Lab notebook Week 6

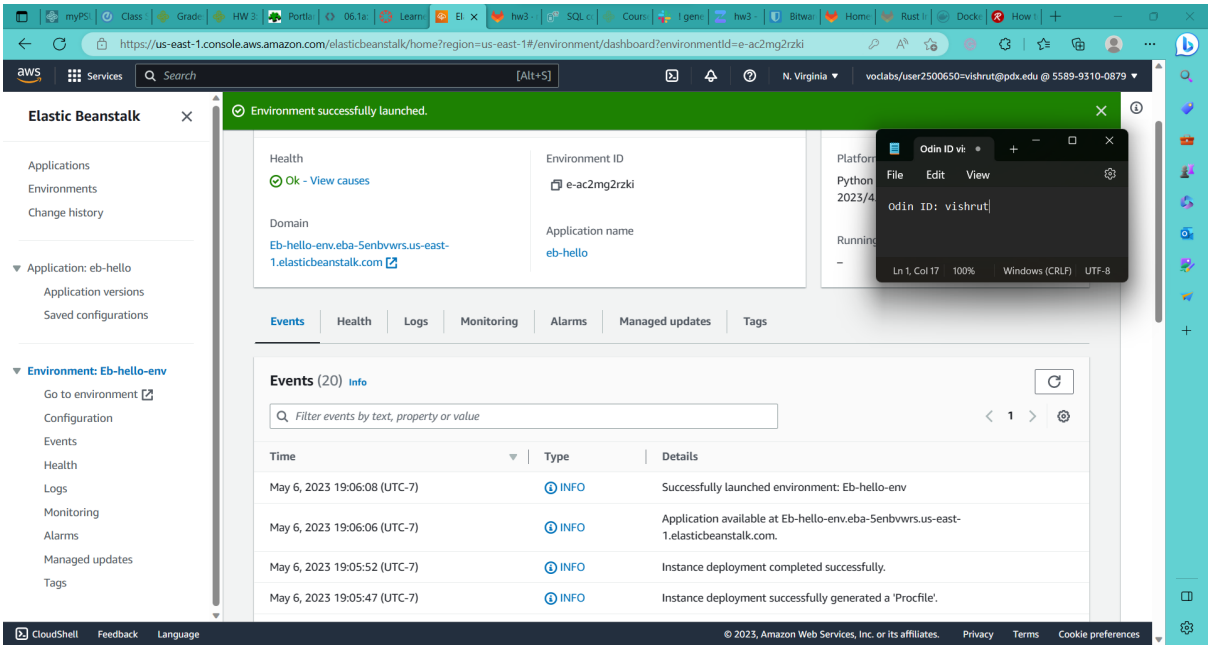
Submitted by: Vishrut Sharma (OdinID: vishrut)

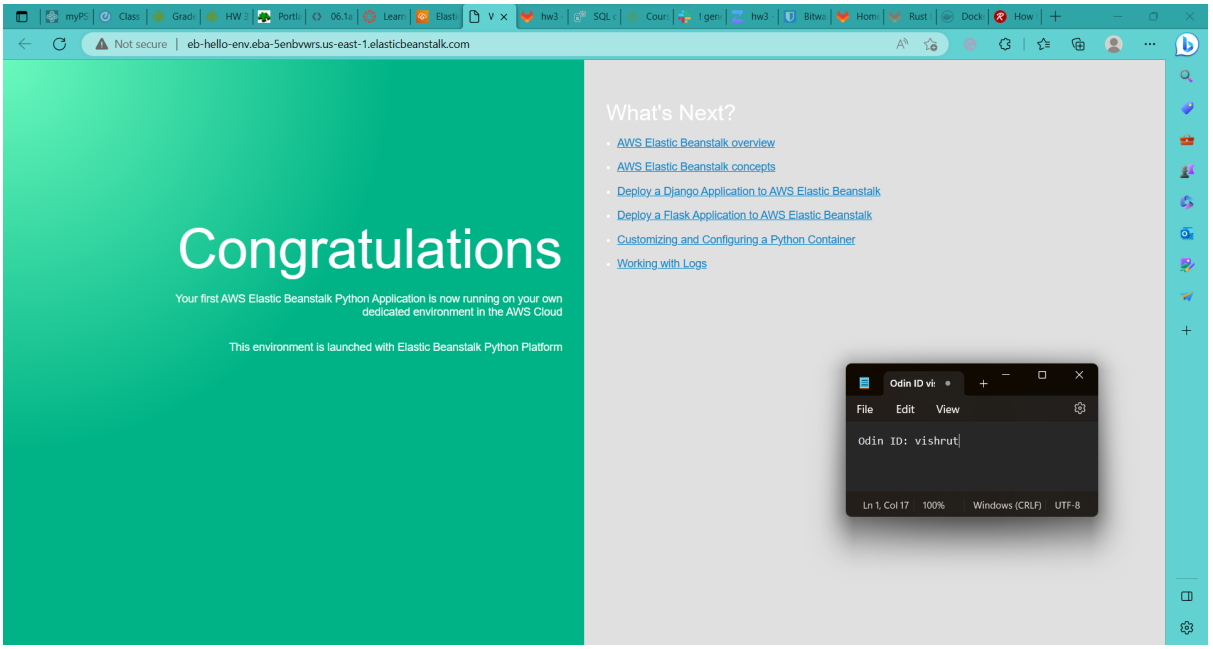
Table of Contents

06.1a: EB Guestbook

3. Running the application

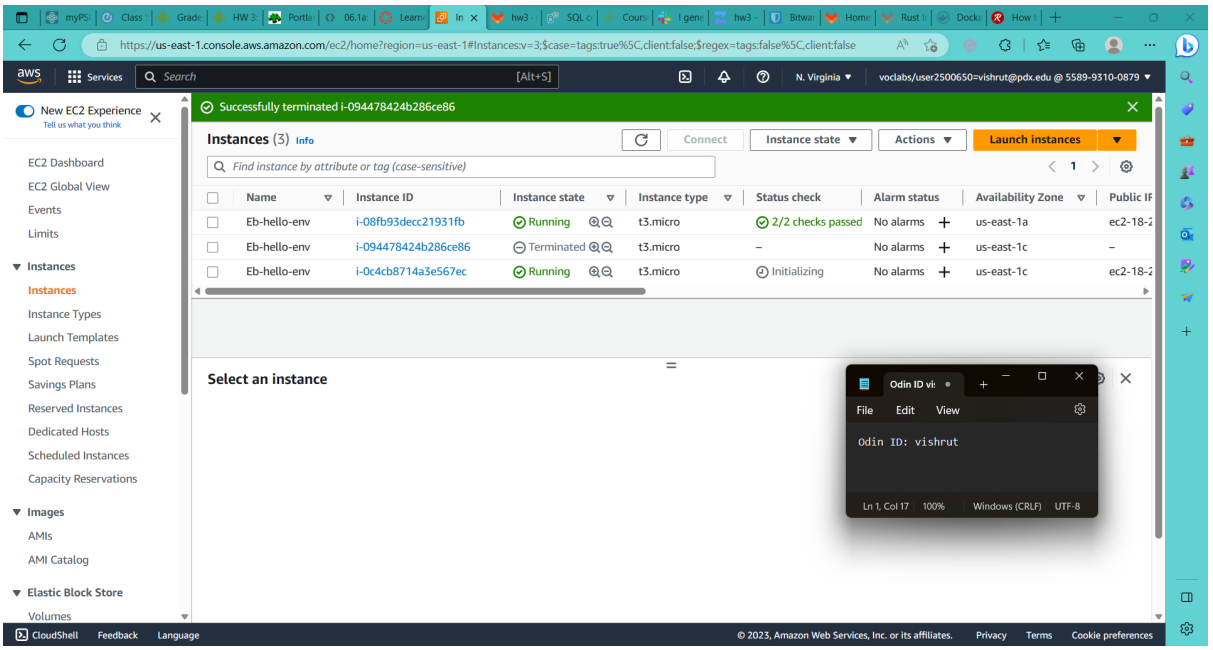
- **Take a screenshot showing it has been brought up successfully**





