

## **What is cloud computing(CC)?**

In simple terms, it is like having your computer on the internet. Whatever you can do on your PC can be done on the cloud, unlocking a treasure chest of possibilities.

## **If cloud computing is the same as having a personal computer then why use cloud computing?**

The PC you use often will have 8 to 16 GB's of RAM , possibly more and a single processor. This works well for personal use, but fails when you have to scale and serve millions of users . In order to do that you would need a Chonky System , Powerful processors , High capacity Ram , High speed storage , GPU's etc. Well building it yourself would be good if you were a crypto bro and earning millions but realistically speaking it's not feasible for a single person to do and maintain it , hence we offload all our troubles to cloud providers which massively helps us .Along the course you will learn the advantages and disadvantages of using cloud services .

This document provides an overview of what can be achieved with cloud computing. Before diving into practical examples, we will first explore the different models of cloud services  
Prerequisite for this lab is a [github](#) account and some Coffee.

# Services available to us

## **Iaas Infrastructure as a service**

Here the cloud providers offer the infrastructure for you to work on. You will need to configure your resources such as ram, cpus, operating system [ maybe more ]. Basically you will get to customize your PC on the internet and use it however you want.  
A good example for that is an EC2 instance of AWS.

## **Paas Platform as a service**

Here the cloud providers hide away the hardware configuration and most of the other stuff, and just provide the service that you would want, for example you would need a database. Well, here you don't have to worry about allocating ram, disk space, or networking. You would need to worry about the schema of the tables, the constraints and stuff.

## **Saas Software as a service**

Here they directly provide you the service, no need to worry about configuring anything. Just directly use it. Good example would be your gmail, google drive etc. Sign up on

# Practice Time...

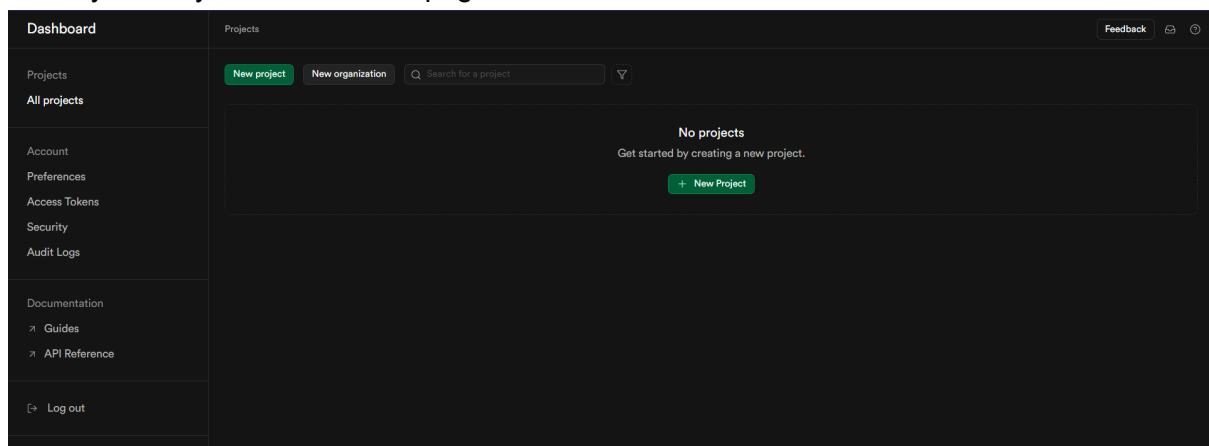
## A >> IAAS - Infrastructure Backend as a service

One of the widely used cloud services is object store, so unlike your traditional file system where data is stored as files and folders here you store data as blobs or objects, these objects are associated with a variable amount of metadata, and a globally unique identifier. Object storage systems allow retention of massive amounts of unstructured data in which data is written and read.

Here for object store we are going to use Supabase, supabase is open source baas (backend as a service), it provides a lot of free services such as Authentication, postgresql database, realtime database and object store.

1. Sign up for Supabase account <https://supabase.com/>

Once you do, you should see a page like this



2. When you create a new project for the first time it will ask you to create an organization, go ahead and do it under personal and free tier.

### Create a new organization

This is your organization within Supabase.  
For example, you can use the name of your company or department.

Name

PES1UGXXCS000


What's the name of your company or team?

Type of organization

Personal

What would best describe your organization?

Plan

Pricing 

Free - \$0/month

Cancel

You can rename your organization later

Create organization

3. While creating new project enter the database password [ ideally its is best to generate one and keep it safe], and select region as South Asia (Mumbai)

### Create a new project

Your project will have its own dedicated instance and full Postgres database.  
An API will be set up so you can easily interact with your new database.

Organization

PES1UGXXCS000

Project name

PES1UGXXCS000


Database Password

●●●●●●●●●●●●●●

Copy

This password is strong. [Generate a password](#)

Region

 South Asia (Mumbai)

Select the region closest to your users for the best performance.

SECURITY OPTIONS >

ADVANCED CONFIGURATION >

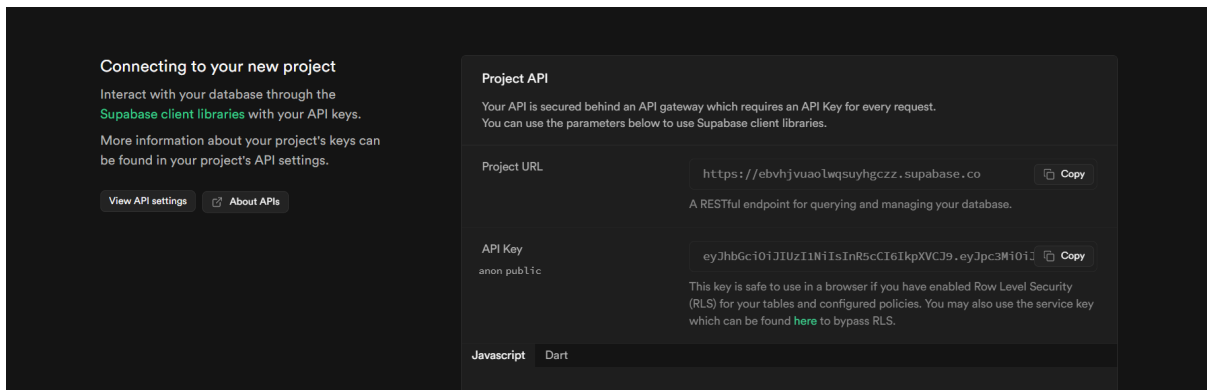
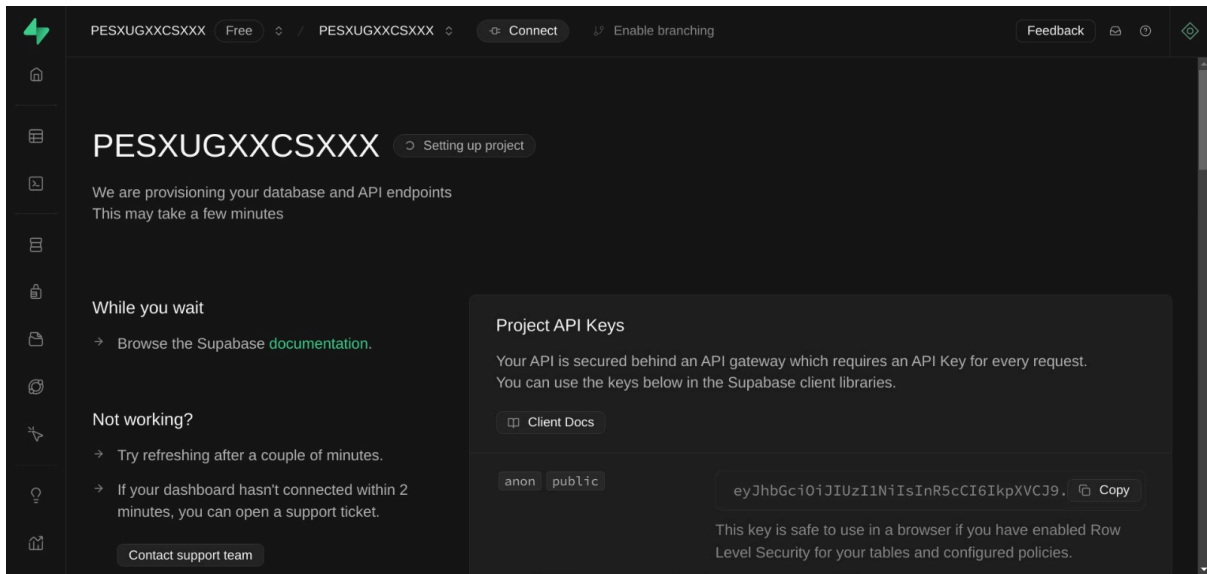
Cancel

