Module 5: Virtual Machines and Compute

Case Study Solution

This is Sample Case Study Solution and shall not be reused in any manner.

Case Study

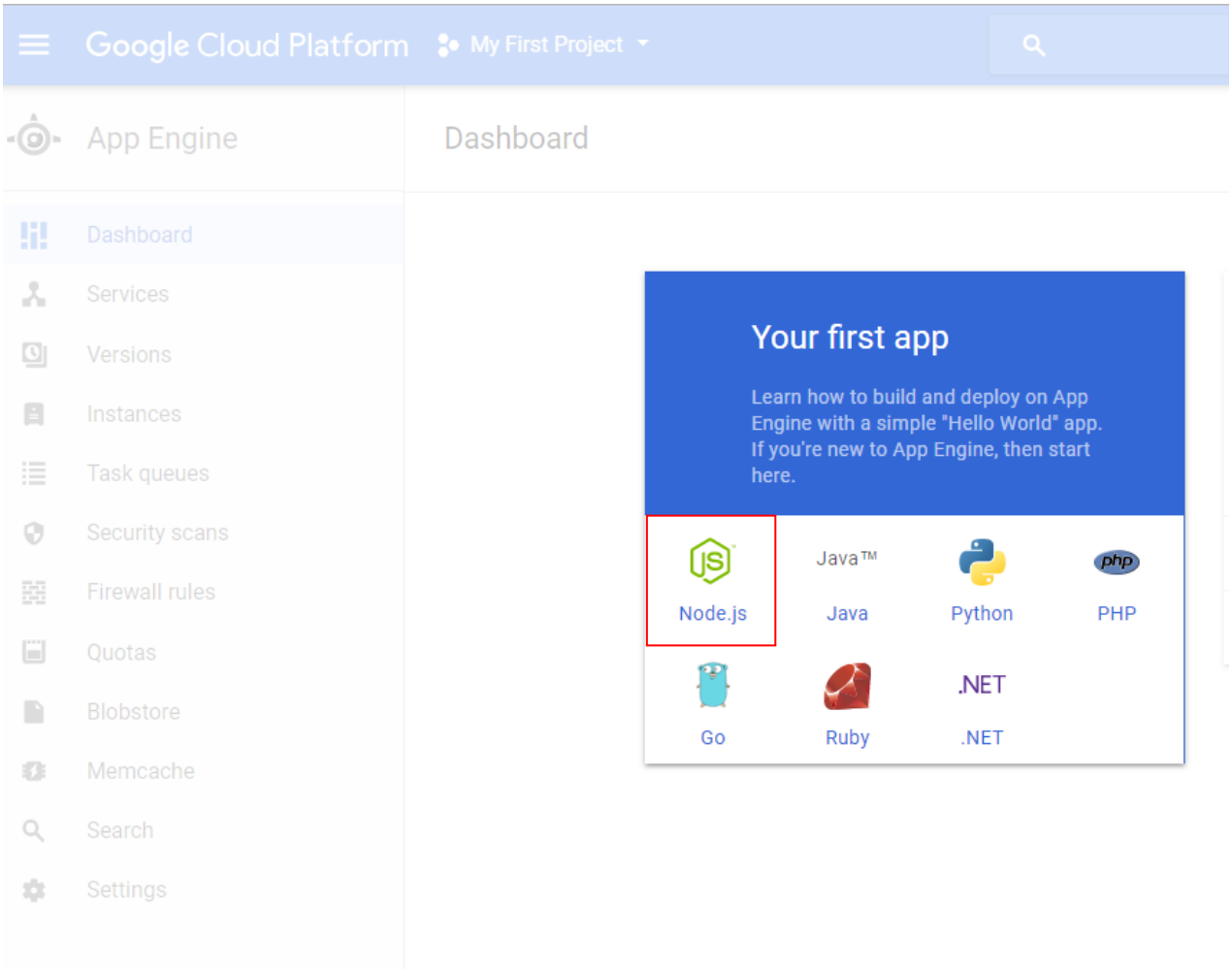
Problem Statement

Mr Lewis, who is CTO of a multinational company CloudGoogly after understanding the GCP and benefits of it, has called for a staff meeting. In the meeting he announced that CloudGoogly will now move to GCP. But he does not want to take any risk as its new for the organisation. He wants the sample app to be deployed first and then based on that experience would want to take further call. Ownership of trying this experiment is given to Roger, the product manager. Roger job was to try out deployment of sample app to google app engine to see how that works. If that is successful, Lewis would like to move the complete infrastructure of CloudGoogly to GCP.

