Documentação para executar os microsserviços – Prodigyan

Passos para a implementação dos containers na AWS

- Criar um banco de dados seguindo as orientações do documento dicionário do banco de dados na gcp.

- Criar os contêineres docker na seguinte ordem abaixo, sempre informando as variáveis de ambiente esperadas, para a conexão com o banco de dados seja feita corretamente.

- Por último, o container contendo o Nginx, deve conter o IP do container na AWS e sua respectiva porta.

Containers

Exemplo ao final do documento

Container tipos

Imagem: brunonasccimentto/tipos:1.0

Porta do container: 3001

Protocol: http

Variáveis de ambientes esperadas:

- DB_HOST = {host do banco de dados}

- DB_PORT = {porta do banco de dados}

- DB_USERNAME = {usuário do banco de dados}

- DB_PASSWORD = {senha do banco de dados}

- DB_DATABASE = {nome do banco que será acessado}

Linhas de comando para executar o container:

- docker pull brunonasccimentto/tipos:1.0

- docker run -p 3001:3001 -d -e DB_HOST={ host do banco de dados } -e DB_PORT={ porta do banco de dados } -e DB_USERNAME={ usuário do banco de dados } -e DB_PASSWORD={ senha do banco de dados } -e DB_DATABASE={ nome do banco que será acessado } --name tipos brunonasccimentto/tipos:1.0

Chamada para api deste container:

http://{ip público do container}:3001/cliente/tipos

Obs: substituía todo o {} pelo valor da variável

Container gravar

Imagem: brunonasccimentto/gravar:1.0

Porta do container: 3002

Protocol: http

Variáveis de ambientes esperadas:

- DB_HOST = {host do banco de dados}
- DB_PORT = {porta do banco de dados}
- DB_USERNAME = {usuário do banco de dados}
- DB_PASSWORD = {senha do banco de dados}
- DB_DATABASE = {nome do banco que será acessado}

Linhas de comando para executar o container:

- docker pull brunonasccimentto/gravar:1.0
- docker run -p 3002:3002 -d -e DB_HOST={ host do banco de dados } -e DB_PORT={ porta do banco de dados } -e DB_USERNAME={ usuário do banco de dados } -e DB_PASSWORD={ senha do banco de dados } -e DB_DATABASE={ nome do banco que será acessado } --name gravar brunonasccimentto/gravar:1.0

Chamada para api deste container:

http://{ip público do container}:3002/cliente/gravar

Obs: substituía todo o {} pelo valor da variável

Container buscar-id

Imagem: brunonasccimentto/buscar-id:1.0

Porta do container: 3003

Protocol: http

Variáveis de ambientes esperadas:

- DB_HOST = {host do banco de dados}
- DB_PORT = {porta do banco de dados}
- DB_USERNAME = {usuário do banco de dados}
- DB_PASSWORD = {senha do banco de dados}
- DB_DATABASE = {nome do banco que será acessado}

Obs: substituía todo o {} pelo valor da variável

Linhas de comando para executar o container:

- docker pull brunonasccimentto/buscar-id:1.0
- docker run -p 3003:3003 -d -e DB_HOST={ host do banco de dados } -e DB_PORT={ porta do banco de dados } -e DB_USERNAME={ usuário do banco de dados } -e DB_PASSWORD={ senha do banco de dados } -e DB_DATABASE={ nome do banco que será acessado } --name buscar-id brunonasccimentto/buscar-id:1.0

Chamada para api deste container:

http://{ip público do container}:3003/cliente/buscar/id/{id do cliente}

Container buscar-email

Imagem: brunonasccimentto/buscar-email:1.0

Porta do container: 3004

Protocol: http

Variáveis de ambientes esperadas:

- DB_HOST = {host do banco de dados}
- DB_PORT = {porta do banco de dados}
- DB_USERNAME = {usuário do banco de dados}
- DB_PASSWORD = {senha do banco de dados}
- DB_DATABASE = {nome do banco que será acessado}

Obs: substituía todo o {} pelo valor da variável

Linhas de comando para executar o container:

- docker pull brunonasccimentto/buscar-email:1.0
- docker run -p 3004:3004 -d -e DB_HOST={ host do banco de dados } -e DB_PORT={ porta do banco de dados } -e DB_USERNAME={ usuário do banco de dados } -e DB_PASSWORD={ senha do banco de dados } -e DB_DATABASE={ nome do banco que será acessado } --name buscar-email brunonasccimentto/buscar-email:1.0

Chamada para api deste container:

http://{ip público do container}:3004/cliente/buscar/email/{email do cliente}

Container todos

Imagem: brunonasccimentto/todos:1.0

Porta do container: 3005

Protocolo: http

Variáveis de ambientes esperadas:

- DB_HOST = {host do banco de dados}
- DB_PORT = {porta do banco de dados}
- DB_USERNAME = {usuário do banco de dados}
- DB_PASSWORD = {senha do banco de dados}
- DB_DATABASE = {nome do banco que será acessado}

Linhas de comando para executar o container:

- docker pull brunonasccimentto/todos:1.0
- docker run -p 3005:3005 -d -e DB_HOST={ host do banco de dados } -e DB_PORT={ porta do banco de dados } -e DB_USERNAME={ usuário do banco de dados } -e DB_PASSWORD={ senha do banco de dados } -e DB_DATABASE={ nome do banco que será acessado } --name todos brunonasccimentto/todos:1.0

Chamada para a api deste container:

http://{ip público do container}:{porta}/cliente/todos

Obs: substituía todo o {} pelo valor da variável

Container prodigyan-nginx

Imagem: brunonasccimentto/prodigyan-nginx:1.0

Porta: 8000

Protocolo: http

Variáveis de ambiente esperadas:

- IP_CONTAINER_TIPOS = {ip público do container tipos}
- PORT_TIPOS = {porta do container tipos}
- IP_CONTAINER_GRAVAR = {ip público do container gravar}
- PORT_GRAVAR = {porta do container gravar}
- IP_CONTAINER_BUSCAR_ID = {ip público do container buscar-id}
- PORT_BUSCAR_ID = {porta do container buscar-id}
- IP_CONTAINER_BUSCAR_EMAIL = {ip público do container buscar-email}
- PORT_BUSCAR_EMAIL = {porta do container buscar-email}
- IP_CONTAINER_TODOS = {ip público do container todos}
- PORT_TODOS = {porta do container todos}

Linhas de comando para executar o container:

- docker pull brunonasccimentto/prodigyan-nginx:1.0
- docker run -d -p 8000:80 -e IP_CONTAINER_TIPOS={ ip público do container tipos } -e PORT_TIPOS={ porta do container tipos } -e IP_CONTAINER_BUSCAR_EMAIL={ ip público do container buscar-email } -e PORT_BUSCAR_EMAIL={ porta do container buscar-email } -e IP_CONTAINER_BUSCAR_ID={ ip público do container buscar-id } -e PORT_BUSCAR_ID={ porta do container buscar-id } -e IP_CONTAINER_GRAVAR={ ip público do container gravar } -e PORT_GRAVAR={ porta do container gravar } -e IP_CONTAINER_TODOS={ ip público do container todos } -e PORT_TODOS={ porta do container todos } --name prodigyan-nginx brunonasccimentto/prodigyan-nginx:1.0

Obs: substituía todo o ${} pelo valor da variável

Chamada para a api deste container:

http://{ip público do container}:8000/cliente/todos

/cliente/buscar/email/{email do cliente}

/cliente/buscar/id/{id do cliente}

/cliente/gravar

/cliente/tipos

Exemplos

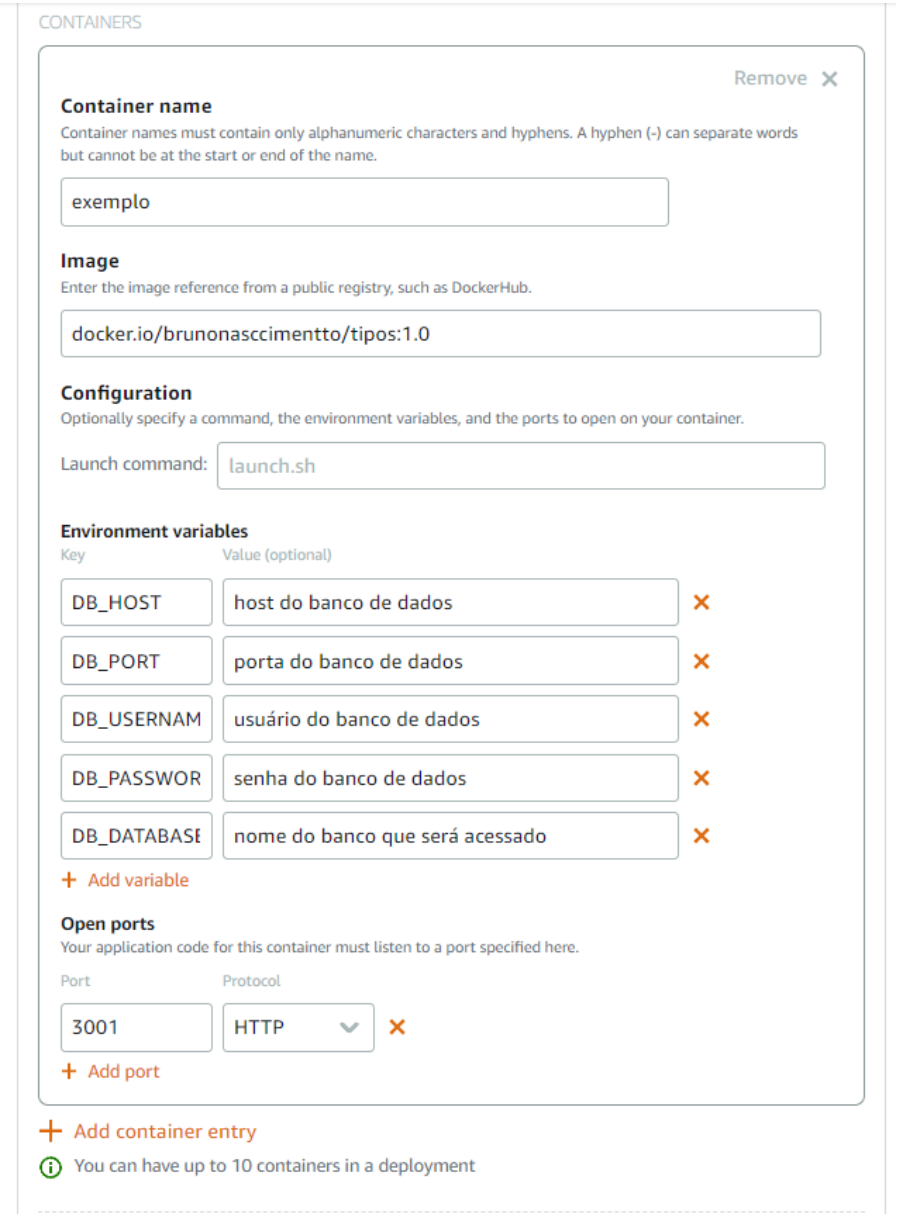

Figura 1 Exemplo de como os dados devem ser inseridos

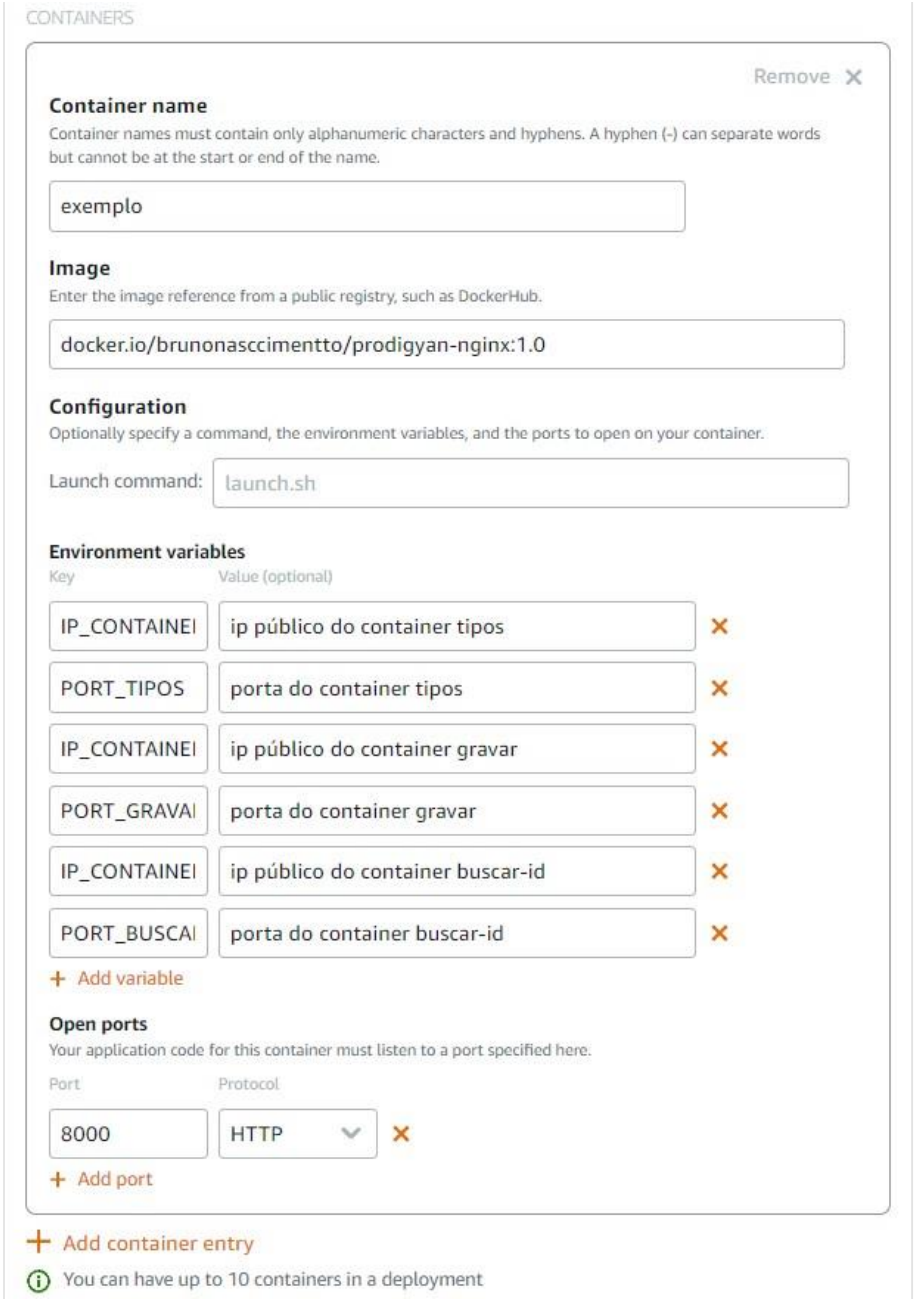

Figura 2 Exemplo do container Nginx