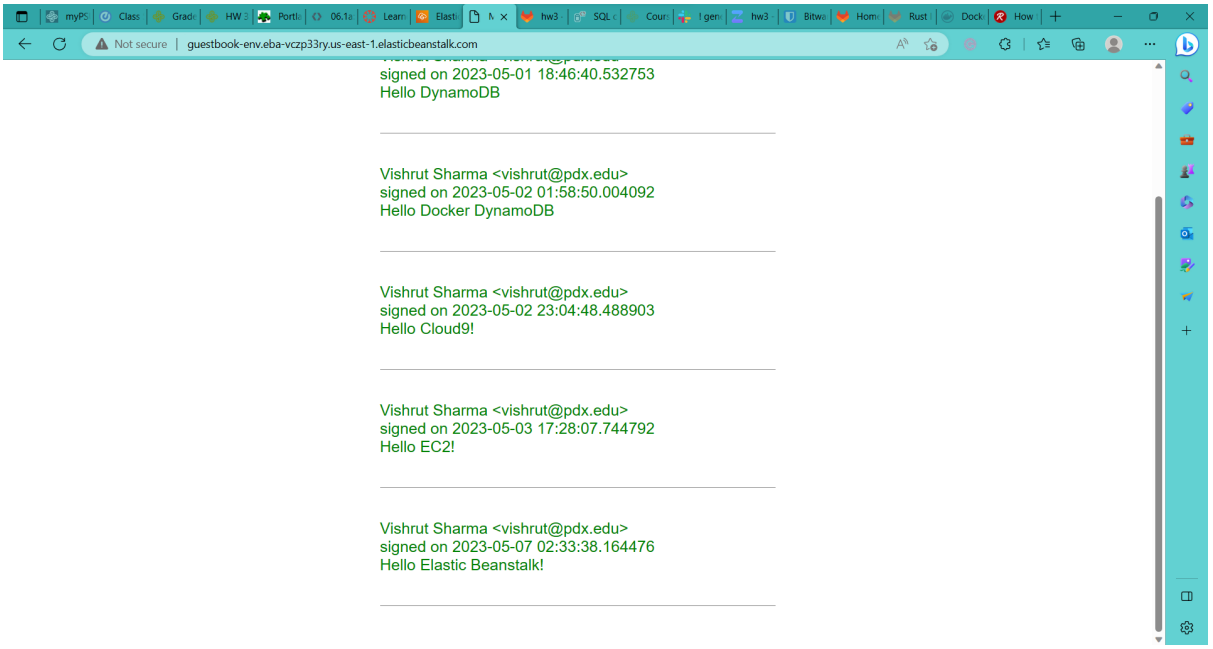
4. Handling failures seamlessly

- **Take a screenshot of the replacement VM being started.**

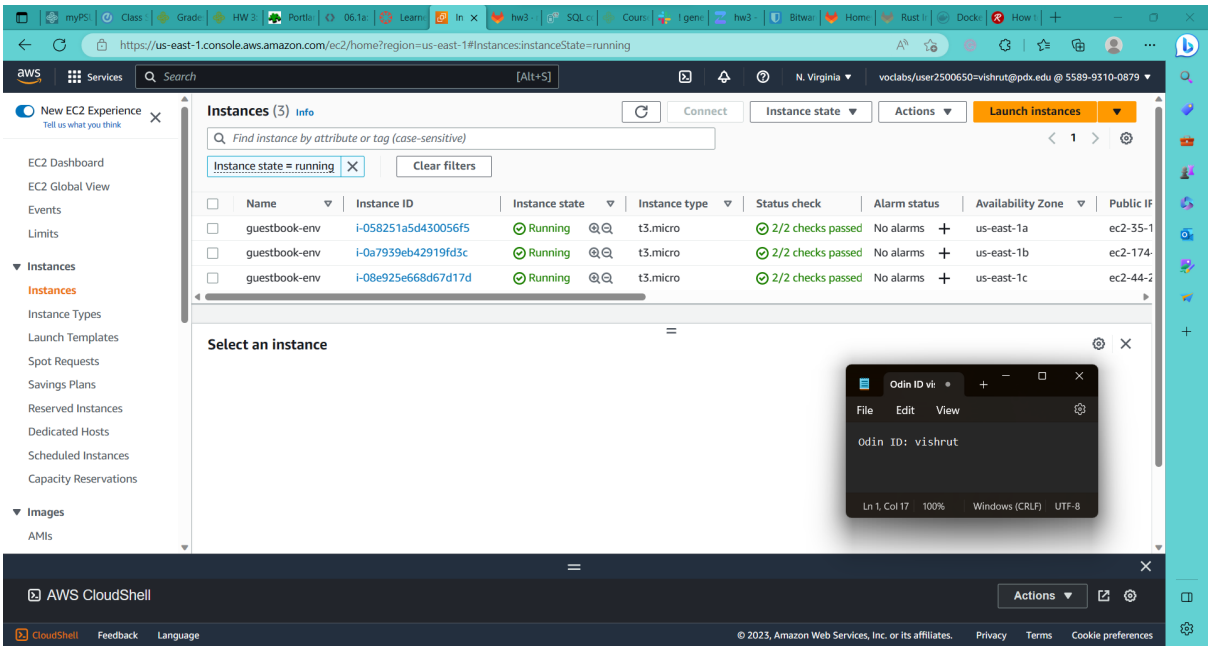


7. Deploying the Guestbook

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



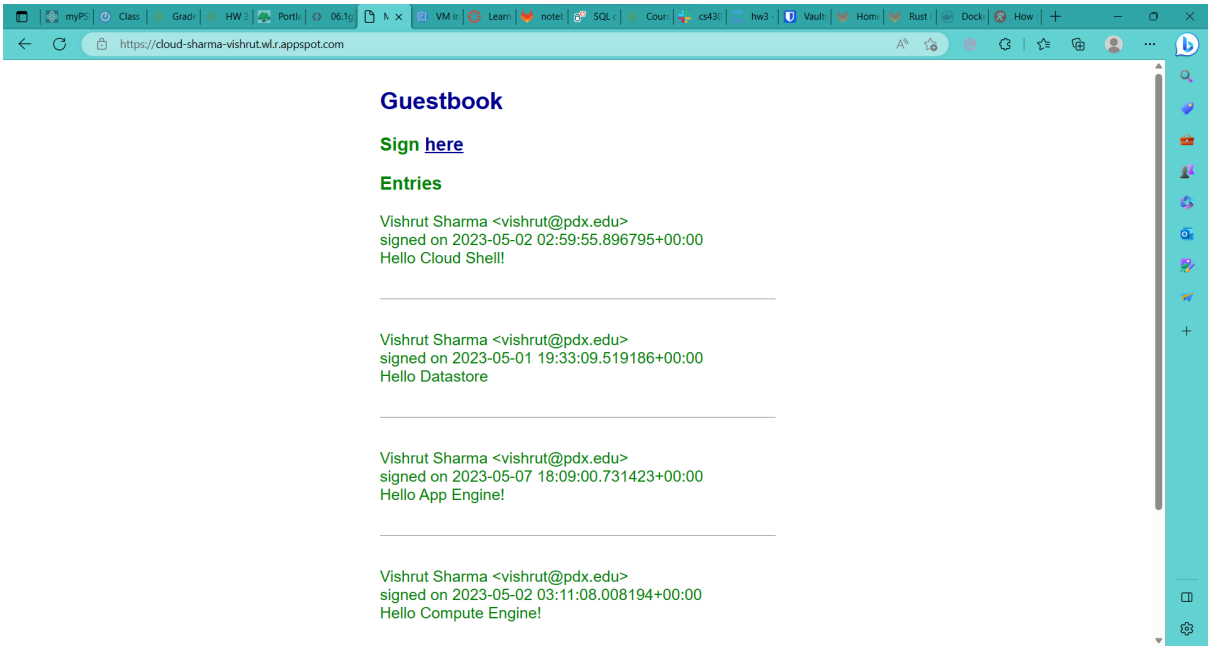
- **Take a screenshot of them.**



06.1g: App Engine Guestbook

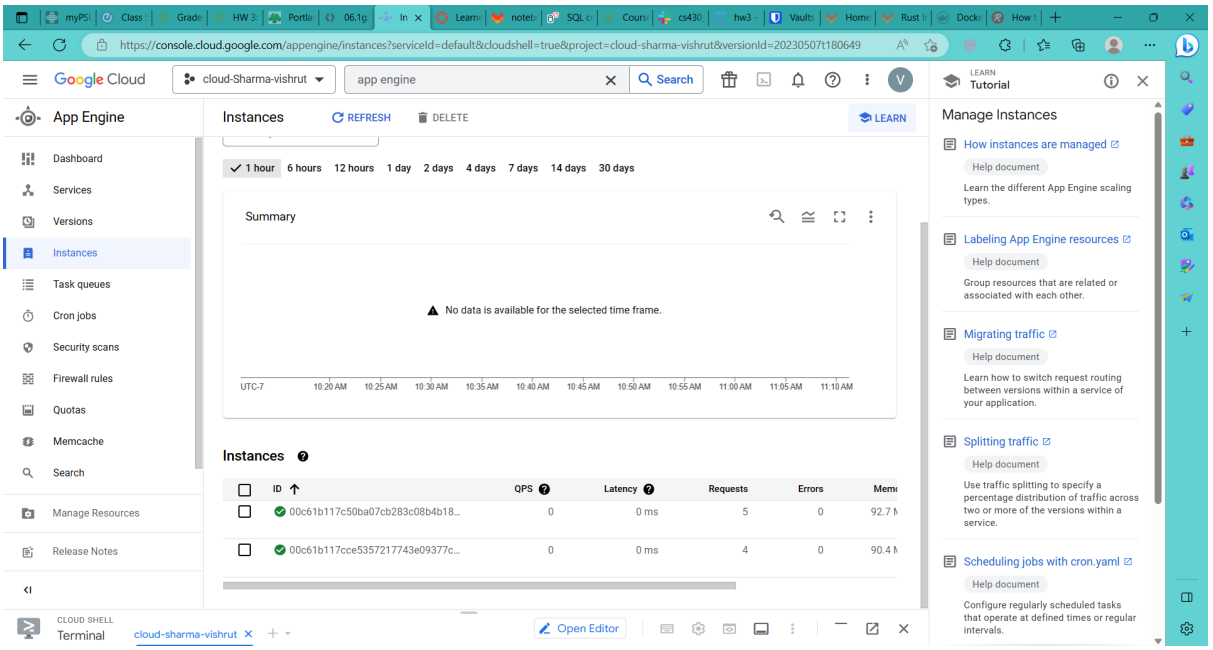
4. Deploying the Guestbook

- **Take a screenshot of the output that includes the URL in the address bar for your lab notebook.**