You can rename your project later

Create new project

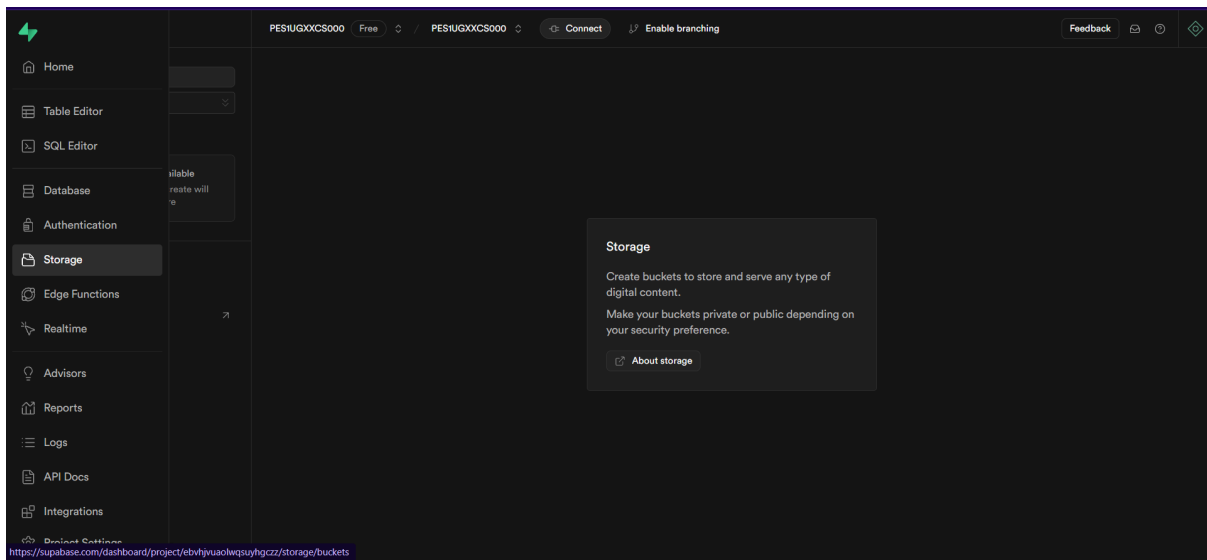
4. Once you create the new project your screen will display something similar to the first image below. This would be the first **Screenshot(SS1) required**.

Once you create the new project you will also get the Project api Keys and Project URL [ Scroll down a bit]. Copy those 2 and paste it somewhere that's easy to access.



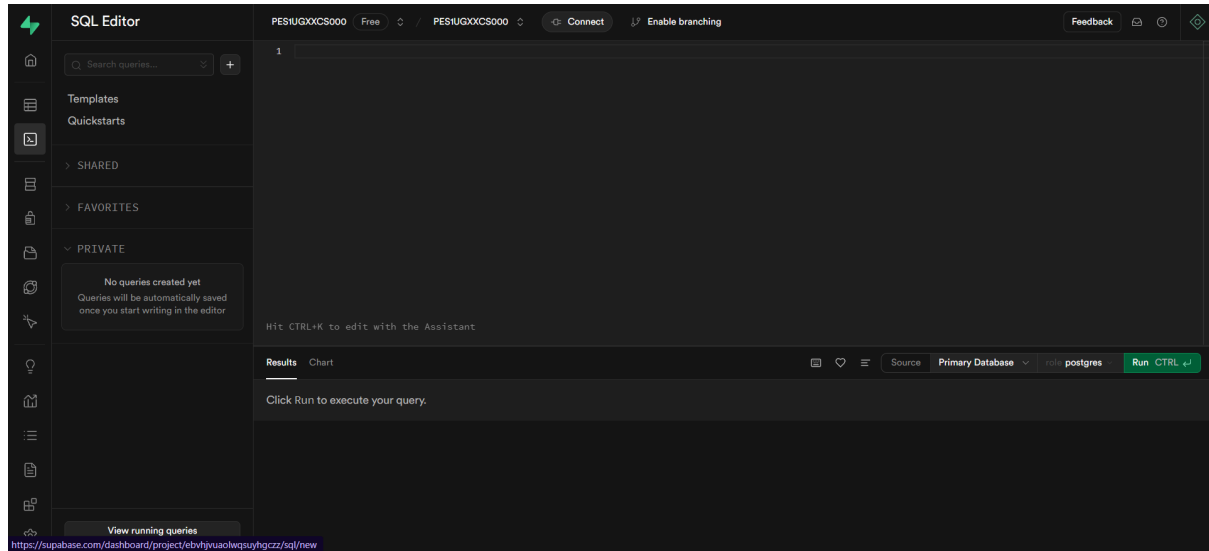
(You don't need to copy the shown api key and url, by the time you see this docs the project would have been deleted XD)

5. You can explore any services, for now we will focus on object store.



Well here you can create a bucket, and store data by uploading them by using the ui, but since that is not suitable for large projects, We will do it through automated script.

6. Before we jump to python script there is one thing we need to do, that is we need to tell supabase to allow api calls to allow operations on object storage. For that we need to move to SQL Editor.

