

Lab notebook Week 7

Submitted by: Vishrut Sharma (OdinID: vishrut)

Table of Contents

07.1a: Terraform AWS Guestbook.....	2
4. Launching configuration.....	2
6. Adding ssh access.....	3
7. Adding the Guestbook application.....	3
8. View the Guestbook.....	4
07.1g: Terraform GCP Guestbook.....	4
4. Launching configuration.....	4
5. Adding an external IP address.....	5
6. Adding ssh access.....	6
7. Adding the Guestbook application.....	6
8. View the Guestbook.....	7
07.2g: Kubernetes Guestbook.....	8
4. Create Kubernetes cluster.....	8
5. Prepare a container image.....	8
7. Deploy the configuration.....	8
8. View the Guestbook.....	9
07.3a: APIs.....	11
4. Test code.....	11
8. Test code.....	12
07.3g: APIs (Slack, Knowledge Graph).....	12
2. Code.....	12
3. Code.....	12
8. Test the command.....	13
07.4a: Lambda, API Gateway Guestbook.....	13
10. Deploy API to production and view entries.....	13
12. API endpoint for signing (2).....	14
16. Configure and Deploy the Frontend.....	14
07.4g: Cloud Functions API Gateway Guestbook.....	15
10. Create the API Gateway.....	15
11. Test the API via Python Requests (GET).....	15
12. Test the API via Python Requests (POST).....	15
15. Version #1: Local file system.....	16
16. Version #2: Google Cloud Storage bucket.....	17

07.1a: Terraform AWS Guestbook

4. Launching configuration

- Take a screenshot showing the completion of the command including its output

The screenshot shows the AWS CloudShell interface in a browser window. The terminal session is titled 'us-east-1' and displays the output of a Terraform apply command. The output includes resource changes, a confirmation prompt asking if actions should be performed, and the final 'Apply complete!' message. A small modal window titled 'Odin ID v' is open, showing the value 'vishrut'. The bottom of the screen shows the AWS navigation bar.

```
us-east-1
  + security_group          = (known after apply)
  + source_dest_check       = true
  + subnet_id                = (known after apply)
  + tags_all                 = (known after apply)
  + tency                   = (known after apply)
  + user_data               = (known after apply)
  + user_data_base64         = (known after apply)
  + user_data_replace_on_change = false
  + vpc_security_group_ids   = (known after apply)

Plan: 1 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  ec2instance = (known after apply)

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.guestbook: Creating...
aws_instance.guestbook: Still creating... [10s elapsed]
aws_instance.guestbook: Still creating... [20s elapsed]
aws_instance.guestbook: Still creating... [30s elapsed]
aws_instance.guestbook: Creation complete after 32s [id=1-0c44ba3cb0532b8a7]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:
ec2instance = "34.238.116.67"
[cloudshell-user@ip-10-4-161-59 tf]$ 
```

- Take a screenshot that includes the VM's IP addresses

The screenshot shows the AWS EC2 Instances page. It displays the 'Instance summary for i-0c44ba3cb0532b8a7' with detailed information about the instance, including its public and private IP addresses, state, and VPC ID. Below the main summary, there is a smaller AWS CloudShell window showing the same instance details.

Instance summary for i-0c44ba3cb0532b8a7

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0c44ba3cb0532b8a7	34.238.116.67	172.31.84.21
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-34-238-116-67.compute-1.amazonaws.com
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-84-21.ec2.internal	ip-172-31-84-21.ec2.internal	-
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
-	t2.micro	Opt-in to AWS Compute Optimizer for recommendations.
Auto-assigned IP address	VPC ID	
34.238.116.67 [Public IP]	vpc-0b9c4137070e0204a	

6. Adding ssh access

- Take a screenshot of the successful ssh login from Cloud Shell.

The screenshot shows the AWS CloudShell interface. At the top, there's a browser-like header with tabs and a search bar. Below it, the AWS logo and 'Services' button are visible. The main area displays a terminal window titled 'us-east-1'. The terminal output shows a successful SSH connection to an Ubuntu 20.04 LTS server. It includes system information, update status, and a warning about adding the host to known hosts. A small modal window titled 'Odin ID v' is open in the top right, showing the user 'vishrut'. The bottom of the screen has a footer with copyright information and links for 'CloudShell', 'Feedback', and 'Language'.

```
new you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '107.21.145.87' (EDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1048-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 System information as of Sun May 14 18:23:58 UTC 2023

 System load: 0.51           Processes:          104
 Usage of /: 16.4% of 7.69GB Users logged in: 0
 Memory usage: 24%          IPv4 address for eth0: 172.31.94.166
 Swap usage: 0%             

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright*.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-94-166:~$
```

7. Adding the Guestbook application

- Take a screenshot of the output of the command that includes the IP address of the instance