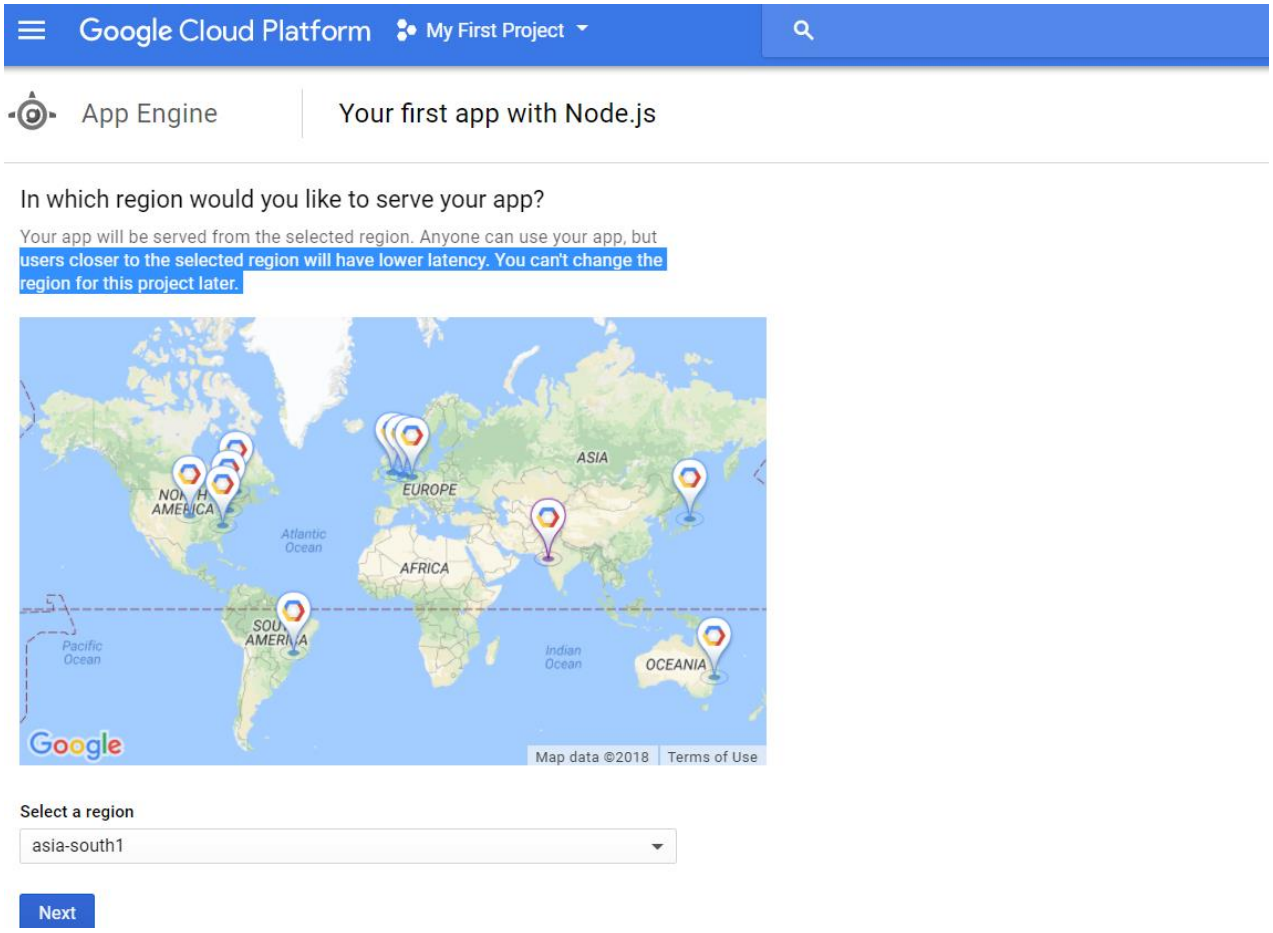
Solution

This assumes candidate has working knowledge on git and nodejs. If you don't know nodejs use a programming language of your choice like java, python or Go. You need to adapt the steps for your language.