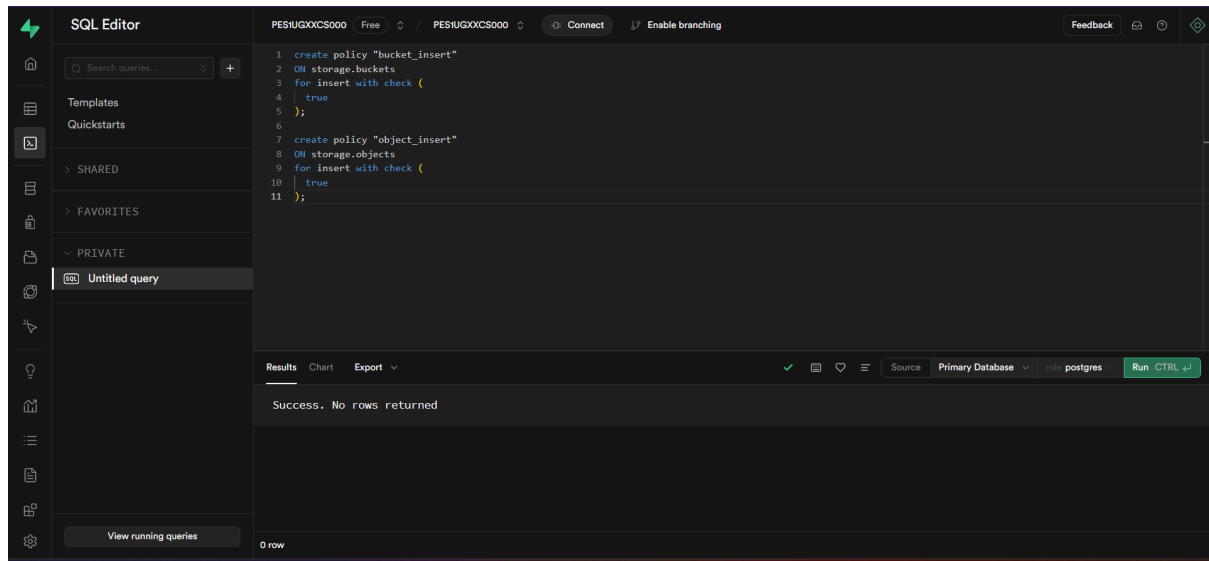


7. On this you need to run two sql queries. Once you click Run, take the **Screenshot**(SS2, Similar to the second image under this step) indicating the successful execution of those 2 queries.

```
create policy "bucket_insert"
ON storage.buckets
for insert with check (
    true
);

create policy "object_insert"
ON storage.objects
for insert with check (
    true
);
```

[Well you see most of the permission handling on supabase is done through postgresql database, these 2 queries modify the table to allow insert operations on object store by api calls.]



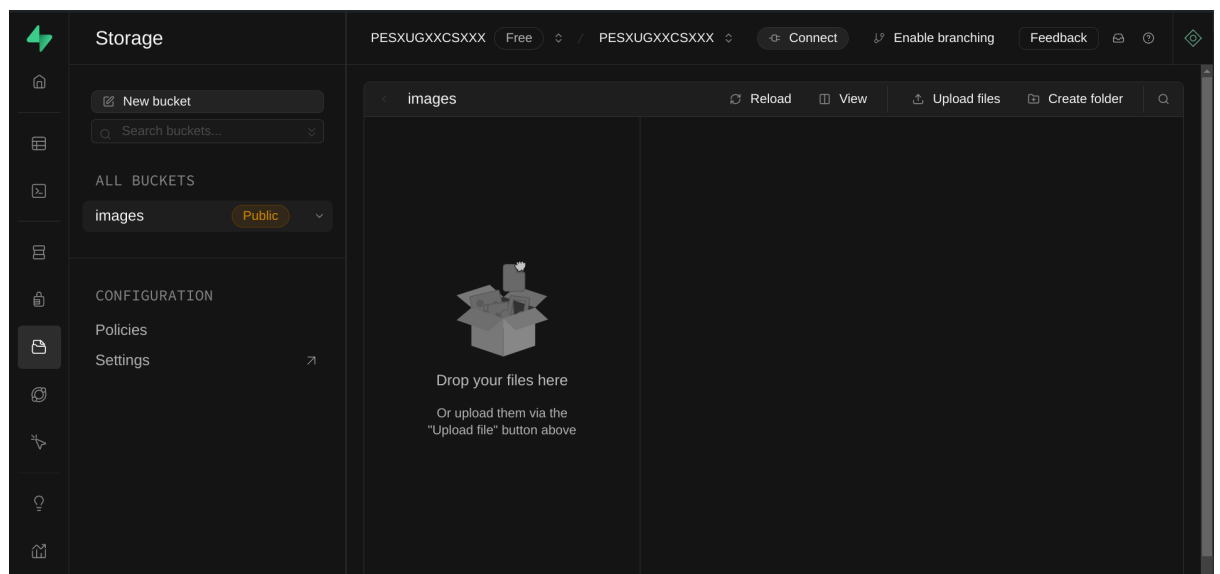
8. For the supabase api, you would need to install the supabase module for python.

```
pip install supabase
```

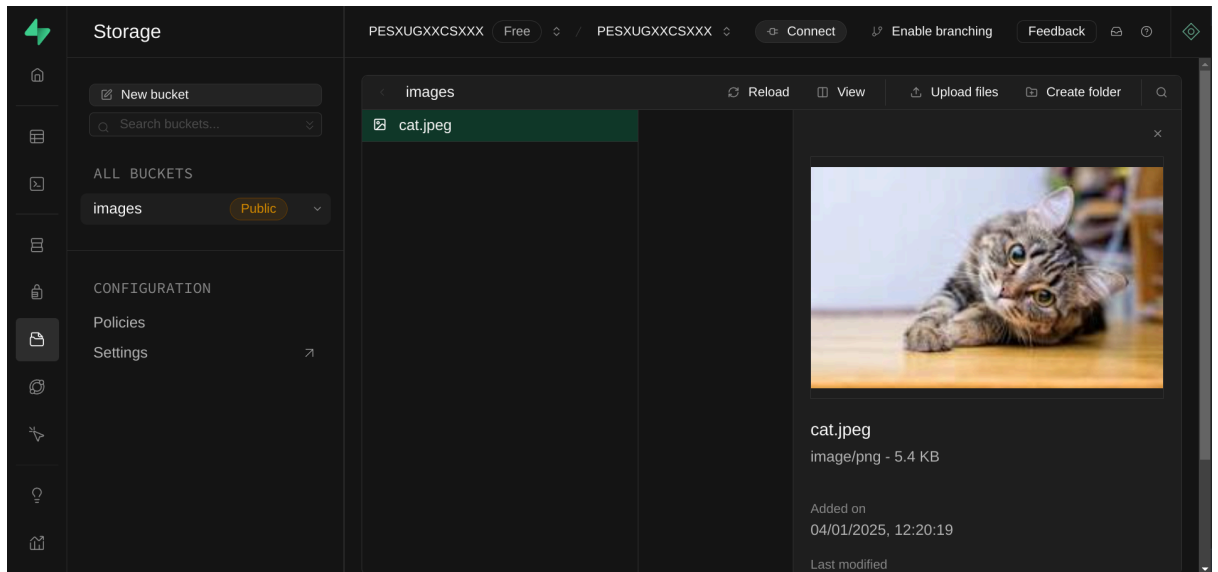
9. Next open up the **supabase\_object\_store.py** file, now you need to add your api key and project url. You can change the bucket name or the image path.

10. Run each step individually by commenting and uncommenting the other step. Code is extensively commented so that you understand what each part of the code does. At the end you should see a bucket created, cat image uploaded, and a public url to view the image. (Note you can upload the portfolio image here, and can use the public url for the upcoming practical . The **Screenshots** and **Links** required for evaluation are mentioned below.

