Steps to implement Hands-on Project - Mission 2

Amazon Web Services

- Access AWS console and go to IAM service
- Under Access management, Click in "Users", then "Add user" and create a programmatic user **"luxxy-covid-testing-system-en-app1"**
- On Set permissions, click on "Attach existing policies directly" button.
- Select AmazonS3FullAccess.
- Click on Next: Tags
- Click on Next: Review
- Click on Create user
- Click on Download .csv
- After download, rename .csv to **luxxy-covid-testing-system-en-app1.csv**

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

```
cd mkdir mission2_en cd mission2_en wget https://objectstorage.us-ashburn-
1.oraclecloud.com/p/t1RPAY7hbbdaSj9fhLJhYetCsFe5CEKufO56vb1l2_6FVz35PJzIukvTJKh
thecloudbootcamp/o/ICP2/mission2.zip unzip mission2.zip
```

- Connect to MySQL DB running on Cloud SQL (once it prompts for the password, provide **welcome123456**)

```
mysql --host=<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/mission2/en/db/create_table.sql;
show tables; exit;
```

- Build the Docker image and push it to Google Container Registry. Please replace the <PROJECT_ID> with your My First Project ID. ⚠️ Enable Cloud Build API.

```
# Command to enable Cloud Build API gcloud services enable
cloudbuild.googleapis.com
```

Known issue during this step

```
ERROR: (gcloud.builds.submit) INVALID_ARGUMENT: could not resolve source: goo
gleapi: Error 403: 989404026119@cloudbuild.gserviceaccount.com does not have
storage.objects.get access to the Google Cloud Storage object., forbidden To
solve it: 1. Access IAM & Admin; 2. Click on your Cloud Build Service Account
Example: 989404026119@cloudbuild.gserviceaccount.com Cloud Build Service Acco
unt 3. On your Cloud Build Service Account, right side, click on Edit princip
al 4. Click on Add another role 5. Click on Select Role, and filter by Storag
e Admin or gcs. Select Storage Admin (Full control of GCS resources). 6. Clic
k on Save and go to Cloud Shell.
```

```
cd ~/mission2_en/mission2/en/app gcloud builds submit --tag
gcr.io/<PROJECT_ID>/luxxy-covid-testing-system-app-en
```

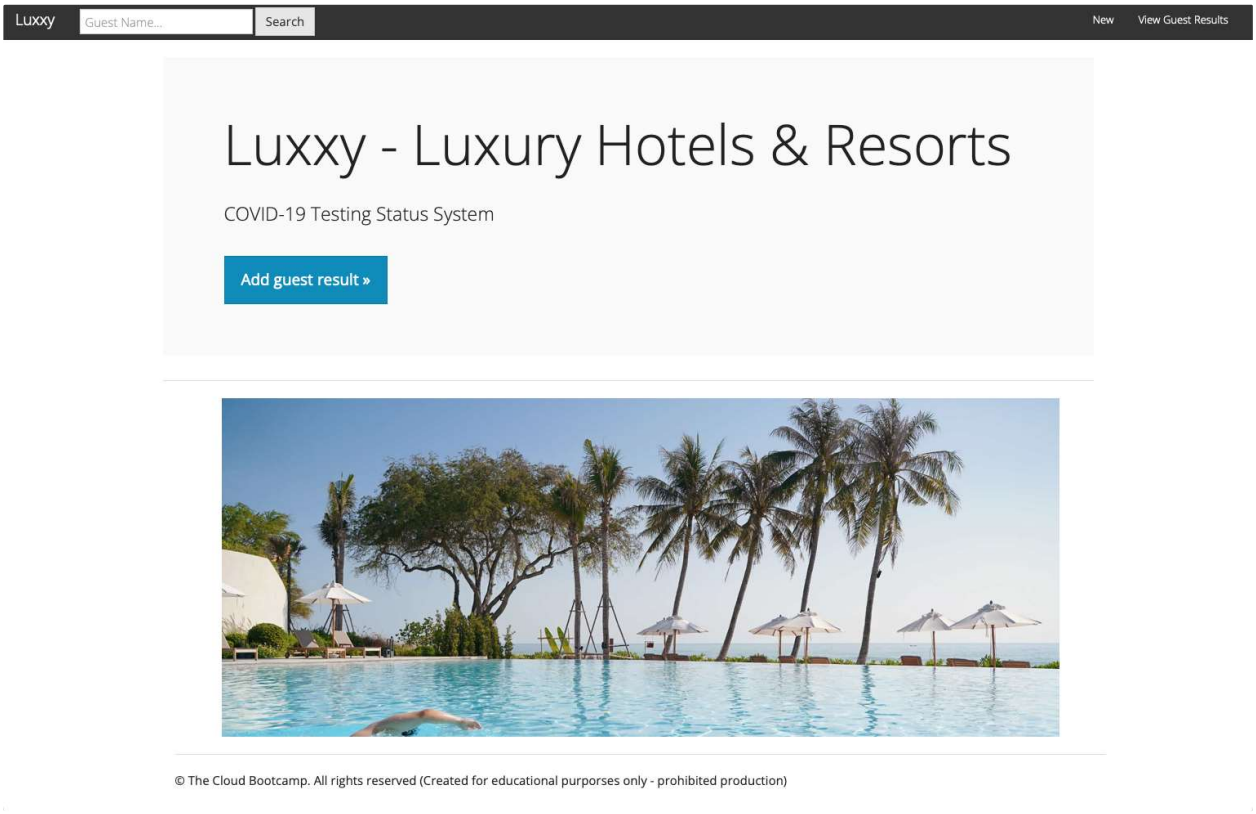
- Open the Cloud Editor and edit the Kubernetes deployment file (luxxy-covid-testing-system.yaml) and update the variables below in **red** with your <PROJECT_ID> on the Google Container Registry path, AWS Bucket name, AWS Keys (from **luxxy-covid-testing-system-en-app1.csv**) and Cloud SQL Database Private IP.

```
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: "xxxxxxxxxxxxxxxxxxxx" - 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 (follow the video)
- Deploy the application Pharma Store in the Cluster

```
cd ~/mission2_en/mission2/en/kubernetes kubectl apply -f luxxy-covid-testing-system.yaml
```

- Get the Public IP and test the application (CLICK HERE to download COVID-19 Testing result sample)
- You should see the app up & running! Congrats! 🎉


Luxxy
Guest Name...
Search
New
View Guest Results
Luxxy - Luxury Hotels & Resorts
COVID-19 Testing Status System
Add guest result »
© The Cloud Bootcamp. All rights reserved (Created for educational purposes only - prohibited production)