The screenshot shows the AWS CloudShell interface. The terminal window displays the output of a Terraform 'apply' command. It shows the creation of a key pair ('aws_key_pair.kp'), security group rules ('aws_security_group.sg-guestbook'), and an EC2 instance ('aws_instance.guestbook'). The instance creation process is shown with progress bars and times. The output ends with a message 'Apply complete! Resources: 3 added, 0 changed, 0 destroyed.' A small modal window titled 'Odin ID v' is open in the top right, showing the user 'vishrut'. The bottom of the screen has a footer with copyright information and links for 'CloudShell', 'Feedback', and 'Language'.

```
Plan: 3 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ ec2instance = (known after apply)

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

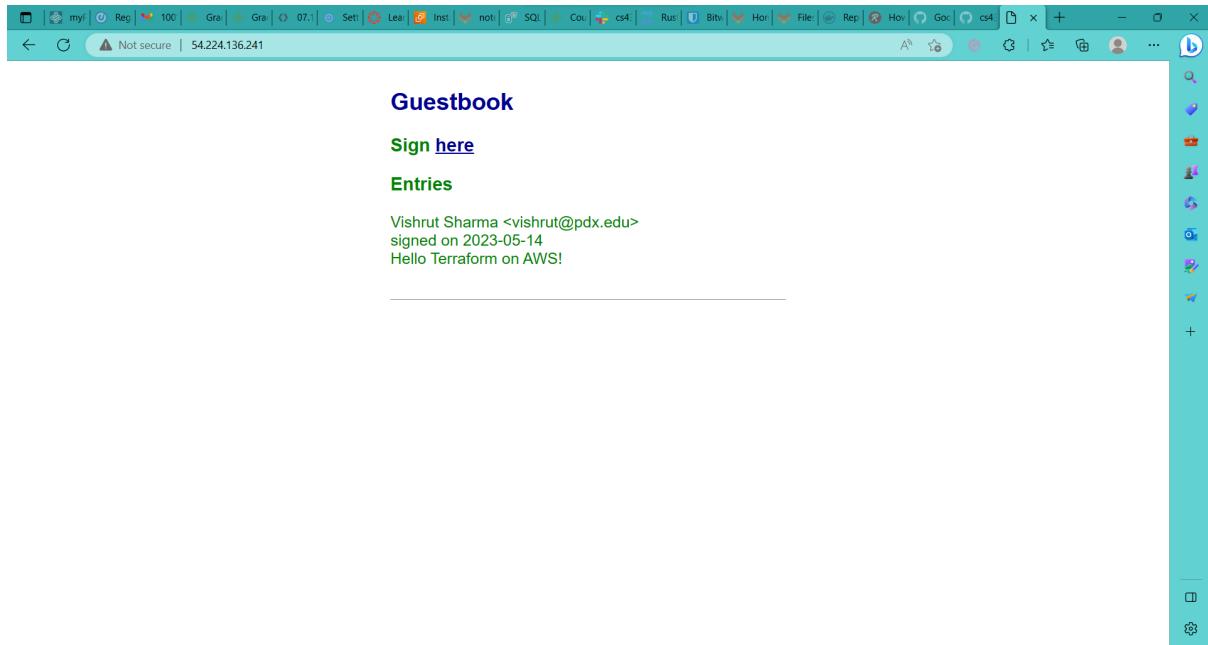
aws_key_pair.kp: Creating...
aws_security_group.sg-guestbook: Creating...
aws_instance.guestbook: Creation complete after 0s [id=sg-0f27c77cca496aabc]
aws_instance.guestbook: Creating...
aws_instance.guestbook: Still creating... [10s elapsed]
aws_instance.guestbook: Still creating... [20s elapsed]
aws_instance.guestbook: Still creating... [30s elapsed]
aws_instance.guestbook: Creation complete after 32s [id=i-0a259ce1039e657eb]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

Outputs:
ec2instance = "54.224.136.241"
[cloudshell-user@ip-10-4-161-59 tf]$
```

8. View the Guestbook

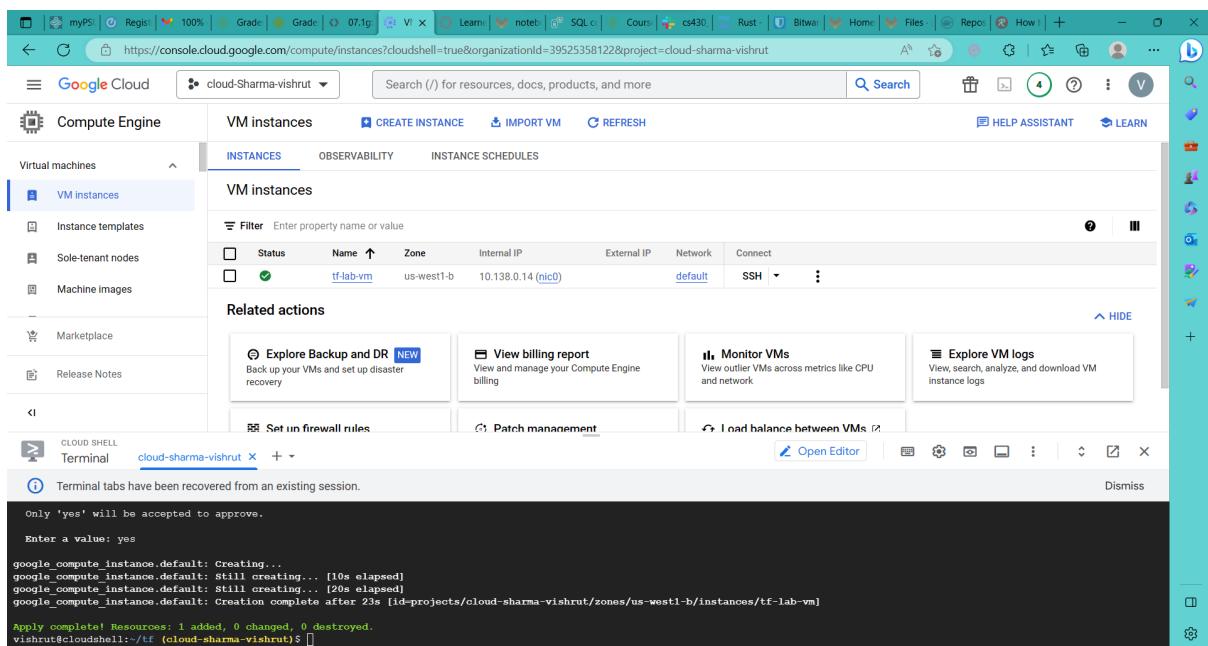
- Take a screenshot of the Guestbook including the URL with the entry in it.