#### - SS3 - Successful bucket creation



#### - SS4 - Successful image upload



- **Paste the public URL of the image obtained from running step4 of the python file given to you.**
  - Example :  
`https://lvxhhflqjupwgezviuqh.supabase.co/storage/v1/object/public/images/cat.jpeg`

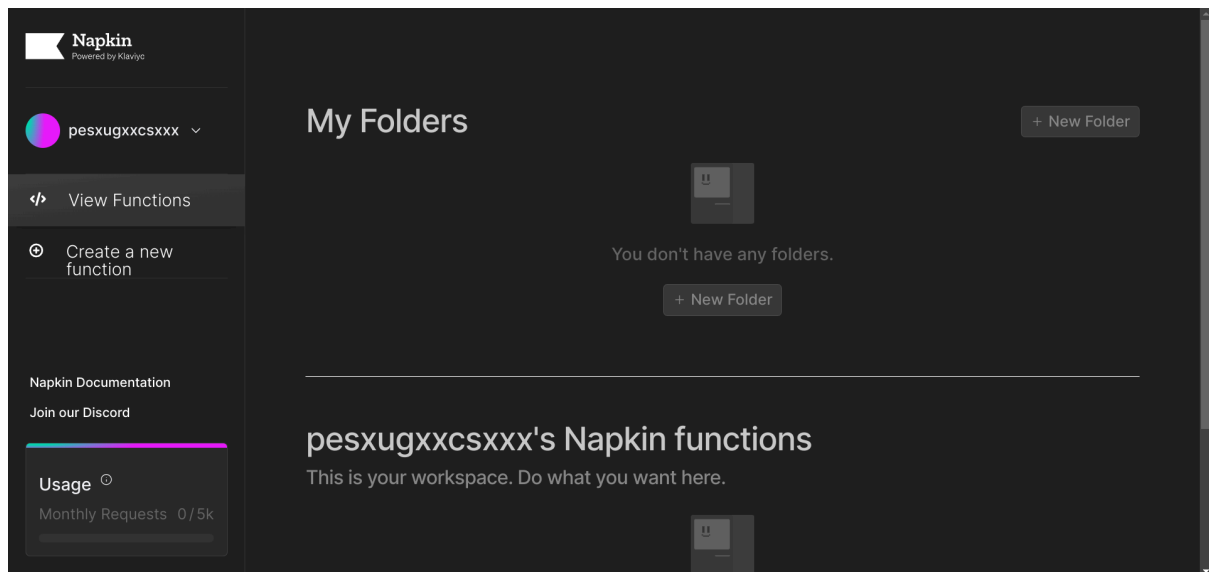


## B >> PAAS - Platform as a Service

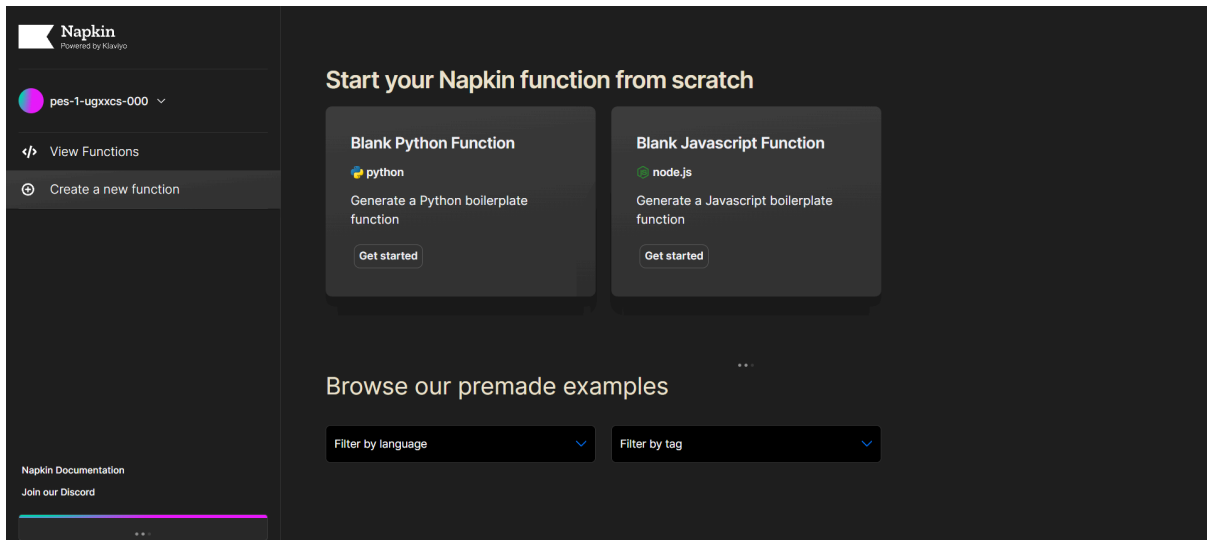
Another most widely used is function as a service or serverless functions. The idea here is to deploy one function to perform a single task, these functions can be triggered when an event happens or you can schedule it to run or you can use it to serve a request. For this we are going to use [Napkin](#).

Napkin allows the creation and deployment of serverless cloud functions instantly, all from the browser.

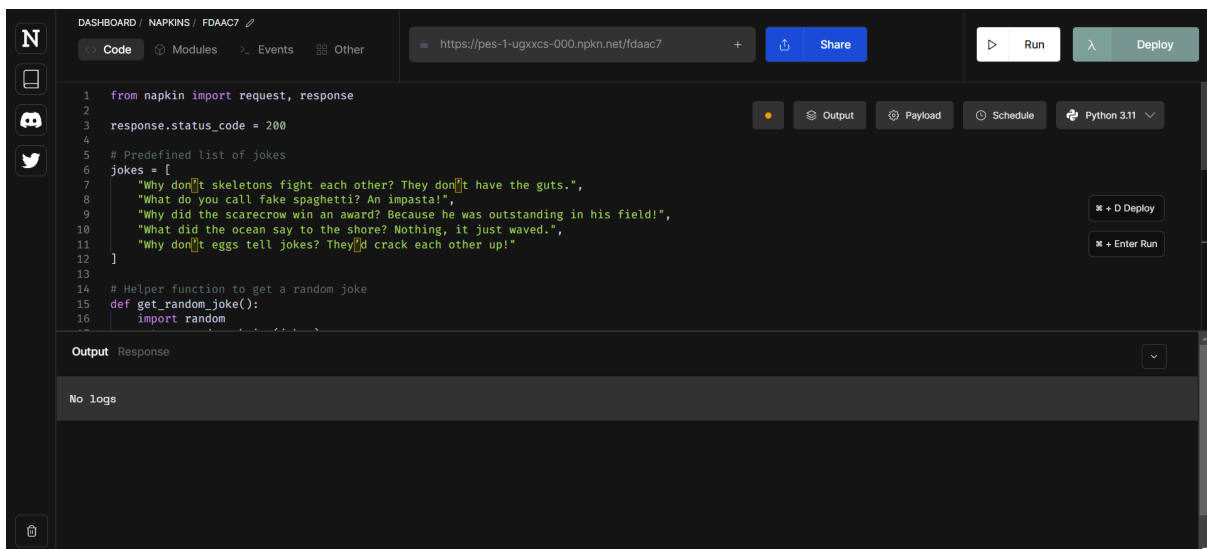
1. Sign up for napkin, <https://www.napkin.io/signup>. Next create a new workspace with the **name of the workspace as your SRN**. Take a **screenshot(SS5)**.



