



National University
of computer and emerging sciences

CS4085 MLOps

Assignment #1

Name:

Mahreen Athar (19i-1712)

Adeen Ayub Vine (19i-0553)

Submitted to:

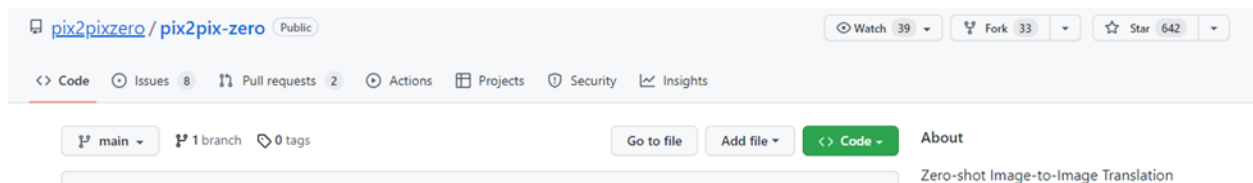
Sir Hussain Shahbaz

Introduction:

The repository chosen is called "pix2pix-zero" which is a "Zero-shot Image-to-Image Translation" using the diffusers library.

Procedure:

We began by forking this repository by selecting the fork option found in the top bar of the repository.



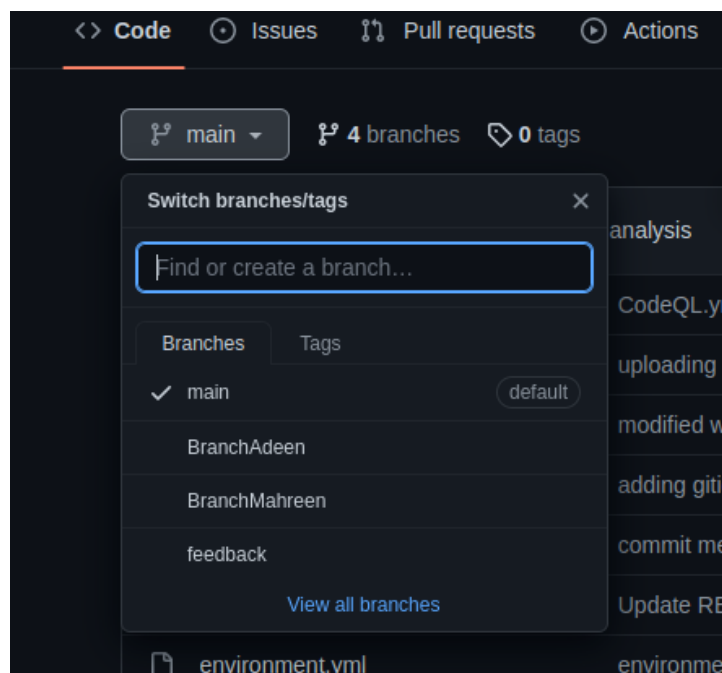
We then proceeded to clone this repository by using the following command in the Linux Terminal.

```
git clone https://github.com/pix2pixzero/pix2pix-zero.git
```

After cloning the selected repository, each member of our repository created a branch specific to them by entering the following command in the Linux Terminal.

```
git checkout -b BranchMahreen
```

```
git checkout -b BranchAdeen
```



Each member of the repository then created .github/workflows directory in their respective branches by inputting the following commands in the Linux Terminal.

```
mkdir .github
```