07.1g: Terraform GCP Guestbook

4. Launching configuration

- Take a screenshot that includes the VM's IP addresses



5. Adding an external IP address

- Take a screenshot showing the completion of the command including its output

The screenshot shows a Google Cloud Compute Engine terminal window. The terminal is running a Terraform script to add an external IP address to a VM. The output shows the plan, changes, and confirmation steps. The final output shows the external IP address assigned to the VM.

```
tags = []
# (17 unchanged attributes hidden)

~ network_interface {
  name     = "nic0"
  # (6 unchanged attributes hidden)

  + access_config {
    nat_ip = (known after apply)
  }
}

# (3 unchanged blocks hidden)
}

Plan: 1 to add, 1 to change, 0 to destroy.

Changes to Outputs:
+ ip = (known after apply)

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

google_compute_address.static: Creating...
google_compute_address.static: Still creating... [10s elapsed]
google_compute_address.static: Creation complete after 11s [id:projects/cloud-sharma-vishrut/regions/us-west1/addresses/ipv4-address]
google_compute_instance.default: Modifying... [id:projects/cloud-sharma-vishrut/zones/us-west1-b/instances/tf-lab-vm]
google_compute_instance.default: Still modifying... [id:projects/cloud-sharma-vishrut/zones/us-west1-b/instances/tf-lab-vm, 10s elapsed]
google_compute_instance.default: Modifications complete after 11s [id:projects/cloud-sharma-vishrut/zones/us-west1-b/instances/tf-lab-vm]

Apply complete! Resources: 1 added, 1 changed, 0 destroyed.

Outputs:

ip = "34.82.33.216"
vishrut@cloudshell:~/tf (cloud-sharma-vishrut)$
```

- Take a screenshot that includes the VM's IP addresses

The screenshot shows the Google Cloud Compute Engine VM instances page. It lists a single VM instance named 'tf-lab-vm' with its details: Internal IP (10.138.0.14), External IP (34.82.33.216), Network (default), and Zone (us-west1-b). Below the table, there are related actions like Explore Backup and DR, View billing report, Monitor VMs, and Explore VM logs.

Status	Name	Zone	Internal IP	External IP	Network	Connect
✓	tf-lab-vm	us-west1-b	10.138.0.14 (nic0)	34.82.33.216 (nic0)	default	SSH

```
Enter a value: yes

google_compute_address.static: Creating...
google_compute_address.static: Still creating... [10s elapsed]
google_compute_address.static: Creation complete after 11s [id:projects/cloud-sharma-vishrut/regions/us-west1/addresses/ipv4-address]
google_compute_instance.default: Modifying... [id:projects/cloud-sharma-vishrut/zones/us-west1-b/instances/tf-lab-vm]
google_compute_instance.default: Still modifying... [id:projects/cloud-sharma-vishrut/zones/us-west1-b/instances/tf-lab-vm, 10s elapsed]
google_compute_instance.default: Modifications complete after 11s [id:projects/cloud-sharma-vishrut/zones/us-west1-b/instances/tf-lab-vm]

Apply complete! Resources: 1 added, 1 changed, 0 destroyed.

Outputs:

ip = "34.82.33.216"
vishrut@cloudshell:~/tf (cloud-sharma-vishrut)$
```

6. Adding ssh access

- Take a screenshot of the successful ssh login from Cloud Shell.

The screenshot shows a Google Cloud Cloud Shell terminal window. The URL in the address bar is <https://console.cloud.google.com/compute/instances?cloudshell=true&organizationId=39525358122&project=cloud-sharma-vishrut>. The terminal output shows a successful SSH login to an Ubuntu 20.04.5 LTS instance (ip = "34.82.33.216"). The session includes system information, memory usage, swap usage, and a note about ESM updates. The command vishrut@tf-lab-vm:~\$ is shown at the end.

```
ip = "34.82.33.216"
vishrut@cloudshell:~/tf (cloud-sharma-vishrut)$ ssh 34.82.33.216
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.15.0-1030-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon May 15 00:45:18 UTC 2023

System load: 0.0          Processes:           101
Usage of /: 19.7% of 9.51GB Users logged in: 0
Memory usage: 5%
Swap usage: 0%
IPv4 address for ens4: 10.138.0.14

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

vishrut@tf-lab-vm:~$
```

7. Adding the Guestbook application

- What resources are being added, changed, or destroyed?

Answer:

1. `cpu_platform`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
2. `current_status`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
3. `enable_display`: This attribute will be removed from the resource. Its value is `false`, and it will be set to `null` after the resource is destroyed.
4. `guest_accelerator`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
5. `id`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
6. `instance_id`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
7. `label_fingerprint`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
8. `labels`: This attribute will be removed from the resource. Its value is an empty map `{}`, and it will be set to `null` after the resource is destroyed.

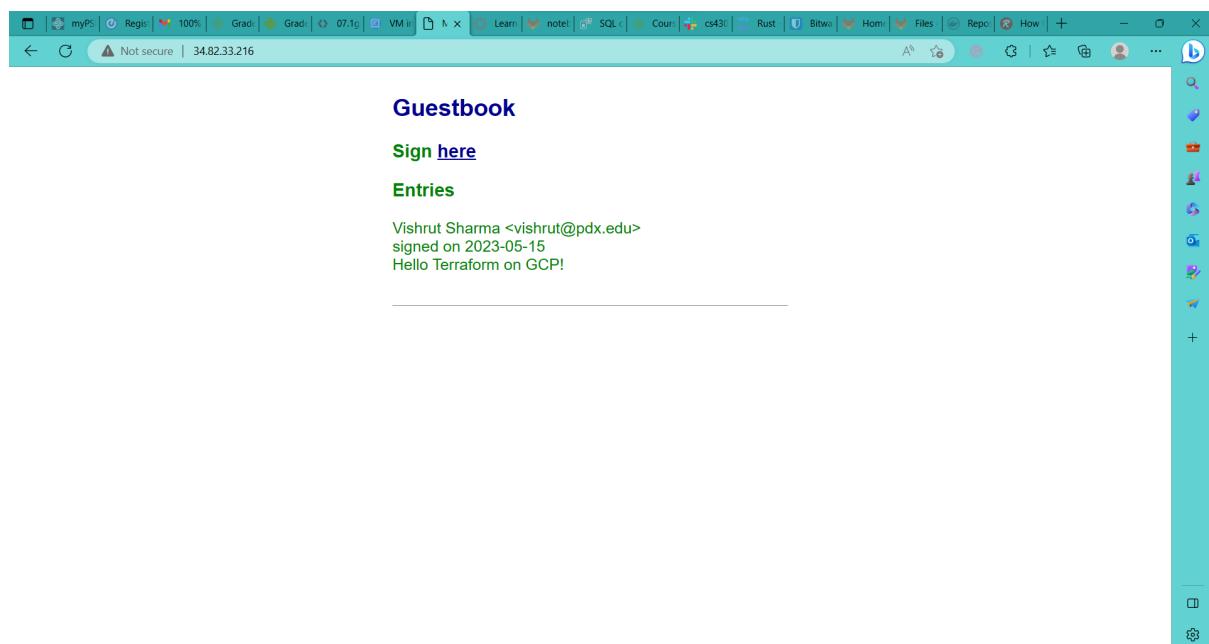
9. `metadata_fingerprint`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
10. `metadata_startup_script`: This attribute will be added to the resource. It contains a multi-line string that defines a Bash script to be executed when the new instance is created.
11. `min_cpu_platform`: This attribute will be added to the resource. Its value will be determined after the `apply` action is performed and is unknown to Terraform.
12. `resource_policies`: This attribute will be removed from the resource. Its value is an empty list `[]`, and it will be set to null after the resource is destroyed.
13. `self_link`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
14. `tags`: This attribute is changing. A new tag "http-server" will be added to the resource.
15. `tags_fingerprint`: The value of this attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform.
16. `boot_disk`: This block is changing. It contains details about the boot disk that will be attached to the new instance. The `device_name` attribute is expected to change after the `apply` action is performed, and its new value is unknown to Terraform. The `initialize_params` block contains attributes whose values are expected to change after the `apply` action is performed, and their new values are unknown to Terraform.