1. On the GCP Console create an App Engine Project. Select nodejs as the platform



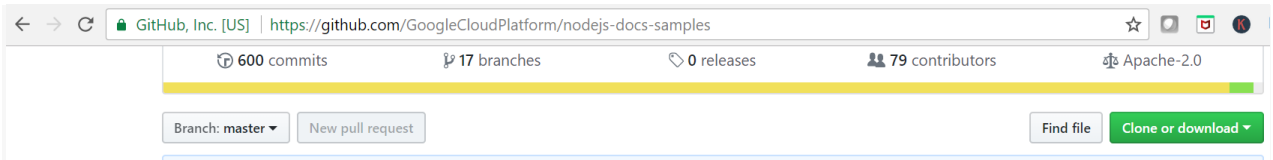
2. Select the region as asia-south1

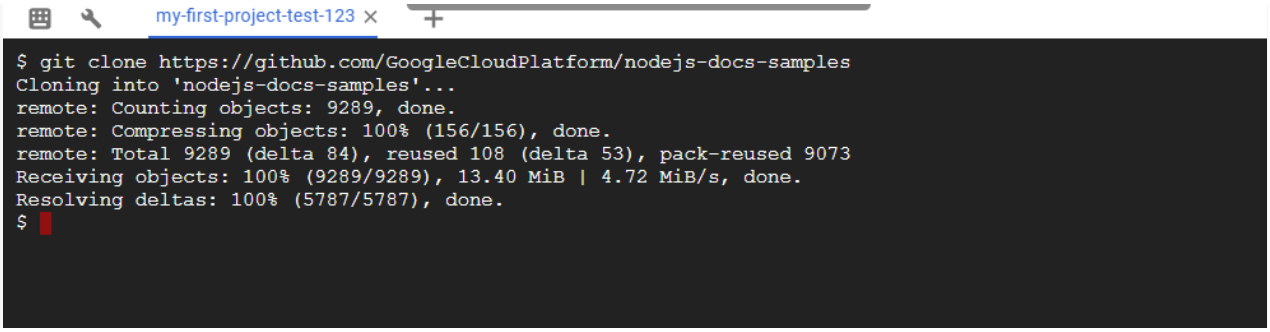


3) You add the project to the billing.

4) Open the gcloud util on cloud shell by giving “gcloud init” command

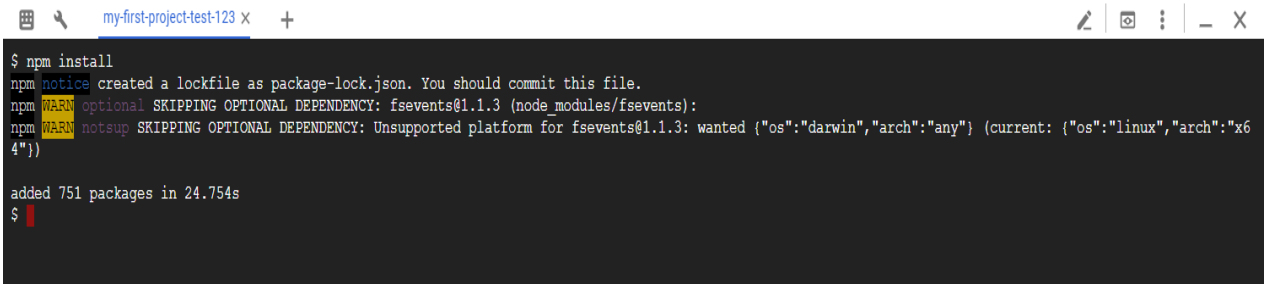
5) Download or clone the sample nodejs code for app engine from git clone https://github.com/GoogleCloudPlatform/nodejs-docs-samples