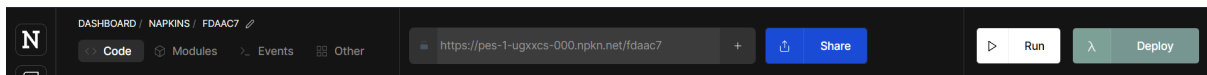
2. They have provided a lot of examples and you can deploy functions in java script or in python. We are deploying it in python. The function will be a simple joke generator ( it is very simple code, you can potentially try to bring in your own creativity here). The code is given in the **function\_as\_a\_service.py** file.
3. Click on Create a new function, Get Started on Blank Python Function.



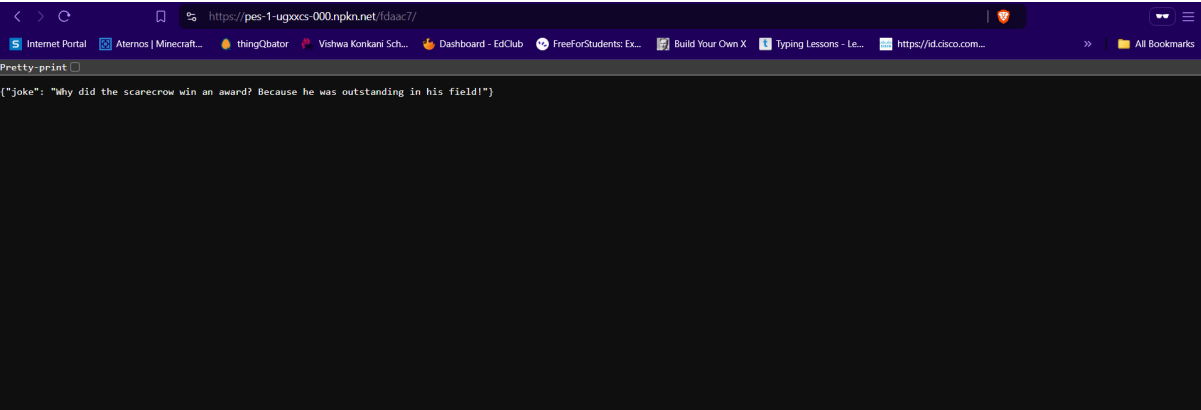
4. You can just remove the default boiler plate code on Napkin and paste in the code we provided(or any code of your own. **BE CREATIVE**). Run the code to check if the code is error free.



5. Go ahead and deploy it.



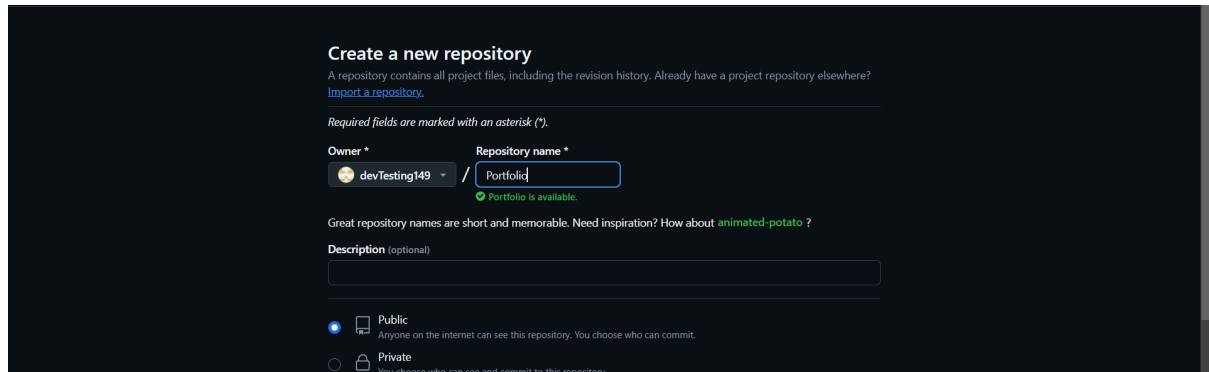
6. Once that's done copy the URL generated and paste it on a browser. Take a **Screenshot(SS6)**. Make sure the **URL is visible** for evaluation.



## C >>Software Application deployment

One of the most important uses of the cloud is to deploy an application so it reaches millions of users. For this we will deploy a portfolio website.

1. Customise the **portfolio-template/index.html** file on your liking (you can use the public url of the image here)
2. Once done make a github repository and keep your repository public.



The screenshot shows the 'Create a new repository' page on GitHub. It includes fields for 'Owner' (devTesting149), 'Repository name' (Portfolio), and 'Description' (optional). The repository is set to 'Public'. A green checkmark indicates 'Portfolio is available'.

**Create a new repository**  
A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \* devTesting149 Repository name \* Portfolio  
Portfolio is available.

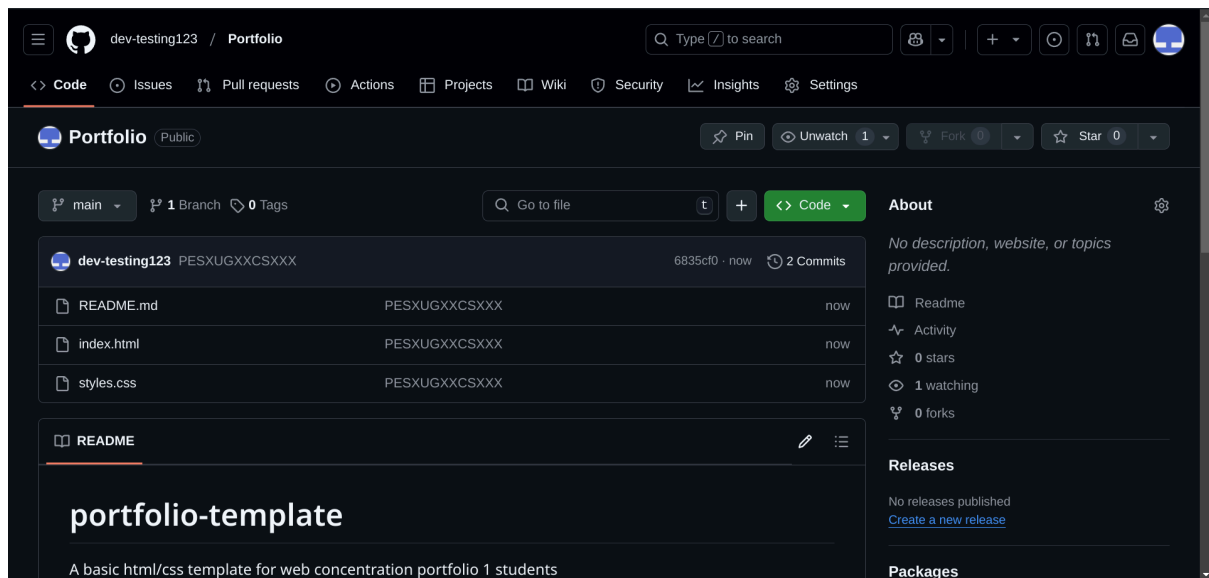
Great repository names are short and memorable. Need inspiration? How about [animated-potato](#) ?