- **What part of the configuration forces a replacement to occur?**

Answer: `metadata-startup-script` forces the replacement to occur.

8. View the Guestbook

- **Take a screenshot of the Guestbook including the URL with the entry in it.**



07.2g: Kubernetes Guestbook

4. Create Kubernetes cluster

- What is the name of the Instance Template dynamically generated to create the two nodes (VMs)?

Answer: gke-guestbook-default-pool-25df0397

- What is the name of the Instance Group dynamically generated that the two nodes belong to?

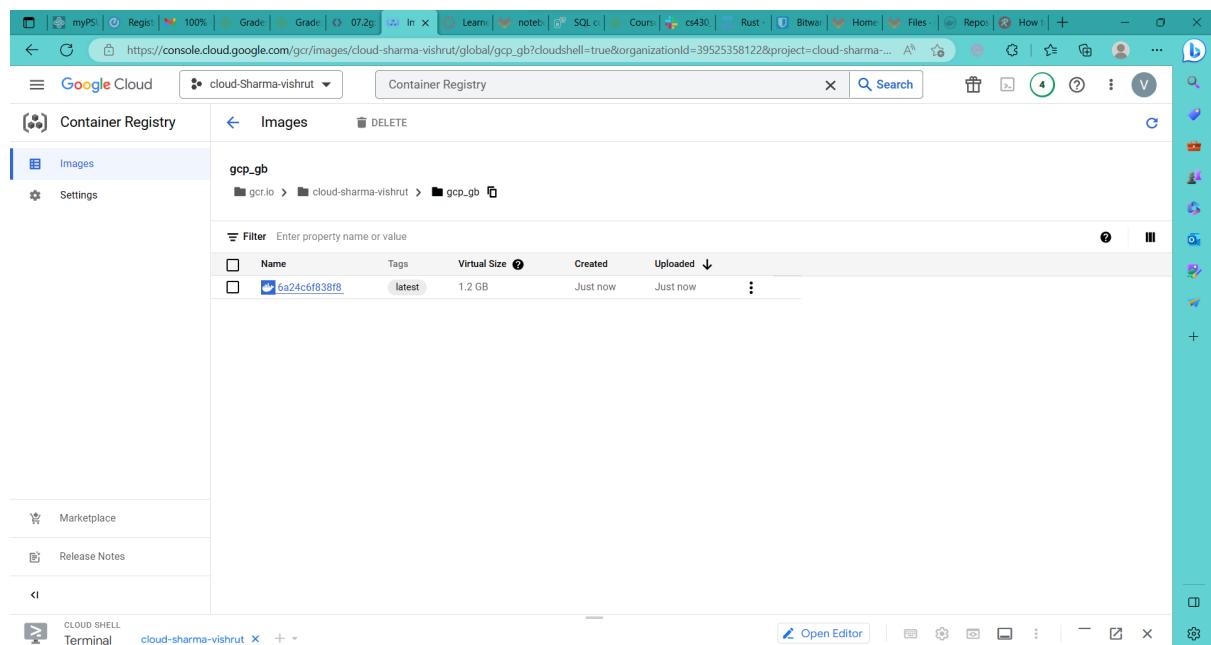
Answer: gke-guestbook-default-pool-25df0397-grp

- What are the names of the two nodes?

Answer: gke-guestbook-default-pool-25df0397-g9zz and
gke-guestbook-default-pool-25df0397-q7bt

5. Prepare a container image

- Take a screenshot of the container image created



7. Deploy the configuration

- Take a screenshot of the output of the following command when all 3 replicas reach a "Running" state.

```
vishrut@cloudshell:~/cs430-src/05_gcp_datastore $(cloud-sharma-vishrut)$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
guestbook-replicas-2gpch  1/1     Running   0          100s
guestbook-replicas-kmfrrn 1/1     Running   0          100s
guestbook-replicas-t7k4w   1/1     Running   0          100s
```