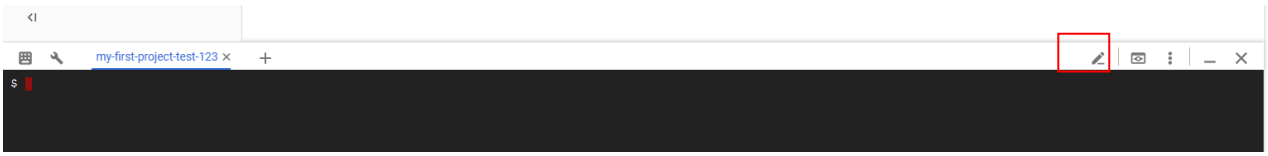


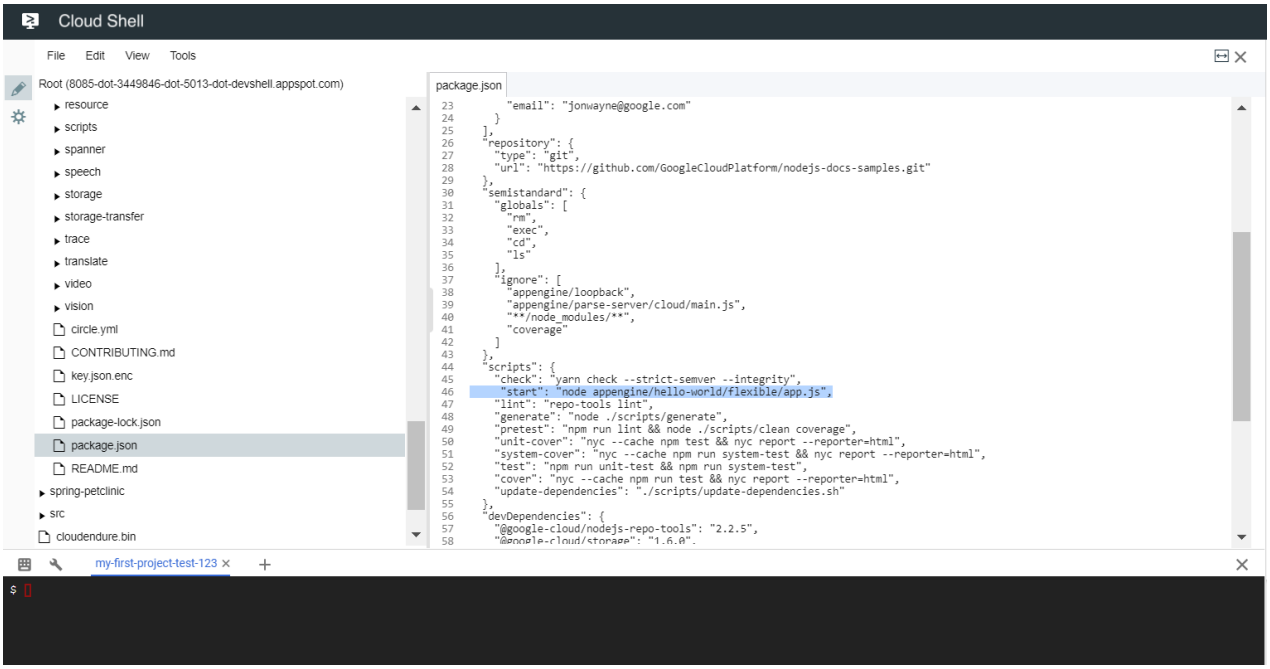
6) Change the working directory to the hello-world in the downloaded code.