5. Handling failures seamlessly

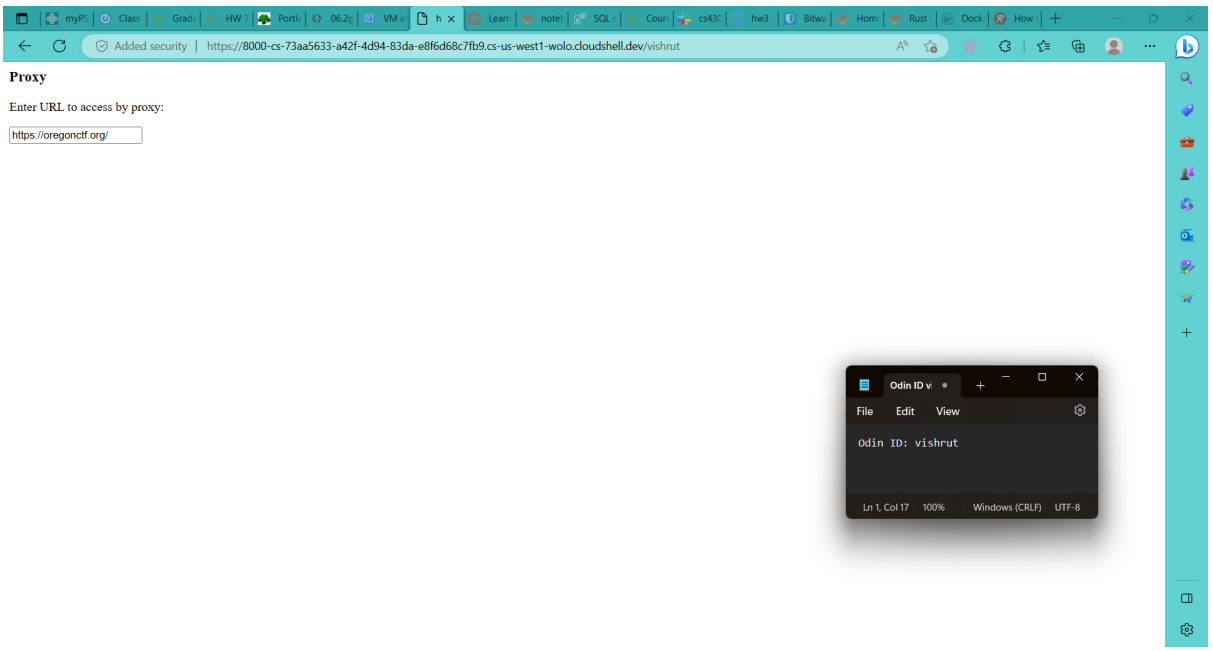
- **Take a screenshot of them.**

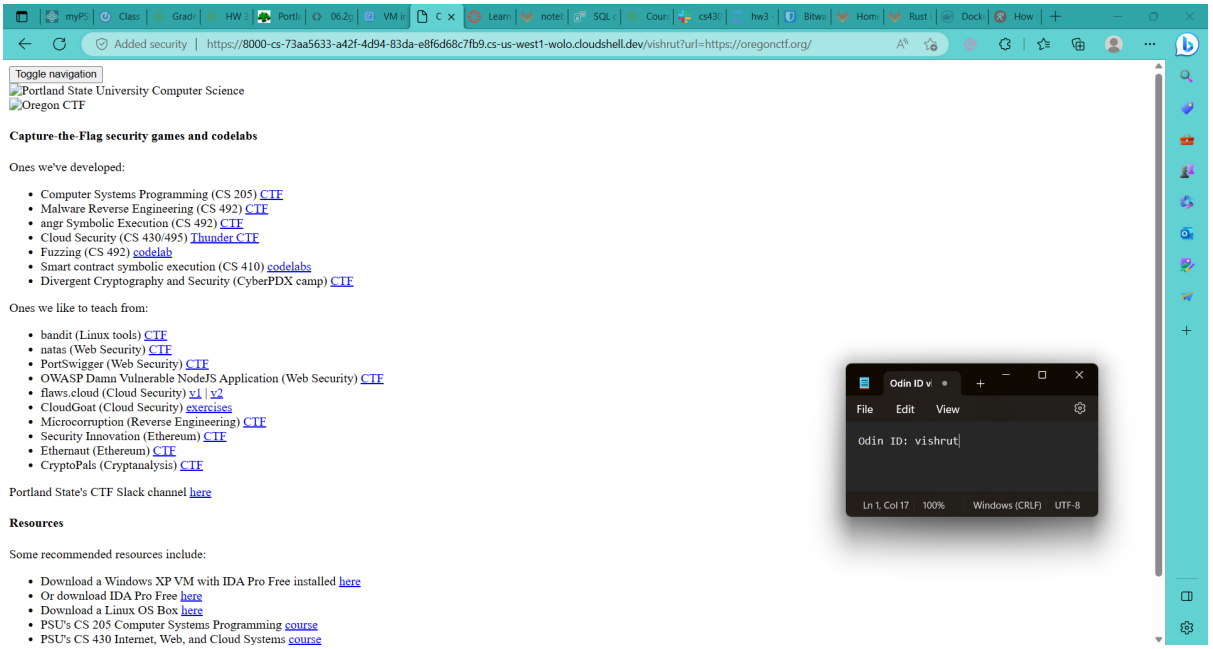


06.2g: Cloud Run, Secret Manager (Web proxy)

8. Setup secret proxy

- **Take a screenshot of the proxy and its results including the URL containing your OdinID**



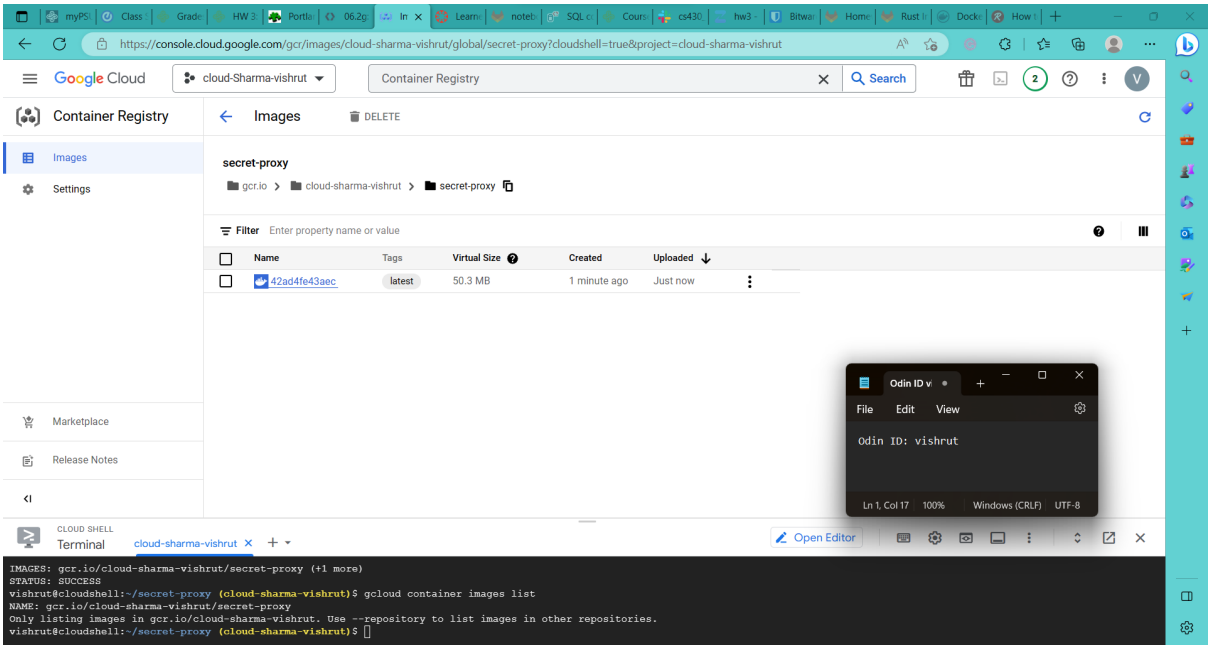


- **What is the security advantage of passing in the secret proxy route as an environment variable?**