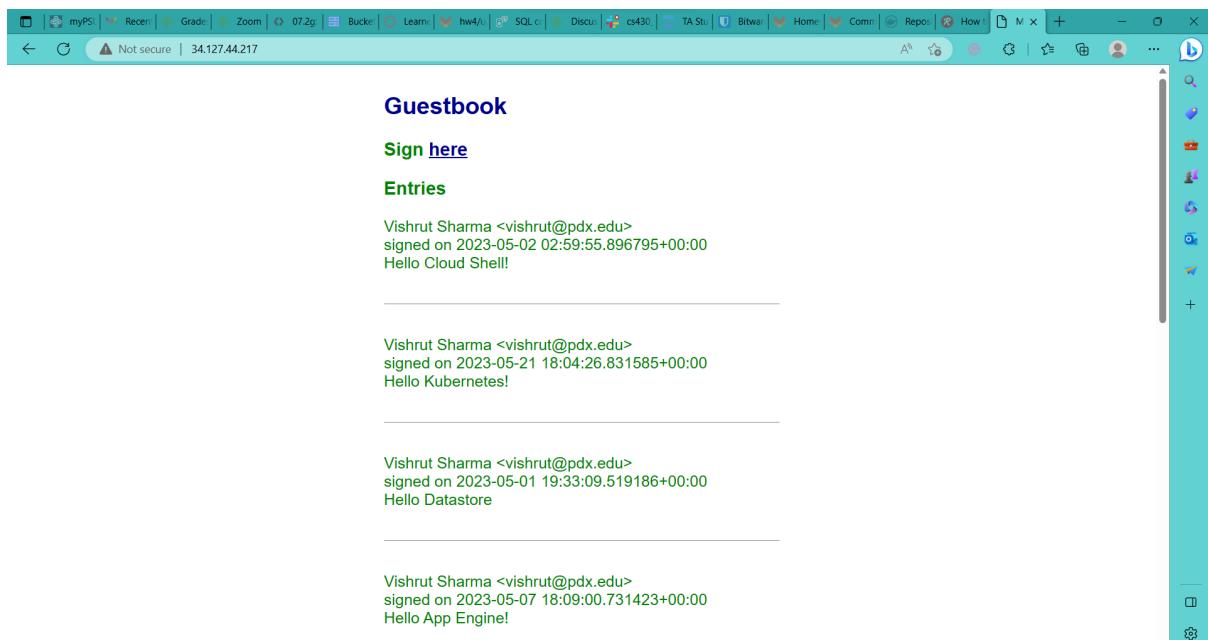
- Take a screenshot of listing services with LoadBalancer indicating an external IP address that is ready for access.

```
vishrut@cloudshell:~/cs430-src/05_gcp_datastore (cloud-sharma-vishrut)$ kubectl get services
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP     PORT(S)        AGE
guestbook-lb   LoadBalancer   10.80.14.73  35.203.142.67  80:31582/TCP  2m6s
kubernetes     ClusterIP    10.80.0.1    <none>          443/TCP       15m
vishrut@cloudshell:~/cs430-src/05_gcp_datastore (cloud-sharma-vishrut)$
```

8. View the Guestbook

- Take a screenshot of the Guestbook including the URL with the entry in it.

The External IP of the load balancer here is different because the previous IP had not worked and I reran the entire process after fixing the issue.



- Take a screenshot of the managed guestbook pods and the service being exposed.

Managed pods

Name	Status	Restarts	Created on
guestbook-replicas-52r24	Running	0	May 21, 2023, 11:02:10 AM
guestbook-replicas-qkkn6	Running	0	May 21, 2023, 11:02:10 AM
guestbook-replicas-s82dw	Running	0	May 21, 2023, 11:02:10 AM

Odin ID vi: •

File Edit View

Odin ID: vishrut|

Ln 1, Col 17 | 100% | Windows (CRLF) | UTF-8

- Take a screenshot of the load balancer and its details

The screenshot shows the Google Cloud Network services Load balancer details page. A modal dialog at the bottom center displays the message "The cluster was deleted successfully." The main content area shows the configuration for a load balancer named "acd8ecc1d09b4460cbe47a372ef654aa". It includes sections for Frontend (Protocol: TCP, IP version: IPv4, IP:Port: 34.127.44.217:80, Network Tier: Premium) and Backend (Name: acd8ecc1d09b4460cbe47a372ef654aa, Region: us-west1, Health check: k8s-3f7fc0a06bb89520-node). Below these, an "ADVANCED CONFIGURATIONS" section lists two instances: "gke-guestbook-default-pool-8bfd7852-4nx" and "gke-guestbook-default-pool-8bfd7852-lrvk", both in the "us-west1-a" zone with IP addresses 34.127.44.217 and 34.127.44.218 respectively.

- Take a screenshot of the addresses allocated and indicate the ones associated with nodes versus the one associated with the load balancer.

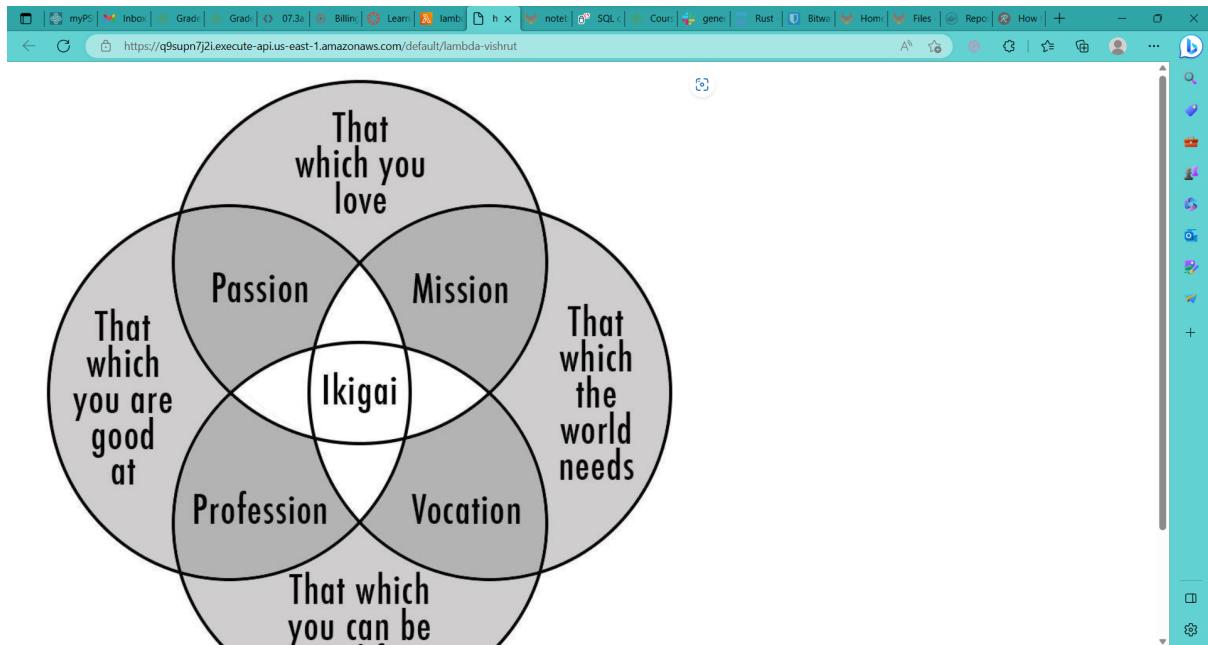
The screenshot shows the Google Cloud VPC network IP addresses list page. The left sidebar shows options like VPC networks, IP addresses, Bring your own IP, Firewall, Routes, VPC network peering, Shared VPC, Serverless VPC access, and Packet mirroring. The main table lists external IP addresses, with columns for Name, IP address, Access type, Region, Type, Version, and In use by. Three entries are listed: "34.82.190.21" (External, us-west1, Ephemeral, IPv4, VM instance gke-gue pool-8bfd7852-lrvk), "34.127.44.217" (External, us-west1, Ephemeral, IPv4, Forwarding rule acd8ecc1d09b4460cbe47a372ef654aa), and "34.168.204.242" (External, us-west1, Ephemeral, IPv4, VM instance gke-gue pool-8bfd7852-4nx). A modal dialog on the right side titled "Select an address" is open, showing a list of labels: "cost_center:sales" and "env:prod". A message at the bottom of the dialog says "No addresses selected."