7) npm install to install the dependency



8) Open the cloud shell editor:





9) In the nodejs-docs-sample folder there will be one file called package.json

Add start variable there as shown below:

```
},
"scripts": {
  "check": "yarn check --strict-semver --integrity",
  "start": "node appengine/hello-world/flexible/app.js",
  "lint": "repo-tools lint",
  "generate": "node ./scripts/generate",
  "pretest": "npm run lint && node ./scripts/clean coverage",
  "unit-cover": "nyc --cache npm test && nyc report --reporter=html",
  "system-cover": "nyc --cache npm run system-test && nyc report --reporter=html",
  "test": "npm run unit-test && npm run system-test",
  "cover": "nyc --cache npm run test && nyc report --reporter=html",
  "update-dependencies": "./scripts/update-dependencies.sh"
},
"devDependencies": {
  "@google-cloud/nodejs-repo-tools": "2.2.5",
  "@google-cloud/storage": "1.6.0",
  "ava": "0.25.0",
  "nyc": "11.6.0",
  "semistandard": "^12.0.1",
  "shellis": "0.8.1"
```

10) npm start to run the app locally.

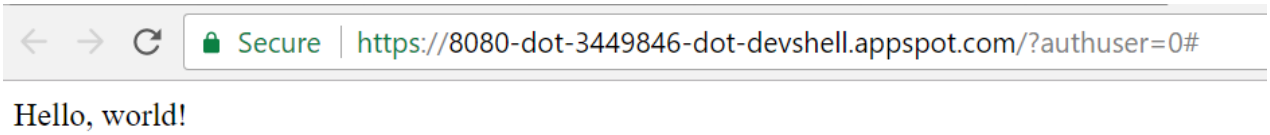
```
$ npm start

> appengine-hello-world@0.0.1 star
cs-samples/appengine/hello-world
> node app.js

App listening on port 8080
Press Ctrl+C to quit.
```

11) Check app on local machine by going to “web preview” option in cloud shell.

12. This will show Hello world:



Hello, world!

10) Run the command “gcloud app deploy flexible/app.yaml” . This will take few minutes to complete.

```
$ gcloud app deploy flexible/app.yaml
Services to deploy:
```

```
target project:   [edureka-gcp-training]
target service:   [default]
target version:   [20180409t001957]
target url:       [https://edureka-gcp-training.appspot.com]
```

