

# FIAP GRADUAÇÃO

# SISTEMAS DE INFORMAÇÃO

MICROSERVICE AND WEB ENGINEERING

PROF. JOSÉ ROMUALDO

# Global Solution

- Introdução
- Dados
- Aplicação Backend
- Aplicação Frontend
- Testes
- Publicação

Os **Objetivos de Desenvolvimento Sustentável (ODS)** representam um chamado universal para a adoção de medidas a fim de erradicar a pobreza, proteger o meio ambiente e assegurar que todas as pessoas desfrutem de paz e prosperidade. Estabelecidos em 2015 pela Assembleia Geral das Nações Unidas e integrados à Agenda 2030, os ODS são um plano de ação composto por **17 objetivos interconectados**, que incluem **169 metas** voltadas para as dimensões **econômica, social e ambiental** do desenvolvimento sustentável.

Dentro desse quadro de objetivos globais, o ODS 3, que visa assegurar uma vida saudável e promover o bem-estar para todas e todos, em todas as idades, destaca-se como uma dimensão fundamental para a construção de sociedades resilientes e inclusivas. Sua importância transcende os cuidados médicos, abrangendo a prevenção de doenças, o acesso a serviços de saúde de qualidade e a promoção de estilos de vida saudáveis.

### Dados

Considerando os dados disponíveis nos datasets do repositório, projete e execute uma estrutura de banco de dados para armazenar e servir os dados para aplicação.

**Os diagramas e scripts deverão ser versionados no repositório da GS e prints dos diagramas devem ser adicionados no arquivo readme do projeto.**

Os datasets com dados de indicadores do Objetivo 3, Saúde e Bem-Estar, estão disponíveis, em arquivos CSV (Comma Separated Values), no repositório GIT abaixo:

<https://github.com/josercf/ods3-datasets>

Você deverá utilizar os dados contidos no dataset deste repositório para realizar os exercícios da Global Solution.

## Global Solution - Dados

### 1) Dados

- a) DER:** Crie um Diagrama Entidade relacionamento
- b) MER:** Crie o Modelo Entidade Relacionamento
- c) Modelo Físico:** Implementação do modelo físico de banco de dados (utilizar SGBD relacional como Oracle, SQL Server, My SQL ou Postgress)
- d) Ingestão de dados:** Persistir os dados disponíveis nos datasets no banco de dados criado.



## Global Solution – Aplicação Backend

Projete e implemente uma aplicação usando Spring Boot (ou outra tecnologia como Node.js, C#, ou Python com Flask) para disponibilizar os dados por meio de endpoints **RESTful**.

## 1) Endpoint GET `/objetivos`

Este endpoint deverá retornar uma lista dos Objetivos ODS 3, com a estrutura ao lado:

```
{
  "id": "3.1",
  "objetivos": {
    "global": "Até 2030, reduzir a taxa de mortalidade ...",
    "brasil": "Até 2030, reduzir a razão de ..."
  },
  "indicadores": [
    {
      "id": "3.1.1",
      "descricao": "Razão de mortalidade materna."
    },
    {
      "id": "3.1.2",
      "descricao": "Proporção de nascimentos ..."
    }
  ]
}
```

### 2) Endpoint GET `/indicador/{id}`

Este endpoint deverá retornar os dados do indicador pesquisado, conforme sugestão ao lado:

Lembre-se de que os datasets podem ter informações distintas, modificando o retorno

```
[
  {
    "ano": 2007,
    "consumo": 9.7
  },
  {
    "ano": 2008,
    "consumo": 10.2
  }
]
```

## Global Solution – Aplicação Backend

**3) Codificação:** Implemente os endpoints no passo anterior.

- Disponibilize a aplicação frontend na pasta **/backend**, no repositório do projeto.
- No arquivo readme, crie uma seção com instruções para execução do backend.
- No arquivo readme, crie uma seção com prints do backend sendo executado no postman.

Projete e implemente uma aplicação frontend utilizando JSP, Thymeleaf, angular, react ou outro framework de frontend de sua preferência. O frontend deverá consumir os endpoints criados na etapa anterior e exibir os dados para o usuário de maneira clara e organizada.

## I Global Solution – Aplicação Frontend

**1) Wireframe:** Faça o protótipo das telas abaixo utilizando a ferramenta de sua preferência:

**a) Página principal** - Deverá listar os indicadores do ODS 3.

**b) Métricas de indicadores** - Ao selecionar um indicador, o usuário deverá ser capaz de analisar as métricas do indicador escolhido, com detalhes de período, estados e outros que estejam disponíveis.

Disponibilize os wireframes nos prints em seção exclusiva para o frontend no repositório do projeto.

- 2) Codificação:** Implemente os wireframes prototipados no passo anterior.
- Disponibilize a aplicação frontend na pasta **/frontend**, no repositório do projeto.
  - No arquivo readme, crie uma seção com instruções para execução do frontend.
  - No arquivo readme, crie uma seção com prints do frontend sendo executado.

## Global Solution – Testes

### 1) Testes de integração

Apresente um conjunto de testes de integração para cada um dos endpoints criados, utilizando frameworks apropriados para a linguagem escolhida.

- Disponibilize os testes na pasta **/tests**, no repositório do projeto.
- No arquivo readme, crie uma seção com instruções para execução dos testes de integração.
- No arquivo readme, crie uma seção com prints dos relatórios de execução dos testes de integração



## Global Solution – Testes

### 2) Testes de Performance

Realize testes de performance usando K6, simulando 100 acessos durante 5 minutos

- Disponibilize os testes na pasta **/tests**, no repositório do projeto.
- No arquivo readme, crie uma seção com instruções para execução dos testes de integração.
- No arquivo readme, crie uma seção com prints dos relatórios de execução dos testes de integração

## Global Solution – Publicação

Containerize sua aplicação frontend e backend utilizando Docker, criando um Dockerfile apropriado e fornecendo instruções de construção e execução no arquivo README da aplicação.

## Global Solution – Pontos de atenção

- Suas entregas deverão ser documentadas claramente, com instruções de configuração, execução dos serviços e testes.
- A qualidade do código, a aderência aos requisitos, e as boas práticas de desenvolvimento e documentação serão consideradas na avaliação.
- As soluções devem ser entregues na plataforma especificada pelo curso no prazo determinado, contendo todos os arquivos de código-fonte, scripts, testes e documentação necessários.

## **| Global Solution** – Critério de Avaliação

- Corretude na implementação de esquemas e modelos de dados (20 pontos).
- Funcionalidade e design da API, incluindo a aderência aos padrões RESTful (30 pontos).
- Qualidade e abrangência dos testes de integração e performance (25 pontos).
- Completude e clareza na containerização e documentação da aplicação (25 pontos).