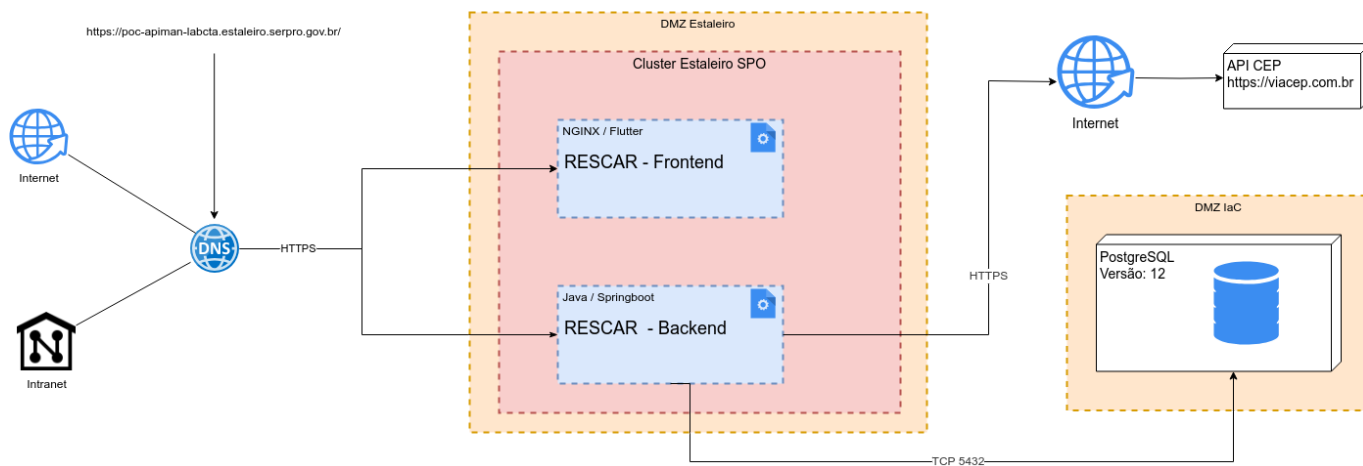


Poc Api Managment

Poc para validação de ferramentas para Api Managment utilizando o **Rescar** como base.

Rescar - Estaleiro

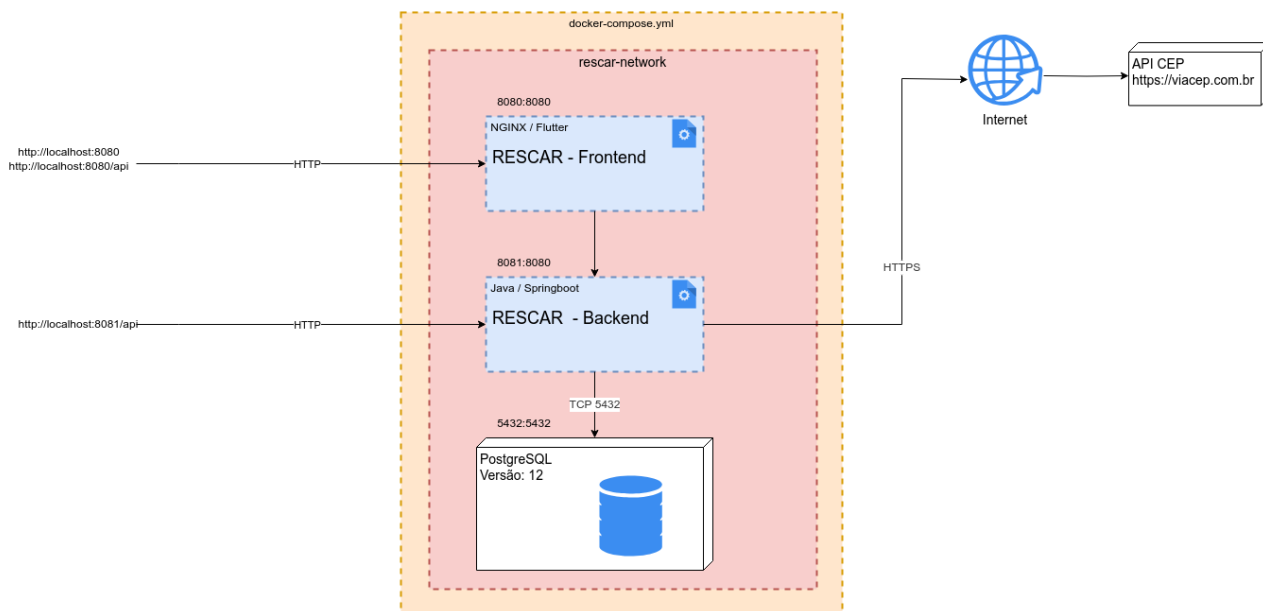


[Frontend Rescar - Estaleiro](#)

[Swagger Rescar - Estaleiro](#)