Description (optional)

☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

3. Upload your portfolio code to this repository using git or manually uploading them.  
**NOTE - MAKE SURE THE MESSAGE ON THE COMMIT IS YOUR SRN.** Take a Screenshot(SS7).



The screenshot shows the GitHub repository page for 'Portfolio' (Public). It displays the repository name, owner (dev-testing123), and a list of files (README.md, index.html, styles.css). The repository has 1 branch and 0 tags. The README content is visible, showing the title 'portfolio-template' and a description 'A basic html/css template for web concentration portfolio 1 students'.

dev-testing123 / Portfolio

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Portfolio (Public) Pin Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file

dev-testing123 PESXUGXXCSXXX 6835cf0 · now 2 Commits

File	Author	Time
README.md	PESXUGXXCSXXX	now
index.html	PESXUGXXCSXXX	now
styles.css	PESXUGXXCSXXX	now

README

## portfolio-template

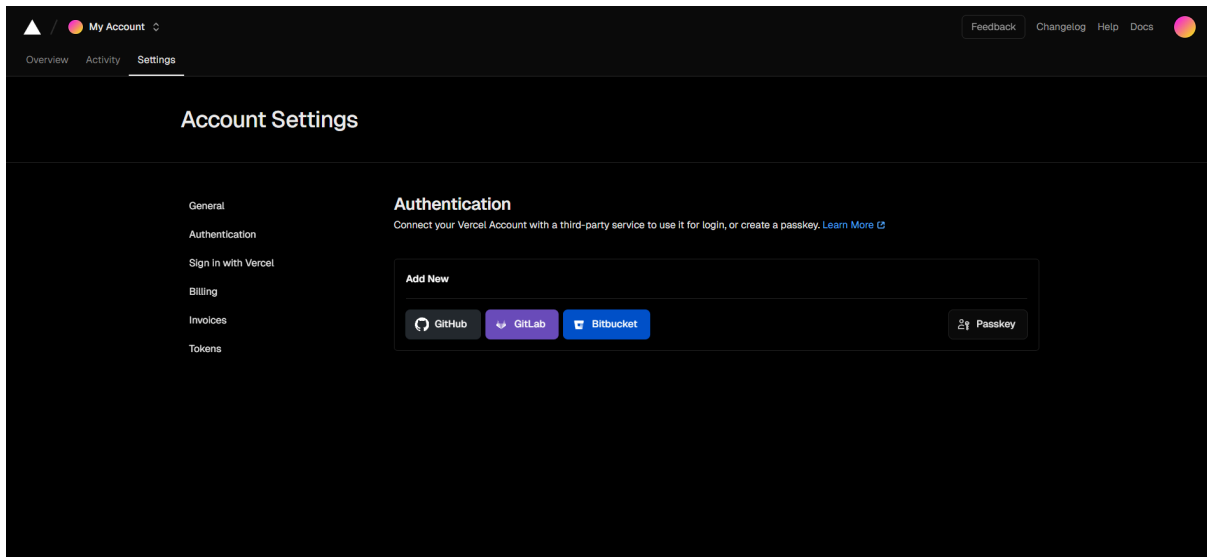
A basic html/css template for web concentration portfolio 1 students

About: No description, website, or topics provided.

Releases: No releases published. [Create a new release](#)