Answer: Using the secret proxy route as an environment variable enhances security by separating sensitive information, enabling access control, simplifying configuration management, providing auditing capabilities, and integrating with cloud service providers.

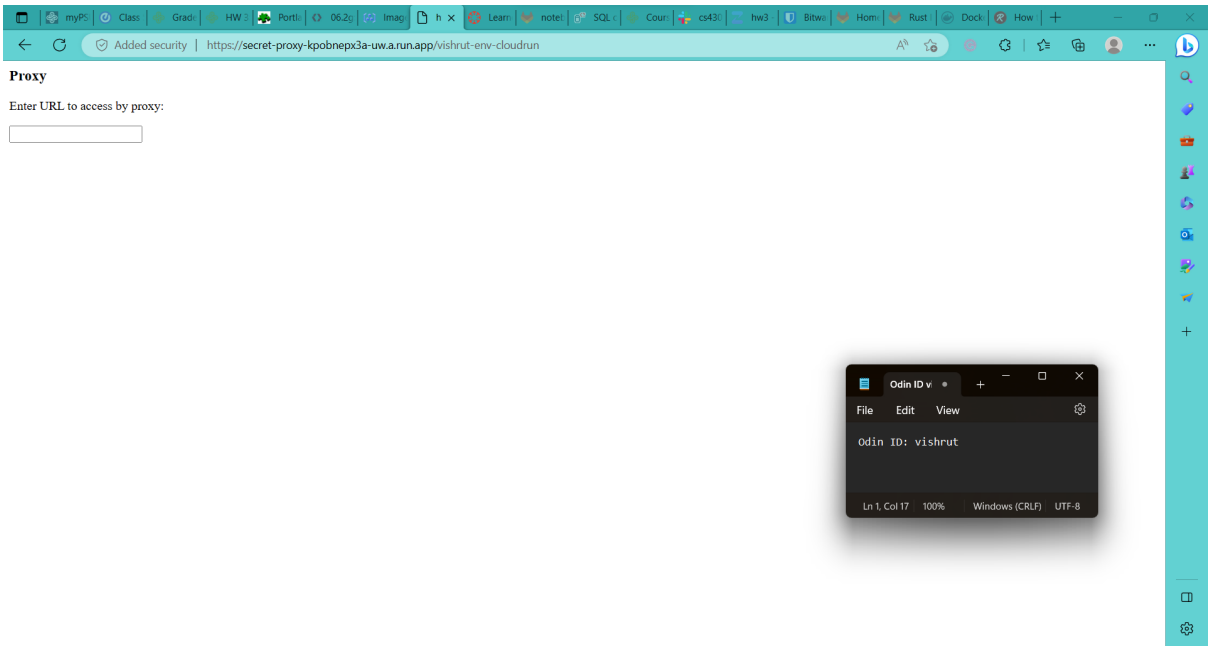
9. Cloud Build and Container Registry

- **Take a screenshot of the image in the registry that shows the size of the container for your lab notebook.**

