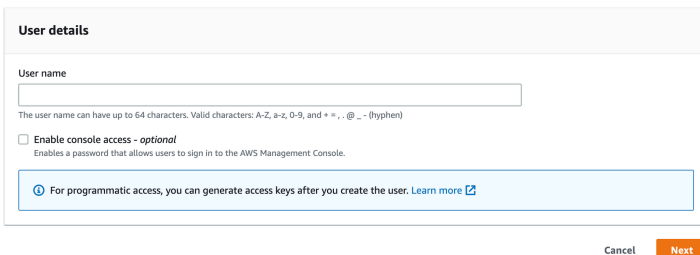


Passos para implementação do Projeto - Parte 2

Amazon Web Services

- Acessar a console da AWS. Na barra de pesquisas, digite IAM. Na seção Services, clique em IAM.
- Clique em Add user, insira o nome **luxxy-covid-testing-system-pt-app1** e clique em Next para criar o usuário do tipo programmatic.

Specify user details



The screenshot shows the 'Specify user details' form in the AWS IAM console. It has a title bar 'User details'. Below it is a 'User name' label and a text input field. A note below the field states: 'The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and +, -, @, _ - (hyphen)'. There is a checkbox labeled 'Enable console access - optional' with a sub-note: 'Enables a password that allows users to sign in to the AWS Management Console.' At the bottom, there is a light blue box with an information icon and the text: 'For programmatic access, you can generate access keys after you create the user. [Learn more](#)'. At the bottom right of the form are 'Cancel' and 'Next' buttons.

- Após avançar, em **Set permissions**, clique no botão Attach existing policies directly.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

☐ Add user to group

Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ Copy permissions

Copy all group memberships, attached managed policies, and inline policies from an existing user.

☒ Attach policies directly

Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1037)

Choose one or more policies to attach to your new user.

< 1 2 3 4 5 6 7 ... S2 > ⌕

<input type="checkbox"/>	Policy name	Type	Attached entities
<input type="checkbox"/>	AccessAnalyzerServiceRolePolicy	AWS managed	0
<input type="checkbox"/>	AdministratorAccess	AWS managed - job function	1
<input type="checkbox"/>	AdministratorAccess-Amplify	AWS managed	0
<input type="checkbox"/>	AdministratorAccess-AWSElasticBeanstalk	AWS managed	0
<input type="checkbox"/>	AlexaForBusinessDeviceSetup	AWS managed	0
<input type="checkbox"/>	AlexaForBusinessFullAccess	AWS managed	0

- Digite **AmazonS3FullAccess** em **Filter distributions by text, property or value** e aperte **Enter**.
- Selecione **AmazonS3FullAccess**

Permissions policies (1/1037)

Choose one or more policies to attach to your new user.

1 match

Clear filters

< 1 > ⌕

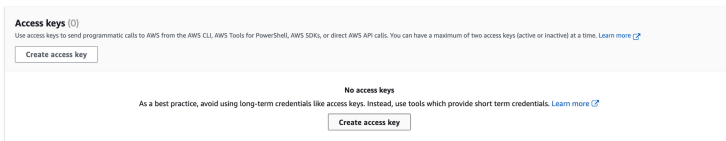
<input checked="" type="checkbox"/>	Policy name	Type	Attached entities
<input checked="" type="checkbox"/>	AmazonS3FullAccess	AWS managed	2

- Clique em **Next**
- Revise todos os detalhes
- Clique em **Create user**

Passos para fazer o download da chave de acesso

- Acesse o usuário **luxxy-covid-testing-system-pt-app1**
- Clique em **Security credentials**

- Navegue até a seção Access keys
- Clique em **Create access key**



- Selecione Command Line Interface (CLI) e I **understand the above recommendation and want to proceed to create an access key.**
- Clique em **Next**.
- Clique em **Create access key**
- Clique em **Download .csv file**
- Após o download finalizar, clique em Done.
- Com o download feito, renomeie o **.csv** para **accessKeys.csv**

Google Cloud Platform (GCP)

- Navegue até a Cloud SQL instance e crie um novo usuário **app** com a senha **welcome123456** no Cloud SQL MySQL database.
- Se conecte ao Google Cloud Shell

- Faça o download dos arquivos da missão 2 **diretamente para o Cloud Shell usando o comando wget abaixo:**

```
cd mkdir mission2_pt cd mission2_pt wget  
https://tcb-public-  
events.s3.amazonaws.com/icp/mission2.zip  
unzip mission2.zip
```

- Conecte ao MySQL DB em execução no Cloud SQL (assim que aparecer a janela para colocar a senha, insira **welcome123456**)

```
mysql --host=<public_ip_cloudsql> --  
port=3306 -u app -p
```

- Após estar conectado ao banco de dados da instância, crie a tabela de produtos para testes.

```
use dbcovidtesting; source  
mission2/pt/db/create_table.sql; show  
tables; exit;
```

- Habilite a Cloud Build API através do Cloud Shell.

```
# Comando para habilitar Cloud Build  
API gcloud services enable  
cloudbuild.googleapis.com
```

Known issue during this step

```
ERROR: (gcloud.builds.submit)
INVALID_ARGUMENT: could not resolve source
googleapi: Error 403:
989404026119@cloudbuild.gserviceaccount.c
does not have storage.objects.get access to
the Google Cloud Storage object., forbidden
Para solucionar: 1. Acesse o IAM & Admin;
2. Clique na sua Cloud Build Service
Account Exemplo:
989404026119@cloudbuild.gserviceaccount.c
Cloud Build Service Account 3. Na sua Cloud
Build Service Account, do lado direito,
clique em Edit principal 4. Clique em Add
another role (Adicionar outra função); 5.
Clique em Select Role, e filtre por Storage
Admin ou gcs. Selecione Storage Admin (Full
control of GCS resources). 6. Clique em
Save and retorne para o Cloud Shell.
```

- Faça o Build da Docker image e suba para o Google Container Registry. Por gentileza, substitua o <PROJECT_ID> com o My First Project ID

```
cd ~/mission2_pt/mission2/pt/app gcloud  
builds submit --tag  
gcr.io/<PROJECT_ID>/luxxy-covid-  
testing-system-app-pt
```

- Abra o Cloud Editor e edite o Kubernetes deployment file (luxxy-covid-testing-system.yaml) e atualize as variáveis abaixo (em vermelho) com o seu <PROJECT_ID> no caminho da imagem Docker no Google Container Registry, AWS Bucket, AWS Keys (do arquivo luxxy-covid-testing-system-pt-app1.csv) e o IP Privado do Cloud SQL Database.

```
cd ~/mission2_pt/mission2/pt/kubernetes
luxxy-covid-testing-system.yaml image:
gcr.io/<PROJECT_ID>/luxxy-covid-
testing-system-app-pt:latest ... -
name: AWS_BUCKET value: "luxxy-covid-
testing-system-pdf-pt-xxxx" - name:
S3_ACCESS_KEY value:
"xxxxxxxxxxxxxxxxxxxxxxxxxx" - name:
S3_SECRET_ACCESS_KEY value:
"xxxxxxxxxxxxxxxxxxxxxxxxxx" - name:
DB_HOST_NAME value: "172.21.0.3"
```

- Se conecte ao GKE (Google Kubernetes Engine) cluster via Console (seguir video)