Below screens just shows that it will take some time for app to get deployed:

```
Starting Step #0
Step #0: Pulling image: gcr.io/gcp-runtimes/nodejs/gen-dockerfile@sha256:ef598c7522a4e82b32280e76e3fca8ca6dc06ee567ce63b2ef5bd1fc08ef12f2
Step #0: sha256:ef598c7522a4e82b32280e76e3fca8ca6dc06ee567ce63b2ef5bd1fc08ef12f2: Pulling from gcp-runtimes/nodejs/gen-dockerfile
Step #0: Digest: sha256:ef598c7522a4e82b32280e76e3fca8ca6dc06ee567ce63b2ef5bd1fc08ef12f2
Step #0: Status: Downloaded newer image for gcr.io/gcp-runtimes/nodejs/gen-dockerfile@sha256:ef598c7522a4e82b32280e76e3fca8ca6dc06ee567ce63b2ef5bd1fc08ef12f2
Step #0: Checking for Node.js.
Finished Step #0
Starting Step #1
Step #1: Pulling image: gcr.io/cloud_builders/docker@sha256:b330fc4ad3784ca4a843e67946c251a4793c00882d2fcdebee3566db21d62abc
Step #1: sha256:b330fc4ad3784ca4a843e67946c251a4793c00882d2fcdebee3566db21d62abc: Pulling from cloud_builders/docker
Step #1: cf2c5e27174e: Already exists
Step #1: a7bc5eb395b5: Already exists
Step #1: a19c45128315: Already exists
Step #1: 711b1986f3b9: Pulling fs layer
Step #1: 711b1986f3b9: Verifying Checksum
Step #1: 711b1986f3b9: Download complete
Step #1: 711b1986f3b9: Pull complete
Step #1: Digest: sha256:b330fc4ad3784ca4a843e67946c251a4793c00882d2fcdebee3566db21d62abc
Step #1: Status: Downloaded newer image for gcr.io/cloud_builders/docker@sha256:b330fc4ad3784ca4a843e67946c251a4793c00882d2fcdebee3566db21d62abc
Step #1: Sending build context to Docker daemon    236kB
Step #1: Step 1/5 : FROM gcr.io/google-appengine/nodejs@sha256:8466a16b162e49e75086cdcbd9556904499b00497df1e6d4a52dd1cacf517fde
Step #1: sha256:8466a16b162e49e75086cdcbd9556904499b00497df1e6d4a52dd1cacf517fde: Pulling from google-appengine/nodejs
Step #1: Digest: sha256:8466a16b162e49e75086cdcbd9556904499b00497df1e6d4a52dd1cacf517fde
Step #1: Status: Downloaded newer image for gcr.io/google-appengine/nodejs@sha256:8466a16b162e49e75086cdcbd9556904499b00497df1e6d4a52dd1cacf517fde
Step #1:  ---> 8690a710dfd2
Step #1: Step 2/5 : COPY . /app/
Step #1:  ---> ddb2462705fd
Step #1: Step 3/5 : RUN /usr/local/bin/install_node '>=4.3.2'
Step #1:  ---> Running in 1829e3965cbc
Step #1: Removing intermediate container 1829e3965cbc
Step #1:  ---> 2a09c79c96a2
Step #1: Step 4/5 : RUN npm install --unsafe-perm ||   ((if [ -f npm-debug.log ]; then       cat npm-debug.log;     fi) && false)
Step #1:  ---> Running in bd1227996de7
```

```
Finished Step #1
PUSH
Pushing asia.gcr.io/edureka-gcp-training/appengine/default.20180409t001957:latest
The push refers to repository [asia.gcr.io/edureka-gcp-training/appengine/default.20180409t001957]
64ec98d9a41f: Preparing
eff224e0bb06: Preparing
77b4570ce8da: Preparing
5823921ad8a8: Preparing
e453f2cd1cc0: Preparing
8a67199ed2ba: Preparing
ea393ce87976: Preparing
51ca3e0924cb: Preparing
3949976ce03f: Preparing
21df82f90a72: Preparing
8a67199ed2ba: Waiting
ea393ce87976: Waiting
51ca3e0924cb: Waiting
3949976ce03f: Waiting
21df82f90a72: Waiting
77b4570ce8da: Layer already exists
e453f2cd1cc0: Layer already exists
5823921ad8a8: Layer already exists
64ec98d9a41f: Pushed
eff224e0bb06: Pushed
8a67199ed2ba: Layer already exists
ea393ce87976: Layer already exists
51ca3e0924cb: Layer already exists
21df82f90a72: Layer already exists
3949976ce03f: Layer already exists
latest: digest: sha256:6de8716171261001cbb7b6400db6e03ec4cdf31beb4e1f19718d687f165c8fff size: 2415
DONE
----------------------------------------------------------------------------------------------------------------

Updating service [default] (this may take several minutes)...-
```

Successful deployment confirmation will be displayed after a while:

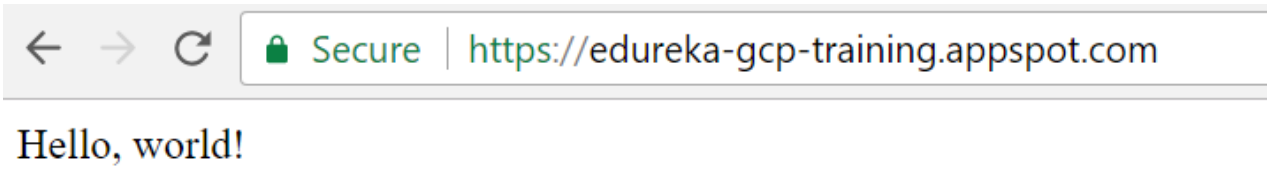
```
latest: digest: sha256:6de8716171261001cbb7b6400db6e03ec4cdf31beb4e1f19718d687f165c8fff size: 2415
DONE
----------------------------------------------------------------------------------------------------------------

Updating service [default] (this may take several minutes)...done.
Setting traffic split for service [default]...done.
Deployed service [default] to [https://edureka-gcp-training.appspot.com]

You can stream logs from the command line by running:
  $ gcloud app logs tail -s default

To view your application in the web browser run:
  $ gcloud app browse
$
```

11) You access the app by https://<projectid>.appspot.com/ so in this case you can access it by https://edureka-gcp-training.appspot.com



Congrats you had first deployed the app in the AppEngine.!!