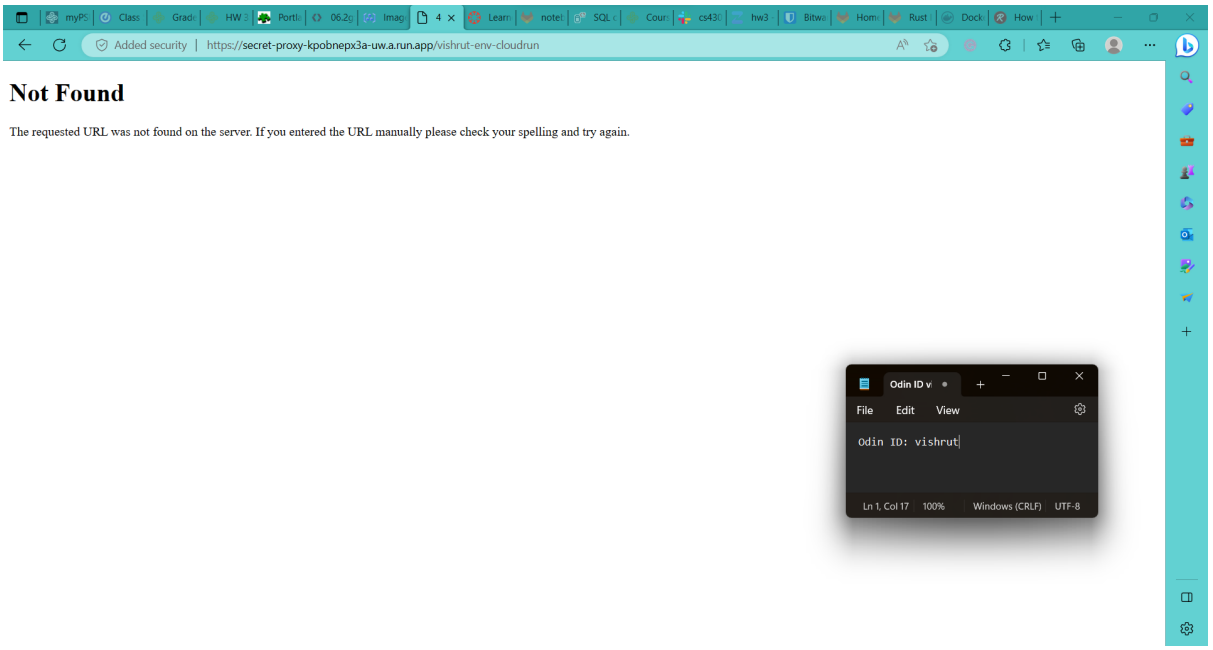


10. Deploy to Cloud Run

- **Take a screenshot of it that includes the proxy URL for your lab notebook.**

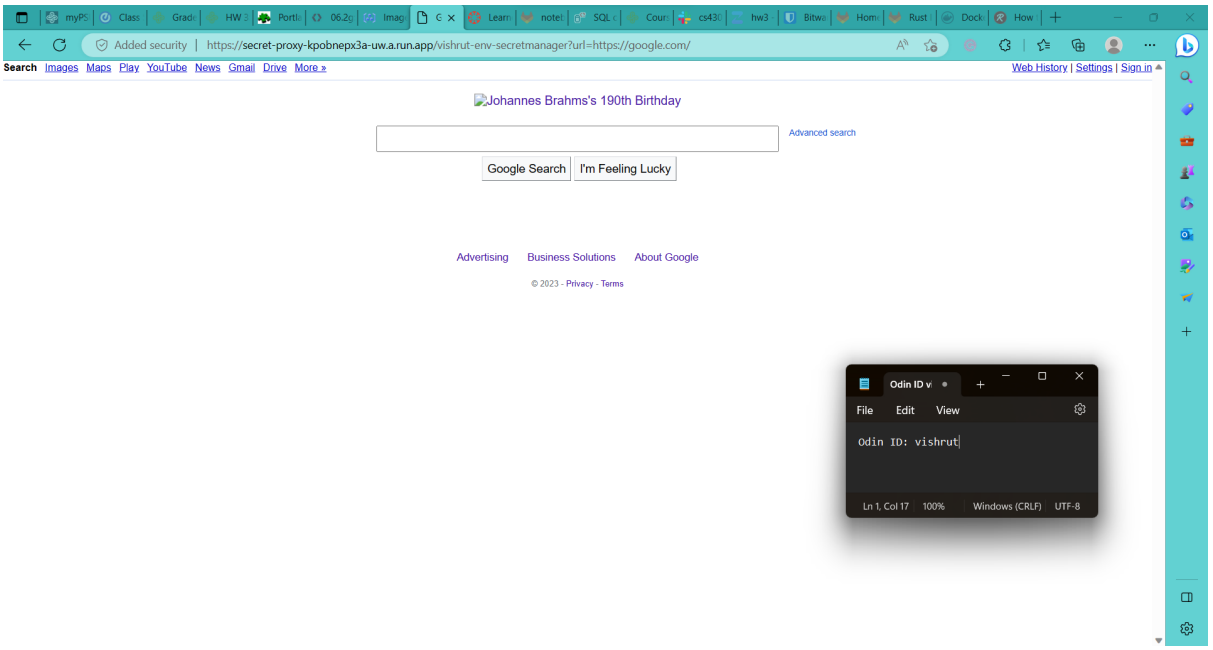


- **Take a screenshot of the error page that includes the proxy URL for your lab notebook.**



12. Deploy to Cloud Run with Secret Manager

- **Take a screenshot of it that includes the proxy URL for your lab notebook.**



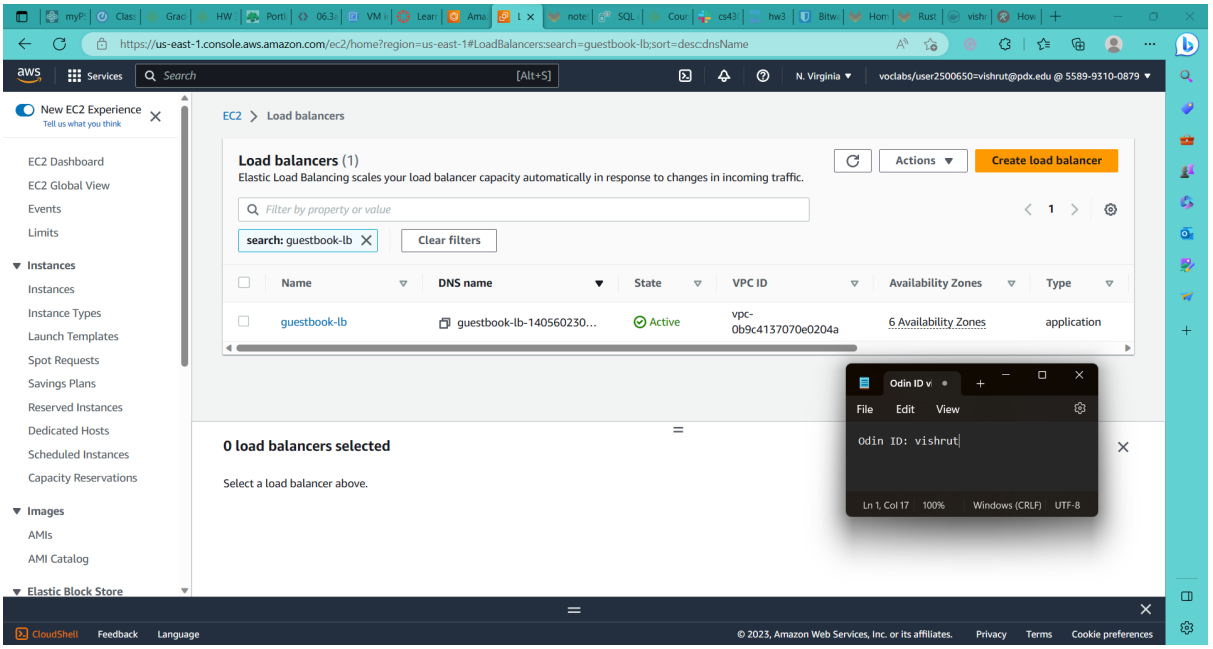
- **Identify the vulnerability in your lab notebook that Google has prevented.**