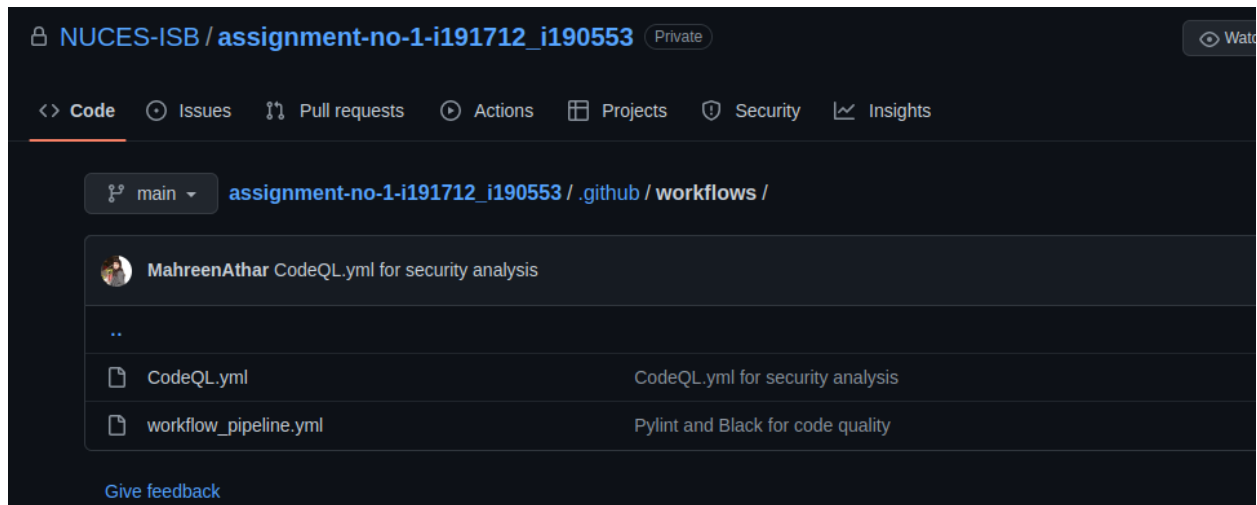
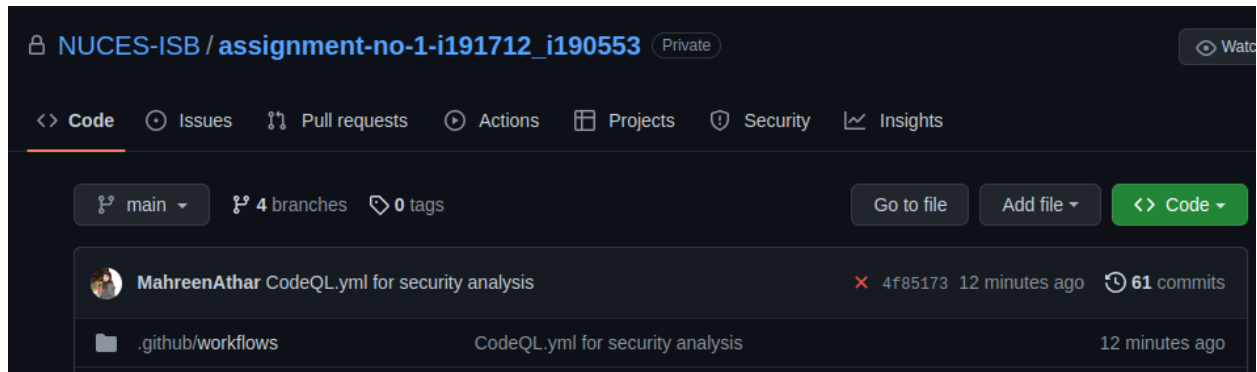
```
cd .github
```

```
mkdir workflows
```

```
cd workflows
```

```
touch workflow_pipeline.yml
```

```
touch CodeQL.yml
```



After the creation of these .yml files, we defined the workflow used for checking the quality of the code by using Pylint and Black in the file “workflow_pipeline.yml”. For security scanning and alerts related to our repository, we used CodeQL in the file “CodeQL.yml”. Both of these files can be found in our github repository. These workflows ran successfully in our repositories, as can be seen below.

MahreenAthar / pix2pix-zero Public
forked from pix2pixzero/pix2pix-zero

<> Code Pull requests Actions Projects Wiki Security Insights Settings

Actions New workflow

All workflows

CodeQL
workflow_pipeline

Management
Caches

All workflows Filter workflow runs

Showing runs from all workflows

41 workflow runs

	Event	Status	Branch	Actor
✓ CodeQL.yml for security analysis workflow_pipeline #34: Commit 4f85173 pushed by MahreenAthar	main	3 minutes ago 57s	...	
✓ CodeQL.yml for security analysis CodeQL #7: Commit 4f85173 pushed by MahreenAthar	main	3 minutes ago 2m 49s	...	
✓ CodeQL.yml for security analysis workflow_pipeline #33: Commit 4f85173 pushed by MahreenAthar	BranchMahreen	3 minutes ago 1m 18s	...	
✓ CodeQL.yml for security analysis CodeQL #6: Commit 4f85173 pushed by MahreenAthar	BranchMahreen	3 minutes ago 2m 44s	...	

The CodeQL was used for the security analysis. The below screenshot shows the result from the code scanning and it shows that there are no security threats, alerts or leaks which means that we don't have to recode anything. The repository is fine.