[Console - Estaleiro](#)

Rescar - Docker



[Download docker_rescar.zip](#)

```
# extrair o conteúdo da pasta docker_rescar.zip
> cd docker_rescar
> docker-compose up
```

Frontend Rescar - Docker: <http://localhost:8080>

Swagger Rescar - Docker: <http://localhost:8080/api/swagger-ui/index.html>

Usuários de Teste

Usuário	Senha
administrador@hotmail.com	123456
colaborador@hotmail.com	123456
motorista@hotmail.com	123456

Usando o Swagger

- Primeiro efetuar o login utilizando o serviço **/login**

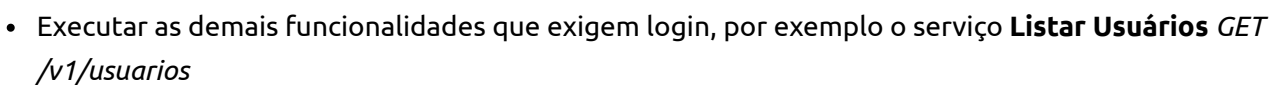
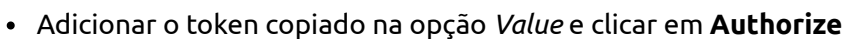
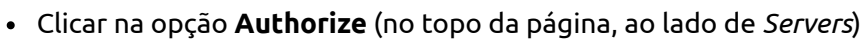
The screenshot shows the Swagger UI for the **auth-resource** API. The selected endpoint is **POST /login**. The **Parameters** section is empty. The **Request body** is required and set to **application/json**. The request body content is:

```
{  "email": "administrador@hotmail.com",  "senha": "123456"}
```

At the bottom, there is a blue **Execute** button.

- Clicar em **Try it out**
- Preencher o **Request body** com email/senha
- Clicar em **Execute**

- Copiar o token presente no cabeçalho **authorization** da resposta, sem o **Bearer**



GET

/v1/usuarios

⌵ 🔒

Listar usuarios

Parameters

Try it out

Name	Description
perfil string (query)	<input type="text" value="perfil"/>

GET

/v1/usuarios

⌵ 🔒

Listar usuarios

Parameters

Cancel

Name	Description
perfil string (query)	<input type="text" value="perfil"/>

Execute

- Clicar em **Try it out**
- Clicar em **Execute**

Request URL

http://localhost:8080/api/v1/usuarios

Server response

CodeDetails

200

Response body

```
[
  {
    "id": "d0ff4b09-b179-4a54-a3b1-dc6090aface2",
    "email": "administrador@hotmail.com",
    "nome": "Administrador Teste",
    "perfil": "A",
    "situacao": "A",
    "endereco": null
  },
  {
    "id": "1c164984-24bb-4655-89d1-a39409d6fd80",
    "email": "motorista@hotmail.com",
    "nome": "Motorista Teste",
    "perfil": "M",
    "situacao": "A",
    "endereco": null
  },
  {
    "id": "f78a817d-83bc-438e-a934-1d2acc6ec3c8",
    "email": "colaborador@hotmail.com",
    "nome": "Colaborador Teste",
    "perfil": "C",
    "situacao": "A",
    "endereco": null
  },
  {
    "id": "dd6938de-1c37-4443-a792-fc85bbf70b7a",
    "email": "colaborador2@hotmail.com",
  }
]
```

Download

Response headers

```
cache-control: no-cache,no-store,max-age=0,must-revalidate
connection: keep-alive
content-type: application/json
date: Mon, 02 Oct 2023 16:35:21 GMT
expires: 0
pragma: no-cache
server: nginx/1.25.1
transfer-encoding: chunked
vary: Origin,Access-Control-Request-Method,Access-Control-Request-Headers
x-content-type-options: nosniff
x-xss-protection: 1; mode=block
```

Desenvolvimento local

Alguns comandos úteis no desenvolvimento local:

```
# build/atualização da versão rodando no docker local
> ./build-rescar-docker.sh
```

```
# build/deploy do Frontend para o Estaleiro
> ./estaleiro/frontend/deploy.sh

# build/deploy do Backend para o Estaleiro
> ./estaleiro/backend/deploy.sh

# build flutter web
> cd ./frontend/rescar
> flutter build web

# executar flutter web em desenvolvimento
> cd ./frontend/rescar
> flutter run -d chrome
```

- **[DICA] Desenvolvimento Frontend x Backend LOCAL**

Se for executar o *frontend* e/ou *backend* localmente sem o auxílio do docker, verificar o arquivo *./frontend/rescar/lib/utils/chopper_clients.dart* se o **baseUrl** está apontando para o *backend* correto. O padrão é **/api** para acesso no *estaleiro* e quando executando backend e frontend via *docker*.

Por exemplo, para o desenvolvimento local executando o frontend manualmente e acessando o backend no docker seria **baseUrl = http://localhost:8081**.