It seems that Google has taken measures to prevent a vulnerability that could potentially allow unauthorized access to sensitive information via the Metadata service linked to the virtual machine (VM) that runs a container. This service can be accessed through the two URLs provided, http://169.254.169.254/computeMetadata and http://169.254.169.254/computeMetadata/v1, and provides crucial information about the VM's identity and configuration, such as authentication tokens. If a container running on a VM gains access to this service, an attacker could exploit this vulnerability to obtain unauthorized access to the sensitive information stored on the VM or conduct other malicious actions.

06.3a: ECS Guestbook

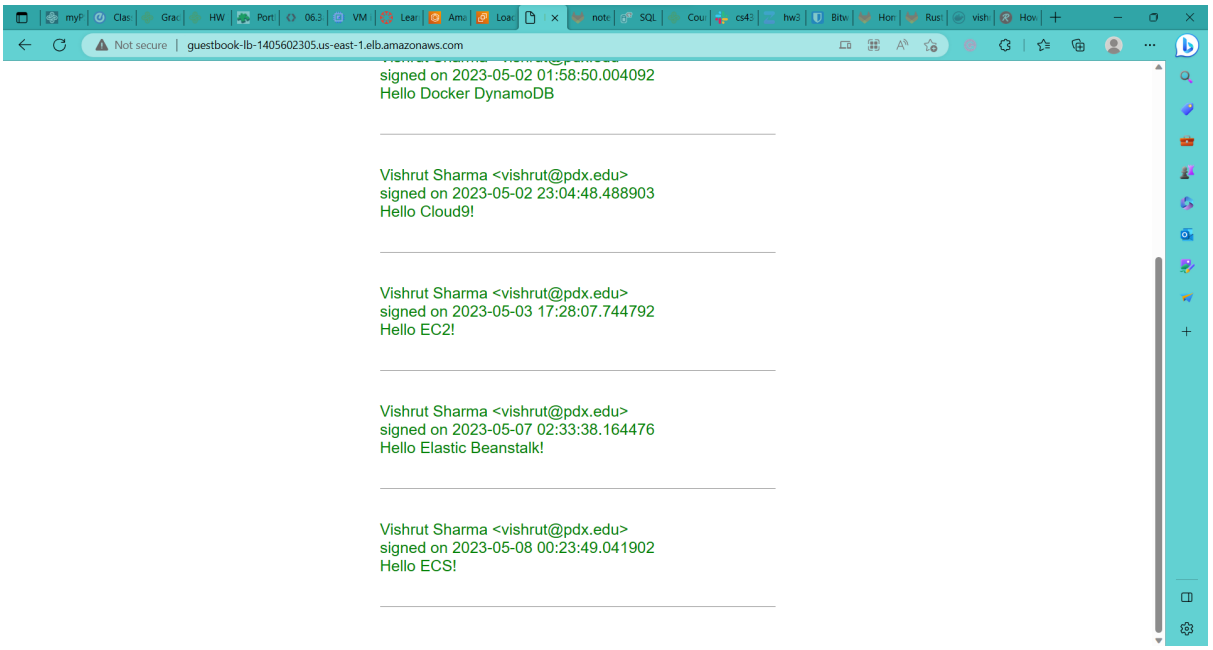
5. Examine the service

- **Take a screenshot of the DNS name of the guestbook-lb load balancer for your lab notebook**



6. Visit the site

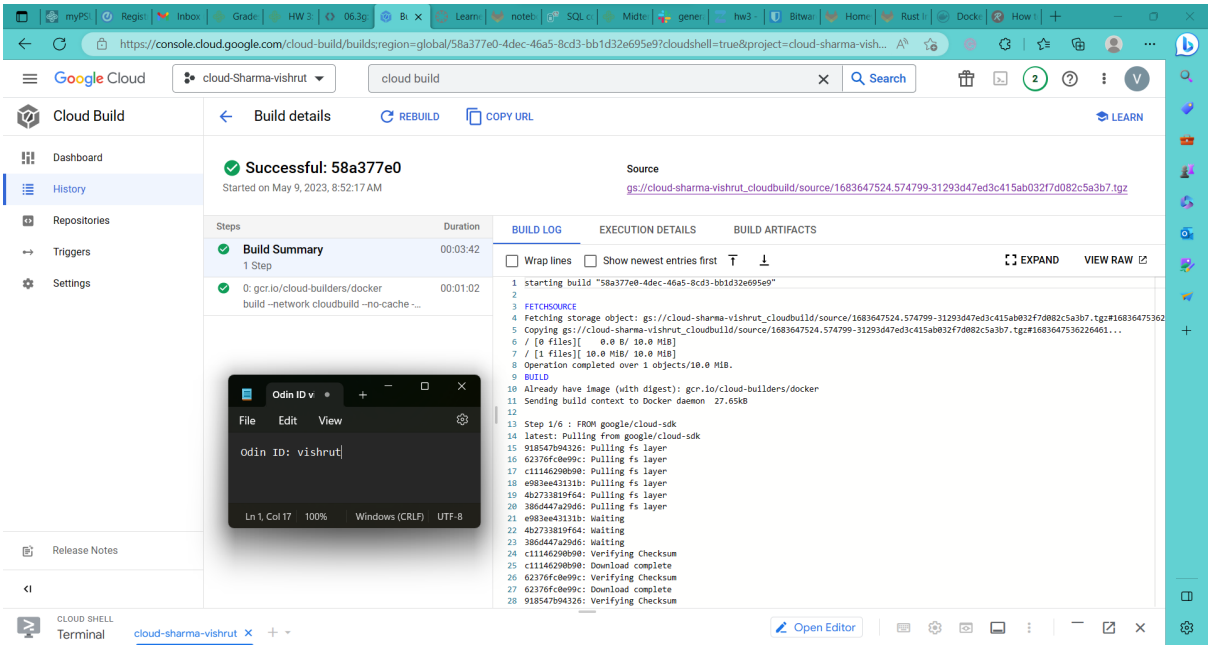
- **Take a screenshot of the Guestbook app running in a browser that includes the DNS name of the site.**



