
      - uses: actions/download-artifact@v4
```

with:

name: java-app

- **uses: azure/login@v2**

with:

client-id: \${{ secrets.AZUREAPPSERVICE_CLIENTID }}

tenant-id: \${{ secrets.AZUREAPPSERVICE_TENANTID }}

subscription-id: \${{ secrets.AZUREAPPSERVICE_SUBSCRIPTIONID }}

- **uses: azure/webapps-deploy@v3**

with:

app-name: 'esgame-prod'

slot-name: 'Production'

package: '*.jar'

💡 **Print 8 – Log Stream do Azure com aplicação iniciando (“Started EsgamesEndpointsApplication...”)**

💡 **Print 9 – Endpoint /api/selos acessado com sucesso (HTTP 200 OK)**

7. Testes e Execução Local

A aplicação pode ser executada localmente com:

- mvn spring-boot:run

ou via **Docker Compose**:

- docker-compose up

A aplicação estará disponível em:

👉 <http://localhost:8080/api>

8. Tecnologias Utilizadas

Categoria	Ferramenta
Linguagem	Java 17
Framework	Spring Boot 3.3
Banco de Dados	MongoDB Atlas
Containerização	Docker + Docker Compose
CI/CD	GitHub Actions
Deploy	Azure App Service
Monitoramento	Log Stream
Build	Maven
Testes	JUnit + Mockito
Autenticação	JWT

9. Conclusão

O projeto **ESGames API** consolida um pipeline completo de **DevOps**, com integração contínua, entrega automatizada e infraestrutura em nuvem. A solução garante **escalabilidade, automação total e disponibilidade contínua**, cumprindo os requisitos de um ambiente corporativo real.

O uso combinado de **Docker**, **Github Actions** e **Azure App Service** permitiu criar um fluxo profissional de publicação, com versionamento, build automatizado, testes, e deploy em ambientes de **staging** e **produção**.

10. Checklist de Entrega

Item	OK
Projeto compactado em .ZIP com estrutura organizada	<input checked="" type="checkbox"/>
Dockerfile funcional	<input checked="" type="checkbox"/>
docker-compose.yml ou arquivos Kubernetes	<input checked="" type="checkbox"/>
Pipeline com etapas de build, teste e deploy (GitHub Actions + Azure)	<input checked="" type="checkbox"/>
README.md com instruções e prints	<input checked="" type="checkbox"/>
Documentação técnica com evidências (PDF)	<input checked="" type="checkbox"/>
Deploy realizado nos ambientes staging e produção	<input checked="" type="checkbox"/>

 FIAP - Escola de Negócios, Tecnologia e Inovação
Análise e Desenvolvimento de Sistemas – Fase 6 (DevOps e Microsserviços)
Outubro / 2025