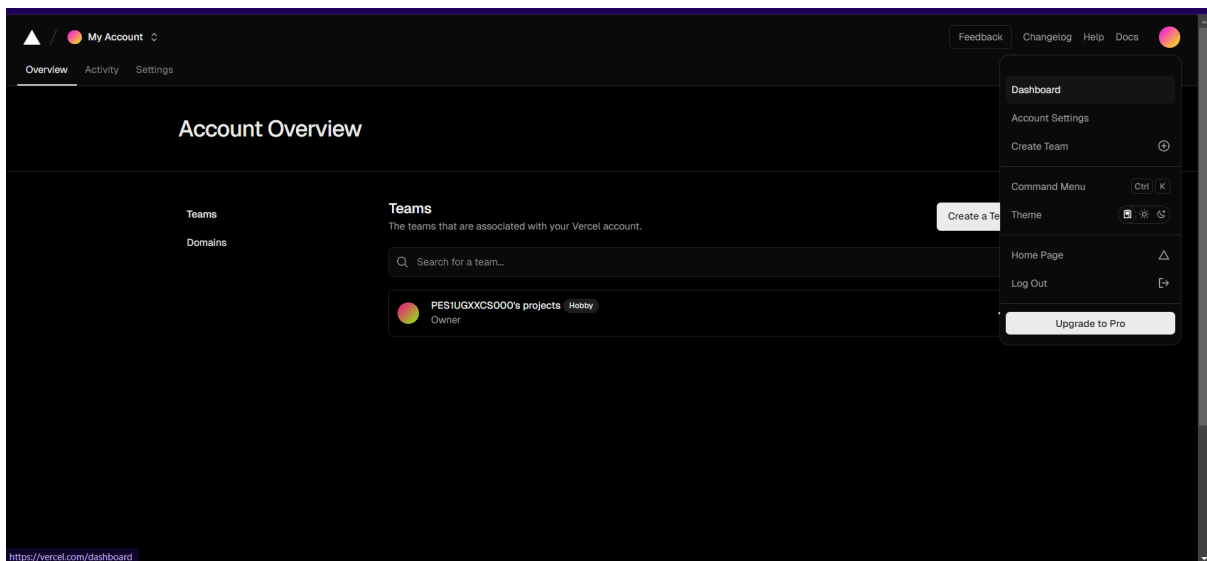
Packages

4. Next we are going to signup for vercel <https://vercel.com/login>, ideally use the same github account.

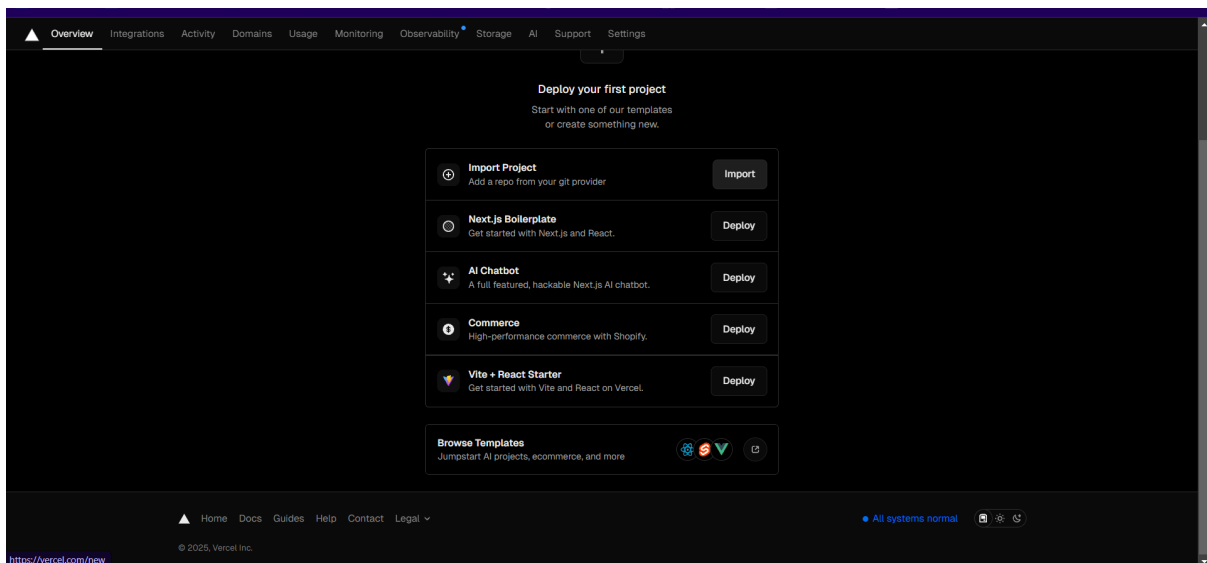


(if you use different email, then you need to connect to the GitHub explicitly)

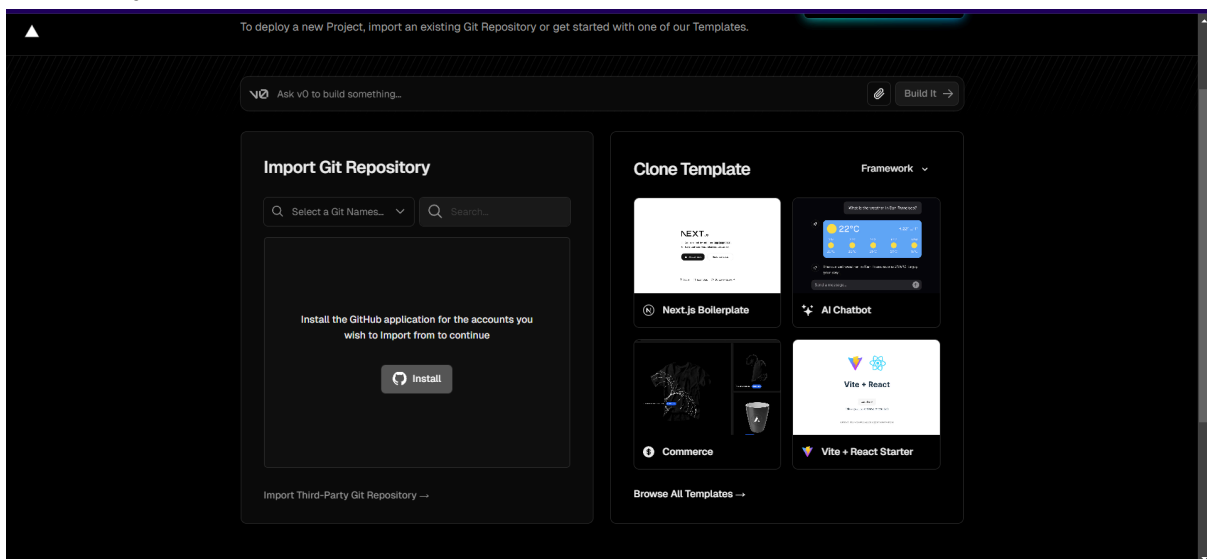
5. Now we will import project from github and deploy