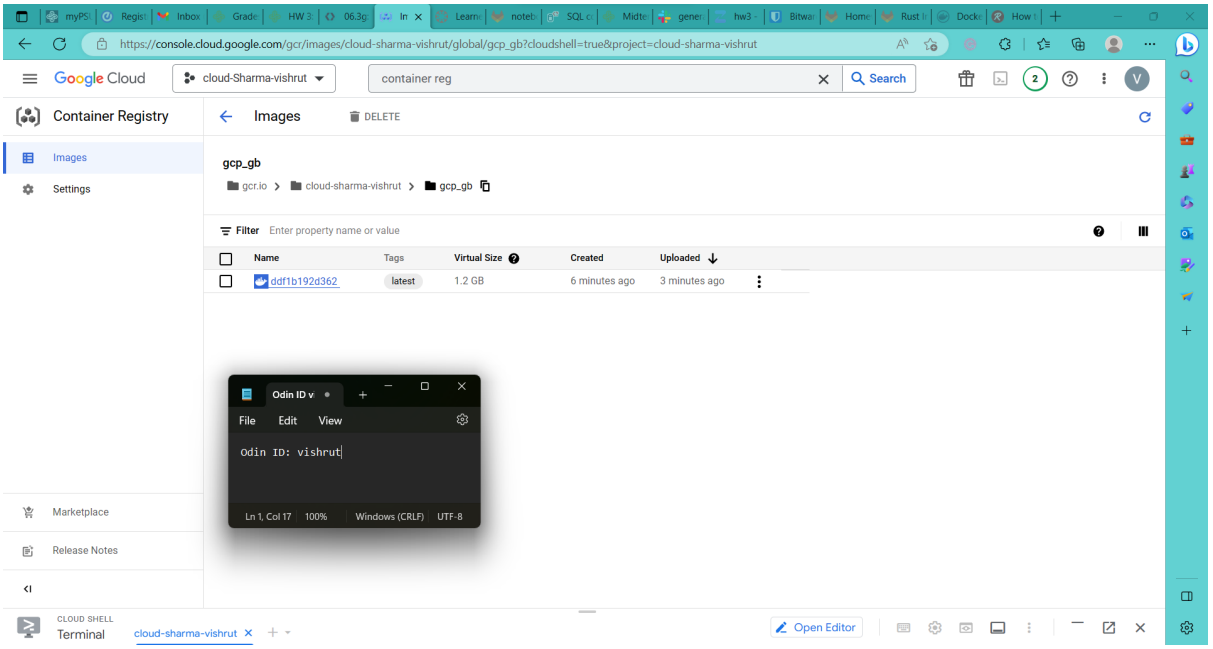
06.3g: Cloud Run Guestbook

2. Prepare a container image

- **Take a screenshot that includes the output of the command and the time it took to execute.**

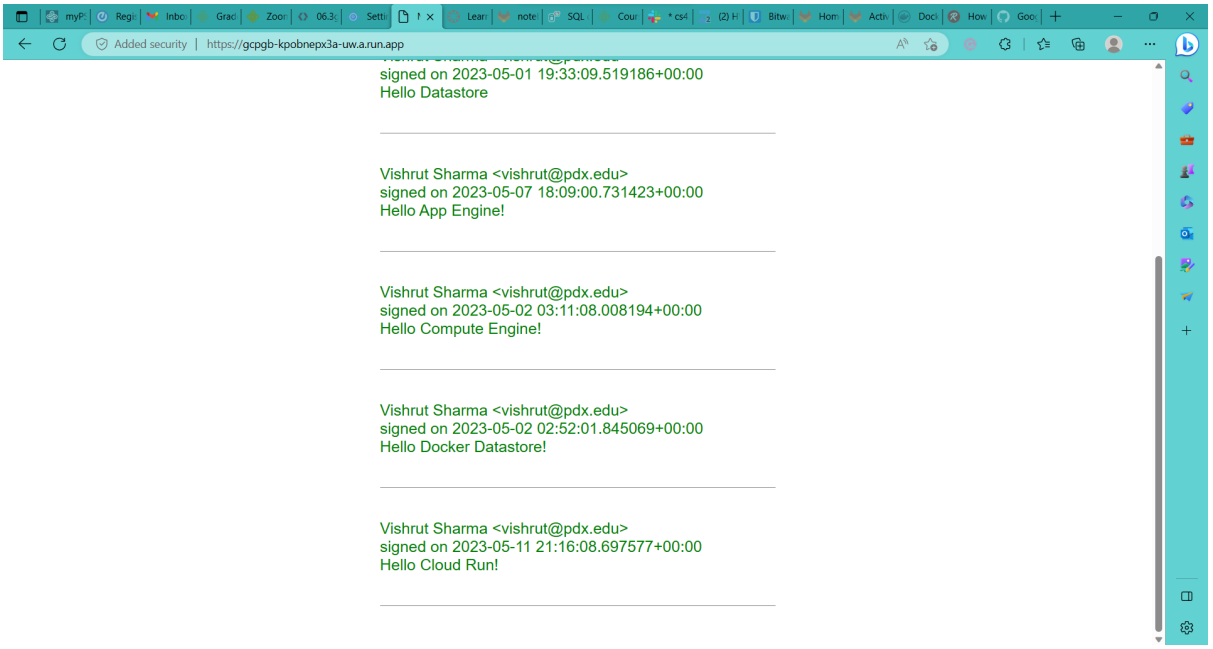


- **Take a screenshot showing the container image and its virtual size**



4. View the Guestbook

- **Take a screenshot that includes the URL Cloud Run has created for your site.**



- **What port do container instances listen on?**

Answer: 8080

- **What are the maximum number of instances Cloud Run will autoscale up to for your service?**

Answer: 100 instances

06.4g: Cloud Functions, PubSub

4. -

- **After downloading the file from the bucket, where is it stored?**

Answer: After downloading the file from the bucket the file is stored in temp_local_filename.

- **What class in the ImageMagick package is used to do the blurring of the file?**

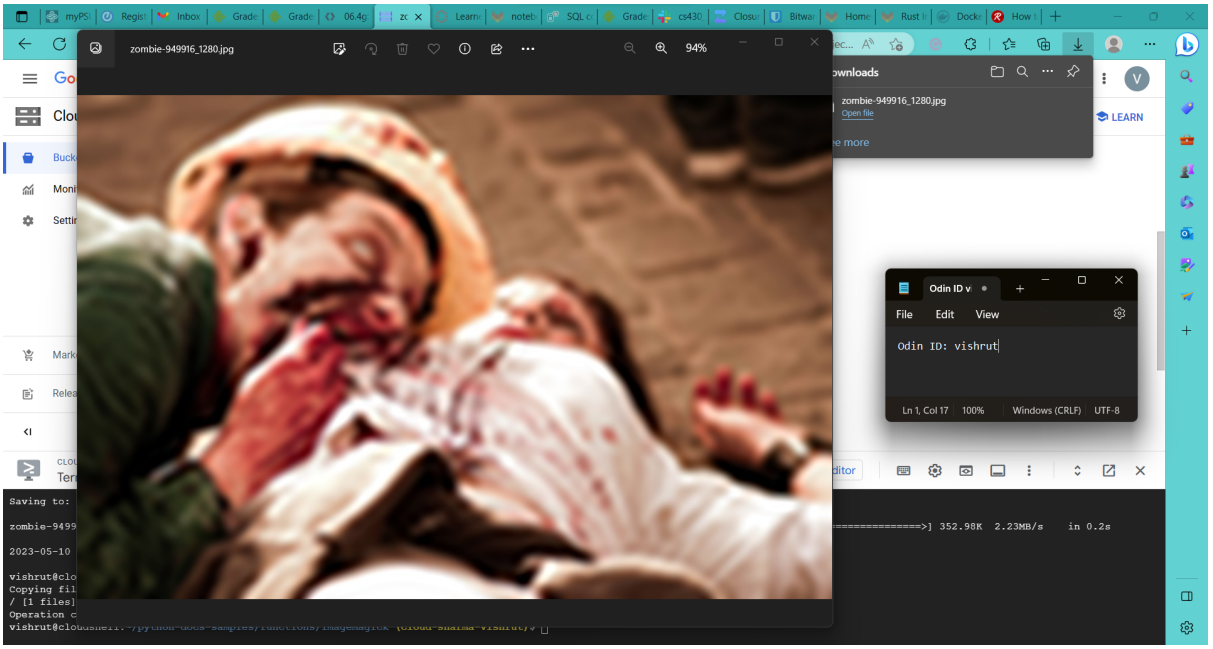
Answer: **__blur_image** class is used to do the blurring of the file.