34.82.190.21 and 34.168.204.242 are associated with nodes and 34.127.44.217 is associated with the load balancer.

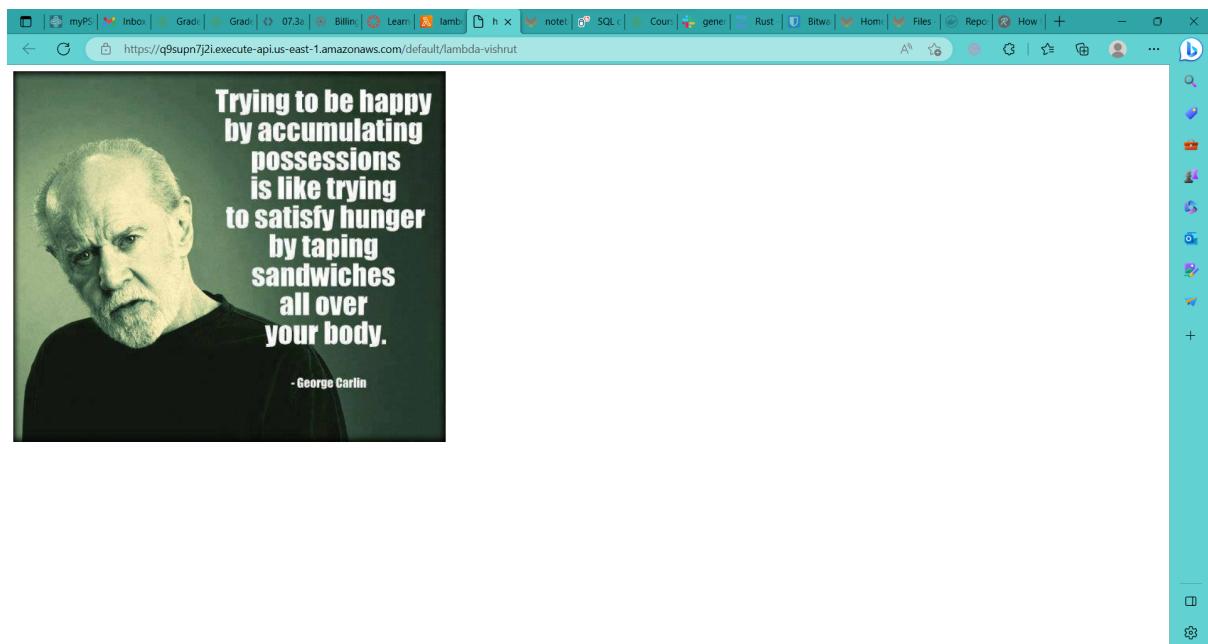
07.3a: APIs

4. Test code

- Take a screenshot of the resulting page including the URL bar.



- Click "Reload" in the browser and take another screenshot showing the image has changed:



8. Test code

- Use curl on your Linux VM to access the API endpoint and show the results.
Take a screenshot for your lab notebook.

```
vboxuser@vishrut:~$ curl https://i1i344npса.execute-api.us-east-1.amazonaws.com/default/gettime-vishrut
{"currentTime": "2023-05-15 16:34:00.139063"}vboxuser@vishrut:~$
```

07.3g: APIs (Slack, Knowledge Graph)

2. Code

- Could we have used the API Discovery package to interact with the Vision API?

Answer: Yes, we can use the API discovery package to interact with the Vision API.

- Does Google provide a Python package specifically for accessing the Knowledge Graph API?

Answer: No, Google does not provide a Python package specifically for accessing the Knowledge Graph API, but it can be accessed by using the Google API Client Library for Python.

3. Code

- Show the source line that constructs the query we wish to send to the Knowledge Graph API.

Answer: Line 86

The terminal window shows the following Python code:

```
83
84 # [START functions_slack_request]
85 def make_search_request(query):
86     req = kgsearch.entities().search(query=query, limit=1)
87     res = req.execute()
88     return format_slack_message(query, res)
89 # [END functions_slack_request]
90
91
```

To the right of the terminal is a screenshot of a Slack message window titled "Odin ID v". The message content is "Odin ID: vishrut|".

- Show the source line that then executes the query and saves the response. What is the name of the method that sends the query to the Knowledge Graph API?

Answer: Line 87 executes and saves the response. "make_search_request" sends the query to the Knowledge Graph API.

The terminal window shows the same Python code as before:

```
83
84 # [START functions_slack_request]
85 def make_search_request(query):
86     req = kgsearch.entities().search(query=query, limit=1)
87     res = req.execute()
88     return format_slack_message(query, res)
89 # [END functions_slack_request]
90
91
```

To the right of the terminal is a screenshot of a Slack message window titled "Odin ID v". The message content is "Odin ID: vishrut|".

- What is the Python data type that is used to represent the formatted message?

