**Steps in Amazon Web Services (AWS)**

* Access AWS console and go to IAM service
* Under Access management, Click in "Users", then "Add users". Insert the User name **luxxy-covid-testing-system-en-app1** and click in **Next** to create a programmatic user.

A screenshot of a computer

Description automatically generated

* On Set permissions, Permissions options, click in "Attach policies directly" button.

A screenshot of a computer

Description automatically generated

* Type **AmazonS3FullAccess** in **Search**.
* Select **AmazonS3FullAccess**

A screenshot of a computer

Description automatically generated

* Click in **Next**
* Review all details and click in Create user

A screenshot of a computer screen

Description automatically generated

**Steps to create access key:**

* Click on the user you have created.
* Go to Security credentials tab.
* Scroll down and go to Access keys section.
* Click on Create access key

A screenshot of a computer

Description automatically generated

* Select **Command Line Interface (CLI)** and **I understand the above recommendation and want to proceed to create an access key** checkbox.
* Click Next
* Click on Create access key
* Click on Download .csv file
* After download, click Done.
* Now, rename .csv file downloaded to **luxxy-covid-testing-system-en-app1.csv**

**Steps in Google Cloud Platform (GCP)**

* Navigate to Cloud SQL instance and create a new user **app** with password **welcome123456** on Cloud SQL MySQL database
* Connect to Google Cloud Shell
* **Download** the mission2 files to Google Cloud Shell using the wget command as shown below
* cd ~
* \*\*wget <https://tcb-public-events.s3.amazonaws.com/icp/mission2.zip\*\*>
* unzip mission2.zip
* Connect to MySQL DB running on Cloud SQL (once it prompts for the password, provide **welcome123456**). **Don’t forget to replace the placeholder with your Cloud SQL Public IP**

A screenshot of a computer

Description automatically generated

mysql --host=\*\*<replace\_with\_public\_ip\_cloudsql>\*\* --port=3306 -u app -p

* Once you're connected to the database instance, create the products table for testing purposes
* use dbcovidtesting;
* source ~/mission2/en/db/create\_table.sql
* show tables;
* exit;
* Enable Cloud Build API via Cloud Shell.
* gcloud services enable cloudbuild.googleapis.com

A screenshot of a computer

Description automatically generated

* Build the Docker image and push it to Google Container Registry.
* GOOGLE\_CLOUD\_PROJECT\_ID=$(gcloud config get-value project)
* cd ~/mission2/en/app
* gcloud builds submit --tag gcr.io/$GOOGLE\_CLOUD\_PROJECT\_ID/luxxy-covid-testing-system-app-en
* Open the Cloud Editor and edit the Kubernetes deployment file (**luxxy-covid-testing-system.yaml**) and update the variables below on **line 33** in **red** with your <PROJECT\_ID> on the Google Container Registry path, on **line 42** AWS Bucket name, AWS Keys (open file l**uxxy-covid-testing-system-en-app1.csv** and use **Access key ID** on **line 44** and **Secret access key** on **line 46**) and Cloud SQL Database Private IP on **line 48**.
* cd ~/mission2/en/kubernetes
* luxxy-covid-testing-system.yaml
* image: gcr.io/\*\*<PROJECT\_ID>/\*\*luxxy-covid-testing-system-app-en:latest
* ...
* - name: AWS\_BUCKET
* value: "\*\*luxxy-covid-testing-system-pdf-en-xxxx\*\*"
* - name: S3\_ACCESS\_KEY
* value: "\*\*xxxxxxxxxxxxxxxxxxxxx\*\*"
* - name: S3\_SECRET\_ACCESS\_KEY
* value: "\*\*xxxxxxxxxxxxxxxxxxxx\*\*"
* - name: DB\_HOST\_NAME
* value: "\*\*172.21.0.3\*\*"
* Connect to the GKE (Google Kubernetes Engine) cluster via Console

**Step 1**

A screenshot of a search box

Description automatically generated**Step 2**

A screenshot of a computer

Description automatically generated

* Deploy the application Luxxy in the Cluster
* cd ~/mission2/en/kubernetes
* kubectl apply -f luxxy-covid-testing-system.yaml
* Under **GKE** > **Workloads** > **Exposing Services**, get the application Public IP

**Step 1**

A screenshot of a computer

Description automatically generated

**Step 2**

A screenshot of a web page

Description automatically generated

* You should see the app up & running!

A screenshot of a hotel

Description automatically generated

* (Optional) Download a sample COVID testing and add an entry in the application.

**Click on the icon below to download the PDF ⬇️**

[covid-testing.pdf](https://prod-files-secure.s3.us-west-2.amazonaws.com/0d1b678b-cd91-4256-93c7-73b2e82396d5/4154a4e6-d3f4-4e33-8720-f97076f19653/covid-testing.pdf)

**Congratulations for finishing the hands-on project part 2! 🚀**

**Appendix I - Destroying the environment and starting over**

In case you have encountered any problem/error and want to reset the environment to start over, follow the step-by-step instructions below to remove the entire MultiCloud environment.

**[Google Cloud] Delete Kubernetes resources**

**Step 1**

A screenshot of a search box

Description automatically generated

**Step 2**

A screenshot of a computer

Description automatically generated

kubectl delete deployment luxxy-covid-testing-system

kubectl delete service luxxy-covid-testing-system

**[Google Cloud] Delete VPC Peering**

A screenshot of a computer

Description automatically generated

**[AWS] Delete files inside of S3**

A screenshot of a computer

Description automatically generated

**[Google Cloud] Delete remaining resources w/ Terraform - Cloud Shell**

cd ~/mission1/en/terraform/

terraform destroy

**Clean the Cloud Shell in AWS and Google Cloud**

**AWS**

cd ~

rm -rf mission\*

**Google Cloud**

cd ~

rm -rf mission\*

rm -rf .ssh