- **What lines of code perform the blurring of the image and its storage back into the filesystem?**

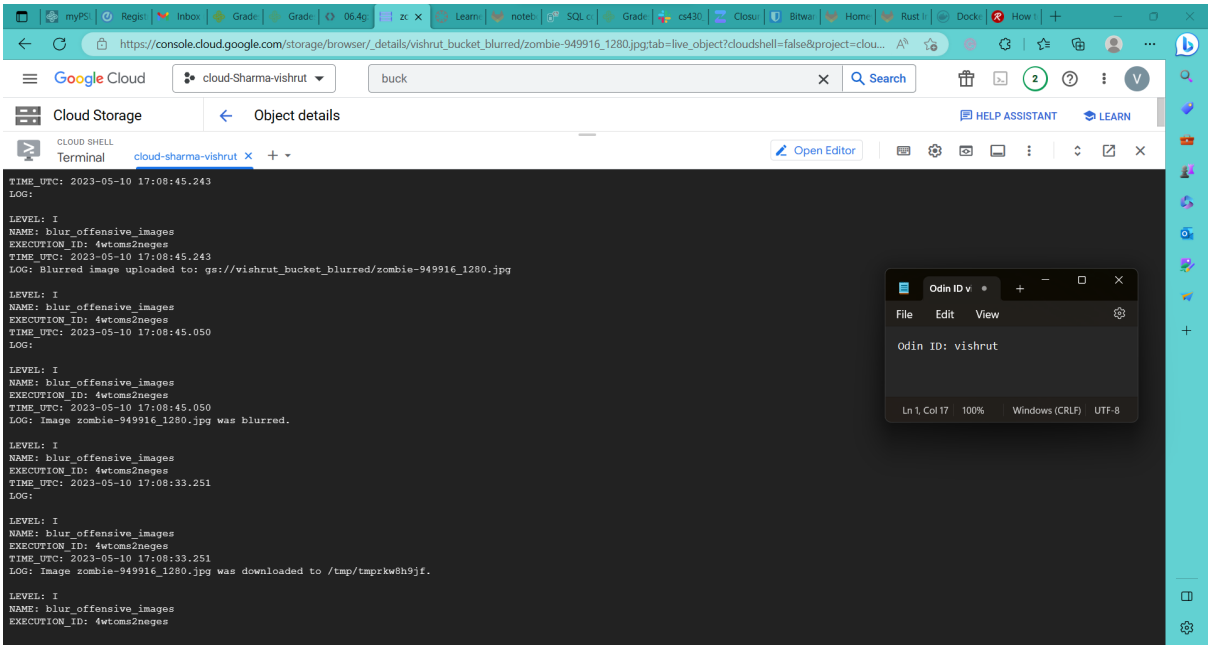
Answer: Lines 72 to 74 of the code are used to perform the blurring of the image and store it back in the filesystem.

7. Test function

- **Take a screenshot of the blurred image in the output bucket for your lab notebook**



- **Include a screenshot of the output logs that show that the above image was blurred.**



10. PubSub via CLI

- **Why are there no items returned?**

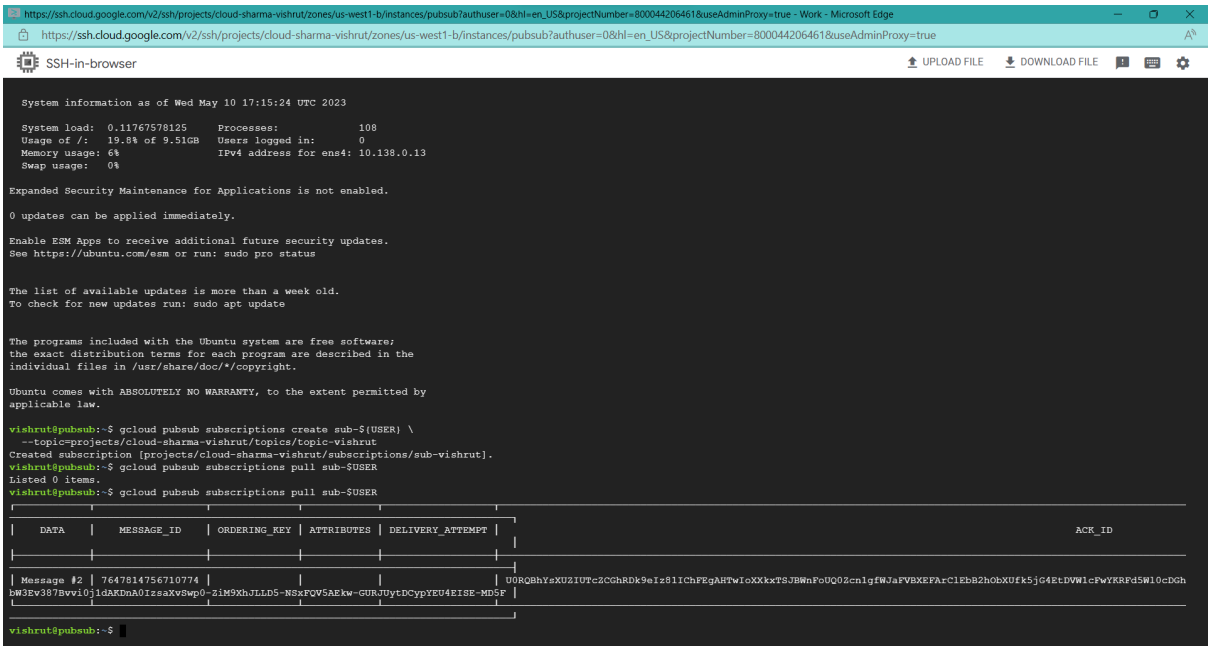
Answer: This is because we created the subscription after publishing message #1. When message #1 was published there were no subscribers to send the message to. When message #2 was published the subscription was already active which is why we were able to view it in the VM.

11. -

- **What is the messageId of the published message?**

```
vishrut@cloudshell:~ (cloud-sharma-vishrut)$ gcloud pubsub topics publish projects/cloud-sharma-vishrut/topics/topic-vishrut --message="Message #1"
messageIds:
- '7647814835072160'
vishrut@cloudshell:~ (cloud-sharma-vishrut)$ gcloud pubsub topics publish projects/cloud-sharma-vishrut/topics/topic-vishrut --message="Message #2"
messageIds:
- '7647814756710774'
vishrut@cloudshell:~ (cloud-sharma-vishrut)$
```

- **Take a screenshot of the output of the successful pull that includes the message and its messageId.**



14. Test programs and clean up

- **Take a screenshot showing the messageIds and messages sent**

```
(env) vishrut@cloudshell:~ (cloud-sharma-vishrut)$ python3 publisher.py
Enter a message to send: message 1
Published 7647939257604256 to topic projects/cloud-Sharma-vishrut/topics/my_topic
Enter a message to send: message 2
Published 7647912924268368 to topic projects/cloud-Sharma-vishrut/topics/my_topic
Enter a message to send: message 3
Published 7647964843866737 to topic projects/cloud-Sharma-vishrut/topics/my_topic
Enter a message to send:
```

- **Take a screenshot showing the same messageIds and messages received**