Answer: Text data type.

- What are the three main attributes of the formatted message passed back to Slack?

Answer: Title, title_link, image_url.

8. Test the command

- Take a screenshot of its response for your lab notebook.

The screenshot shows a Slack interface. On the left, there's a sidebar with 'vishrut-workspace' and a list of channels like '# general', '# iwc-lab', '# random', etc. The '# iwc-lab' channel is selected and highlighted in blue. In the main pane, there are two messages from 'vishrut': one at 9:54 AM joining the channel and another at 10:03 AM sending '/kg chatgpt'. Below these, a message from 'cs530bot' at 10:03 AM provides a detailed description of ChatGPT. The message content is as follows:

```

ChatGPT: Software
ChatGPT is an artificial intelligence chatbot developed by OpenAI and released in November 2022. It is built on top of OpenAI's GPT-3.5 and GPT-4 foundational large language models and has been fine-tuned using both supervised and reinforcement learning techniques.
  
```

At the bottom of the message, there's a link to 'Download Slack for Windows' and a 'Skip all tips' button.

07.4a: Lambda, API Gateway Guestbook

10. Deploy API to production and view entries

- Take a screenshot that shows that you can view the entries in the backend database.

The screenshot shows a Firefox browser window with the URL 'file:///home/vboxuser/cs430-src/06_aws_restapi_lambda/frontend/src/index.html'. The page is titled 'Guestbook' and contains a form with fields for 'Name', 'Email', and 'Message', along with a 'Sign' button. Below the form is a section titled 'Entries' showing two entries:

- Vishrut Sharma <vishrut@pdx.edu>
signed on 2023-05-01 18:46:40.532753
Hello DynamoDB
- Vishrut Sharma <vishrut@pdx.edu>
signed on 2023-05-02 01:58:50.004092
Hello Docker DynamonDB
- Vishrut Sharma <vishrut@pdx.edu>
signed on 2023-05-02 23:04:48.488903
Hello Cloud9!

12. API endpoint for signing (2)

- Take a screenshot showing that the submission worked.

The screenshot shows the AWS API Gateway Method Execution interface. On the left, the navigation pane shows the API named "vishruth-gb-res...". The "Resources" section is selected, and under it, the path "/entry" is expanded to show "POST" as the selected method. The main panel displays the "Method Execution /entry - POST - Method Test" page. It includes fields for "Path", "Request: /entry", "Status: 200", "Latency: 403 ms", and a "Response Body" containing a JSON array of messages. The "Headers" and "Stage Variables" sections are also present.

16. Configure and Deploy the Frontend

- Take a screenshot as before that shows your entry and the static website hosting URL.

The screenshot shows a web browser displaying four separate signed messages, each with a unique timestamp and environment identifier. The messages are:

- Message 1: Signed on 2023-05-07 02:33:38.164476, Hello Elastic Beanstalk!
- Message 2: Signed on 2023-05-08 00:23:49.041902, Hello ECS!
- Message 3: Signed on 2023-05-15 22:46:17.051042, Hello API Gateway
- Message 4: Signed on 2023-05-15 22:50:39.814862, Hello API Gateway from local HTML

Message 5: Signed on 2023-05-15 23:23:00.004964, Hello S3, API Gateway, Lambda!

07.4g: Cloud Functions API Gateway Guestbook

10. Create the API Gateway

- Include the hostname for the API gateway in your lab notebook.

Answer: defaultHostname: gbapigw-a7j9nwjh.uc.gateway.dev

11. Test the API via Python Requests (GET)

```
vishrutm@cloudshell:~/cs430-wk06_gcp_restapi_cloudfunctions (cloud-sharma-vishrutm)$ python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import requests
>>> resp=requests.get("https://gbapigw-a7j9nwjh.uc.gateway.dev/entries")
>>> print(resp.status_code)
200
>>> print(resp.headers)
Content-Type: application/json
Access-Control-Allow-Origin: *
Function-Execution-Id: j1dm48vhai
X-Cloud-Trace-Context: e1bc5beb141bd9ffdb339f7e71ee6e5;+o=1
Alt-Svc: h3="443"; ma=2592000,h3-29="443"; ma=2592000
Date: Sun, 21 May 2023 16:59:59 GMT
Server: Google Frontend
Content-Length: 800
>>> print(resp.text)
[{"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 02:59:55.896795+00:00", "message": "Hello Cloud Shell!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-07 18:09:00.731423+00:00", "message": "Hello App Engine!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 03:11:08.008194+00:00", "message": "Hello Compute Engine!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-11 21:16:08.697577+00:00", "message": "Hello Docker Datastore!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-01 19:33:09.5191864+00:00", "message": "Hello Datastore!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-07 18:09:00.731423+00:00", "message": "Hello App Engine!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 03:11:08.008194+00:00", "message": "Hello Compute Engine!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-11 21:16:08.697577+00:00", "message": "Hello Docker Datastore!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 02:52:01.845069+00:00", "message": "Hello Cloud Run!!"}]
>>> print(type(resp.json()))
<class 'list'>
>>> 
```

- Take a screenshot of the loop and its output

```
>>> for attribute, value in first_entry.items():
...     print(attribute,": ", value)
...
name : Vishrutm Sharma
email : vishrutm@pdx.edu
date : 2023-05-02 02:59:55.896795+00:00
message : Hello Cloud Shell!
>>> 
```

