A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generatedA screen shot of a black rectangular object

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Setting up a CI/CD pipeline with Google Cloud Build to automate deployments of your web app when you push changes to your GitHub repository involves several steps. Here’s a general guide to get you started:

**1. Enable Google Cloud Build**

Ensure that you have an active Google Cloud Platform (GCP) account. Then, enable Google Cloud Build for your project:

* Go to the [Google Cloud Console](https://console.cloud.google.com/).
* Select or create a new project.
* Navigate to "Cloud Build" and enable it if it's not already enabled.

**2. Set Up Source Repository**

Since you've mentioned that your code is already on GitHub, you'll need to connect your GitHub repository to Google Cloud Build:

* In the Cloud Build page, go to the "Triggers" section.
* Click on “Connect repository”.
* Choose "GitHub" as your source.
* You may need to authenticate and authorize Google Cloud to access your GitHub repositories.
* Select the repository where your web app code is stored.

**3. Create a Build Trigger**

After connecting your repository, set up a build trigger:

* Click on “Create Trigger”.
* Enter the name for the trigger and select the GitHub repository.
* Configure the trigger to build your project when changes are pushed to a specific branch or tag.
* You can specify build configurations using a **cloudbuild.yaml** file or use inline configurations.

**4. Configure cloudbuild.yaml**

Create a **cloudbuild.yaml** file in the root of your repository. This file will define the steps that Cloud Build should execute. Here’s a basic example that builds and deploys a simple web application:

yaml

Copy code

steps: - name: 'gcr.io/cloud-builders/gcloud' args: ['app', 'deploy'] timeout: '1600s'

This configuration deploys your application using Google Cloud App Engine. Adjust the steps based on your deployment needs (e.g., building a Docker container, running tests, deploying to a different service).

**5. Push cloudbuild.yaml to GitHub**

Commit and push the **cloudbuild.yaml** file to your repository:

bash

Copy code

git add cloudbuild.yaml git commit -m "Add Cloud Build configuration" git push origin master

**6. Test the Trigger**

Make a small change in your repository and push the change to GitHub to see if Cloud Build triggers a build and deploys your application as expected.

**7. Set Permissions and Roles**

Ensure that Cloud Build has the necessary permissions to deploy your application. You might need to grant additional roles to the Cloud Build service account, such as App Engine Admin, Storage Admin, or Custom roles depending on your resources.

**8. Monitor Builds**

You can monitor the build processes and view logs in the Google Cloud Console under the "Cloud Build" section.

By following these steps, you should be able to set up a CI/CD pipeline using Google Cloud Build for your web app. Adjust the build steps and permissions based on your specific application's architecture and requirements.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The google support only provides limited service. So I configured the Cloud Build trigger for 'Manual invocation'. When I executed the trigger manually, it revealed multiple issues. For instance, there were problems with my cloudbuild.yaml configuration, and I had to enable the IAM API, which I've done through the provided link. Additionally, I was reminded that Cloud Build is only available in certain regions. I have now addressed these problems. Any way, configuring the Cloud Build trigger for 'Manual invocation' is the most important, it will show what problem you have.