Analyze (python)
succeeded 1 minute ago in 2m 39s

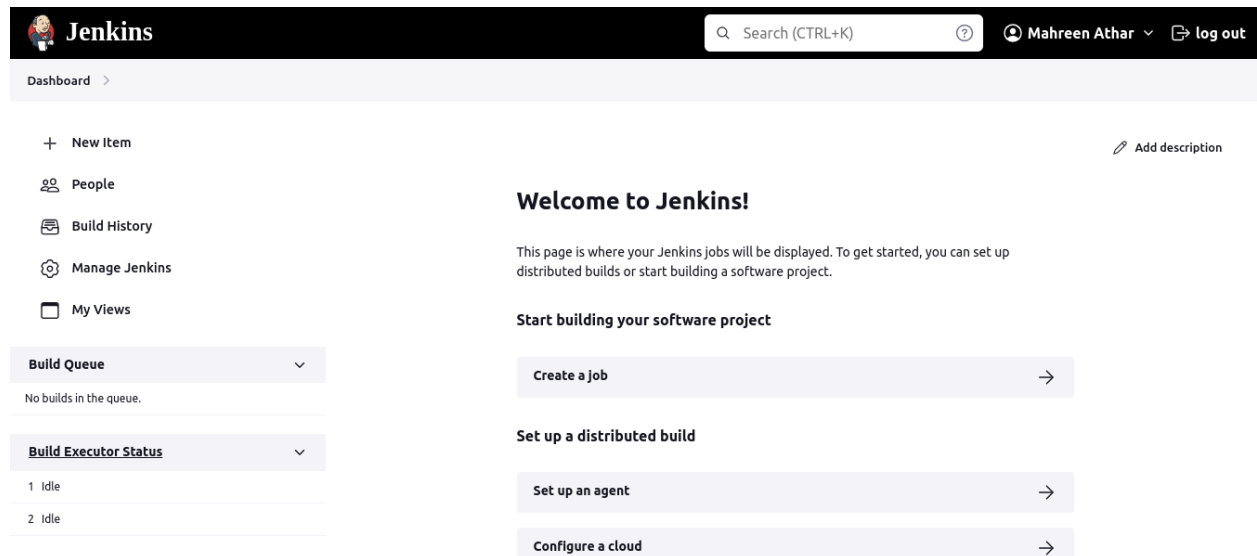
Search logs

CodeQL Analysis for security 2m 5s

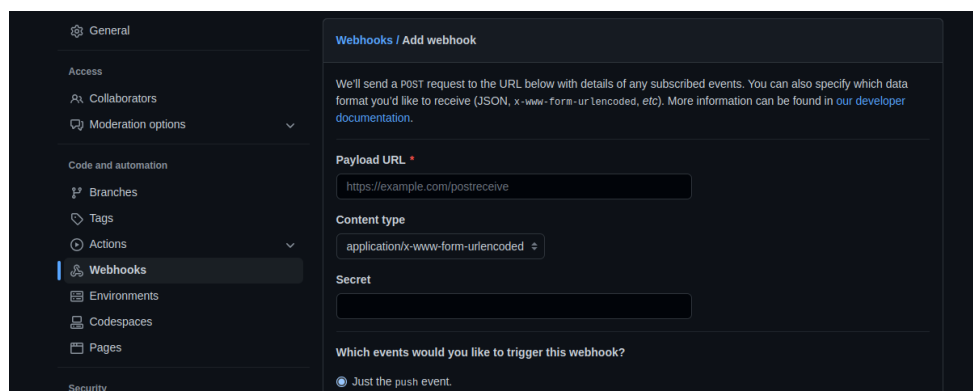
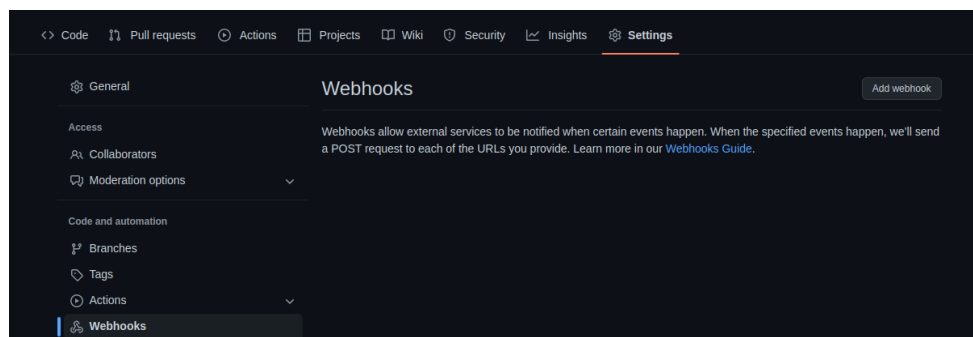
```
1 ▶ Run github/codeql-action/analyze@v2
27 /opt/hostedtoolcache/CodeQL/2.12.2-20230207/x64/codeql/codeql version --format=terse
28 2.12.2
29 ▶ Extracting python
406 ▶ Finalizing python
410 Code Scanning configuration file being processed in the codeql CLI.
411 ▶ Running queries for python
534 ▶ Interpreting results for python
669 Analysis produced the following diagnostic data:
670
671 | Diagnostic | Summary |
672 +-----+-----+
673 | Successfully extracted Python files | 10 results |
674
675 Analysis produced the following metric data:
676
677 | Metric | Value |
678 +-----+-----+
679 | Total lines of Python code in the database | 116533 |
680 | Total lines of user written Python code in the database | 952 |
681
682
```

After the implementation of our above mentioned instructions, we were done with our github actions and we proceeded onto our work regarding Jenkins. The procedure to integrate Jenkins with Github can be found in the link below.

<https://www.blazemeter.com/blog/how-to-integrate-your-github-repository-to-your-jenkins-project>



First we go to 'Settings' on our main interface of the repository and then go into 'Webhooks' from where we create a webhook for our jenkins integration.



After these steps, we will create a freestyle project and then use 'push' as the trigger so anytime something is pushed onto the repository our job will be triggered on Jenkins and it will automatically run.

Our Repository:

https://github.com/NUCES-ISB/assignment-no-1-i191712_i190553