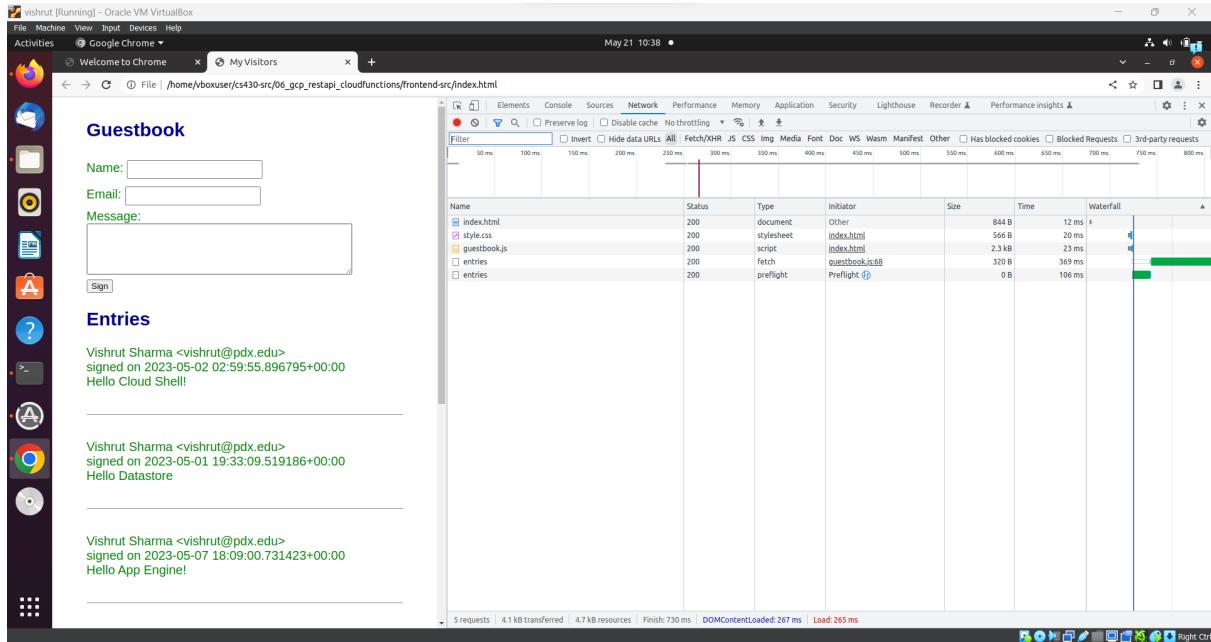
12. Test the API via Python Requests (POST)

- Take a screenshot of the output for your lab notebook

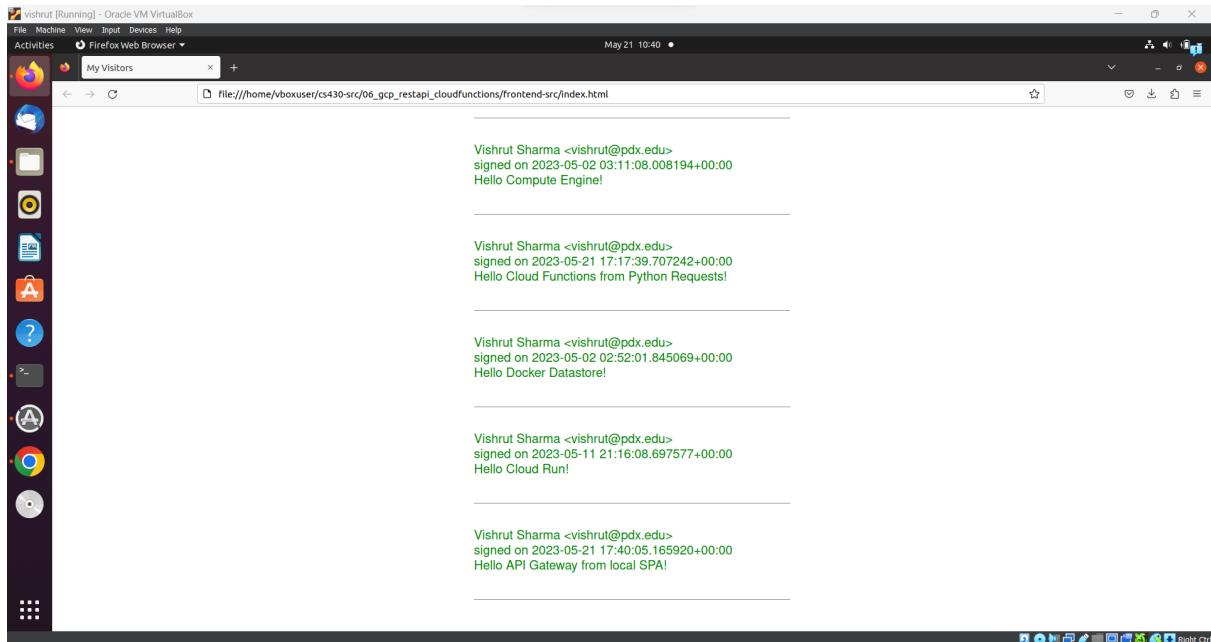
```
>>> mydict={
...     "name": "Vishrutm Sharma",
...     "email": "vishrutm@pdx.edu",
...     "message": "Hello Cloud Functions from Python Requests!!"
... }
>>> resp = requests.post('https://gbapigw-a7j9nwjh.uc.gateway.dev/entry', json=mydict)
>>> print(resp.status_code)
200
>>> print(resp.headers)
Content-Type: application/json
Access-Control-Allow-Origin: *
Function-Execution-Id: bab92d7kugvp
X-Cloud-Trace-Context: 19e1372b0e0c0a7c906b12d1380612f84;+o=1
Alt-Svc: h3="443"; ma=2592000,h3-29="443"; ma=2592000
Date: Sun, 21 May 2023 17:17:41 GMT
Server: Google Frontend
Content-Length: 956
>>> print(resp.text)
[{"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 02:59:55.896795+00:00", "message": "Hello Cloud Shell!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-07 18:09:00.731423+00:00", "message": "Hello App Engine!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 03:11:08.008194+00:00", "message": "Hello Compute Engine!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-11 21:17:39.707242+00:00", "message": "Hello Cloud Functions from Python Requests!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-02 02:52:01.845069+00:00", "message": "Hello Docker Datastore!!"}, {"name": "Vishrutm Sharma", "email": "vishrutm@pdx.edu", "date": "2023-05-11 21:16:08.697577+00:00", "message": "Hello Cloud Run!!"}]
>>> 
```

15. Version #1: Local file system

- Take a screenshot showing the preflight request to the API that allows API access, as well as the subsequent fetch request have been successful.



- Take a screenshot of the Guestbook including the URL.



16. Version #2: Google Cloud Storage bucket

- Take a screenshot of the Guestbook including the URL.

