

DevOps CI/CD StaticWebsite Assignment

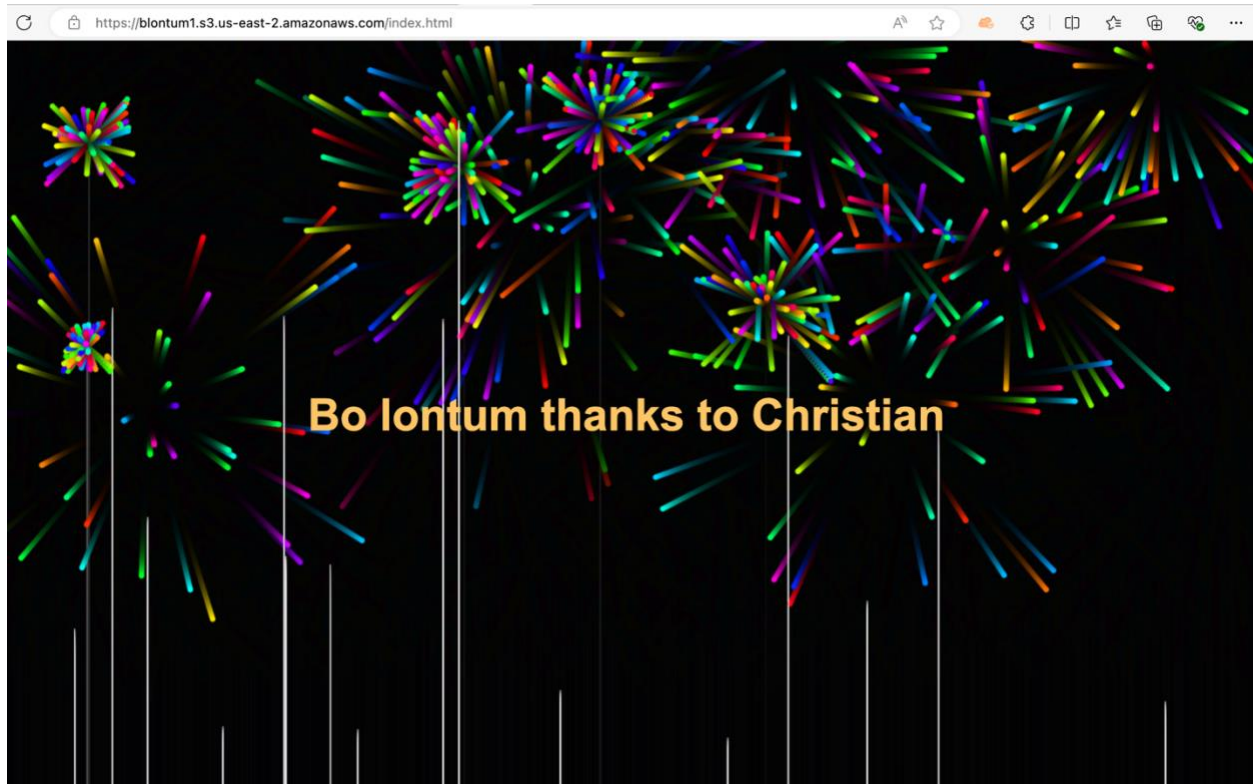
Name: Bo lontum

Website: <http://boyoutubechannel.s3-website.us-east-2.amazonaws.com>

CI/CD Pipeline screenshots

The screenshot displays the AWS CodePipeline console for a pipeline named 'blontum1'. The pipeline is in a 'QUEUED' state. The left sidebar shows the navigation menu with options like 'Source', 'Artifacts', 'Build', 'Deploy', 'Pipeline', 'Getting started', 'Pipelines', 'Pipeline', 'History', 'Settings', and 'Settings'. The main content area shows the pipeline details, including the pipeline type 'V2' and execution mode 'QUEUED'. The pipeline consists of two stages: 'Source' and 'Deploy'. The 'Source' stage is marked as 'Succeeded' and uses the 'GitHub (Version 1)' provider. The 'Deploy' stage is also marked as 'Succeeded' and uses the 'Amazon S3' provider. The pipeline execution ID is 2b81c6da-90a1-45ed-a86c-809a1fe2940b. The 'Deploy' stage has a 'Start rollback' button. The 'Source' stage has a 'View details' button. The 'Deploy' stage has a 'Disable transition' button. The pipeline is currently in a 'QUEUED' state, as indicated by the 'Execution mode: QUEUED' label.

Screen shot of static website



DEVOPS HOMEWORK1

Talking: Justo N

Setup a two stages CI/CD pipeline using **CodePipeline** to automatically deploy a static website on amazon S3 bucket.

The website will be a simple HTML webpage displaying your name.

Source your code from your personal **Github** account.

Submission:

Provide your repository along with screenshots of your CI/CD pipeline and the link to your s3 website.

• **Create a GitHub Repository:**

- Go to [GitHub](https://github.com) and log in to your account.
- Click on "New" to create a new repository.
- Name your repository, for example, "simple-static-website".
- You can add a description if you like and choose "Public".

- Initialize the repository with a README file and create the repository.
- **Upload the HTML File:**
 - On the main page of your new repository, click on "Add file" and select "Create new file".
 - Name the file `index.html`.
 - Copy and paste the original HTML code into this file
 - - Commit the new file by clicking on "Commit new file".
 - **Enable GitHub Pages:**
 - Go to the "Settings" tab of your repository.
 - Scroll down to the "Pages" section on the left sidebar.
 - Under "Source", select the branch that contains your `index.html` file (typically `main` or `master`).
 - Click "Save".

GitHub Pages will now build your site and provide a link to access it.

- **Access the Live Page:**
 - Once GitHub Pages has finished building your site, it will provide a URL where you can view your webpage.
 - The URL will typically be in the format `https://your-username.github.io/repository-name/`.

After completing these steps, you should be able to view your webpage live on GitHub Pages.