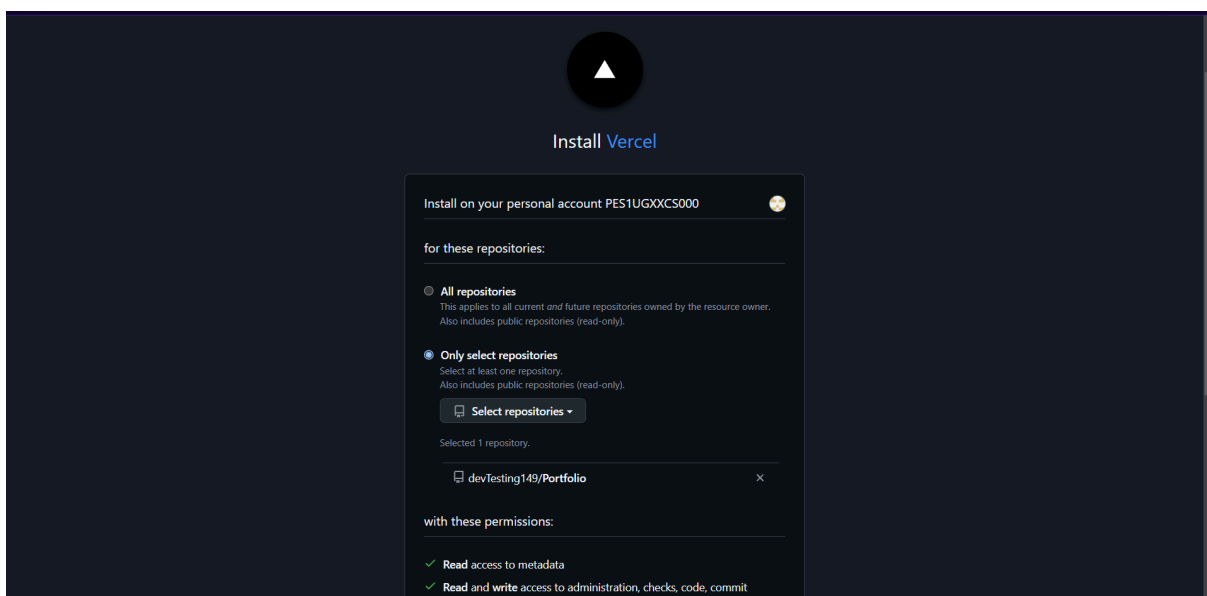
Go to dashboard



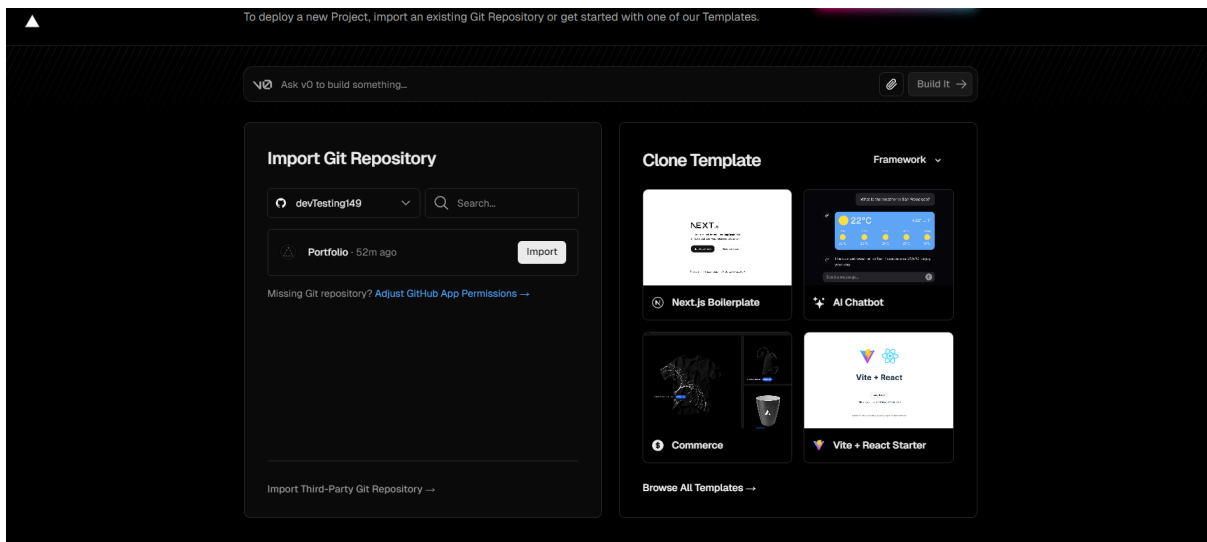
## Import project



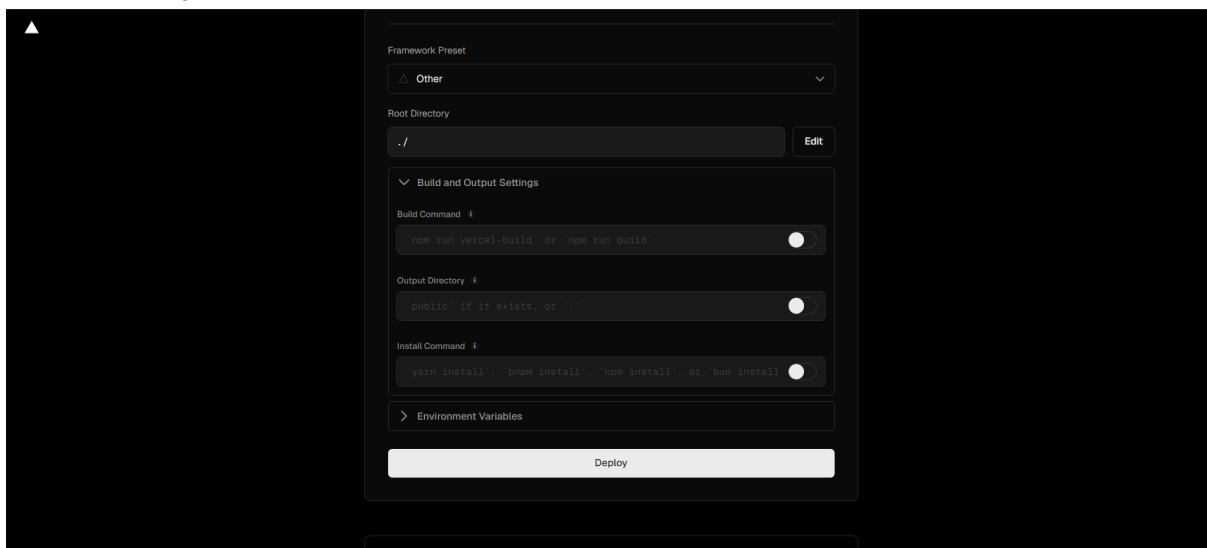
## Install



## Select the repository

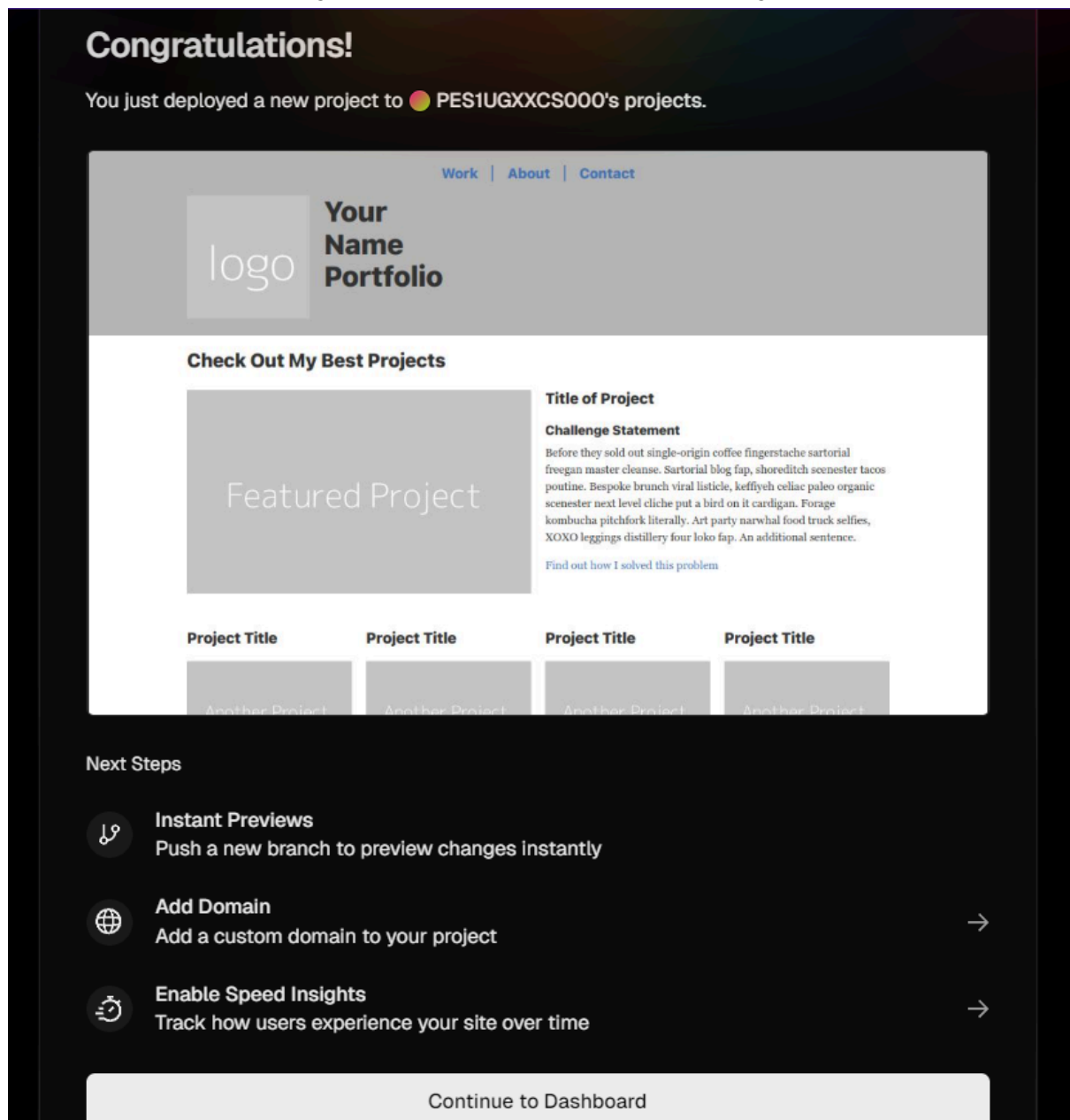


Import the project



Deploy the project

6.Finally the project is deployed. This would be the final **Screenshot(SS8)**. Additionally, also paste the **link of the deployed site**. Example : <https://portfolio-ywnr.vercel.app/>



🎉 Congratulations on completing your first lab session . Hurray!

If you are interested to explore even more cloud services for free here is the extensive list <https://free-for.dev/#/>