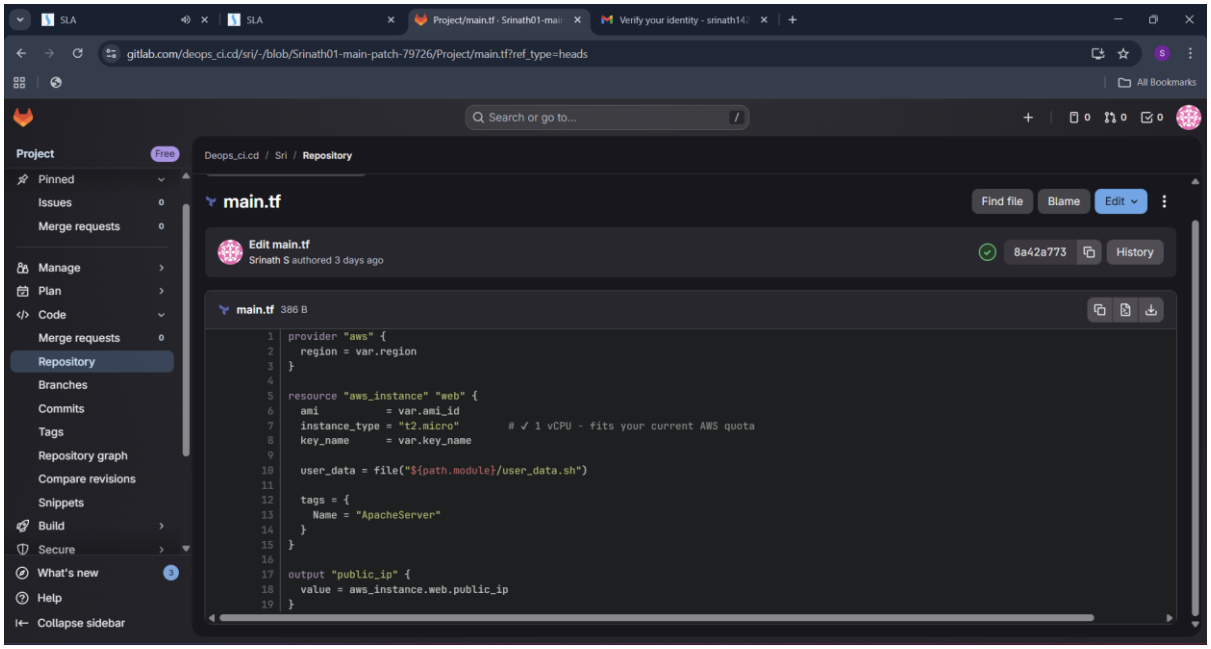
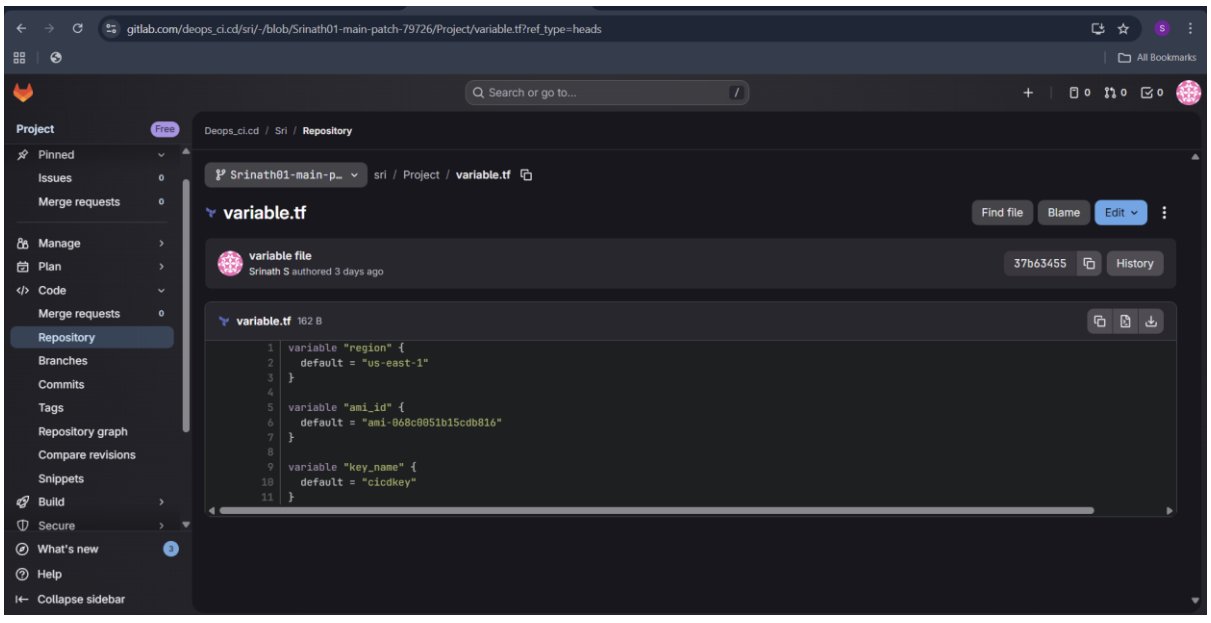
TASK-17

Gitlab pipeline for Infrastructure Terraform

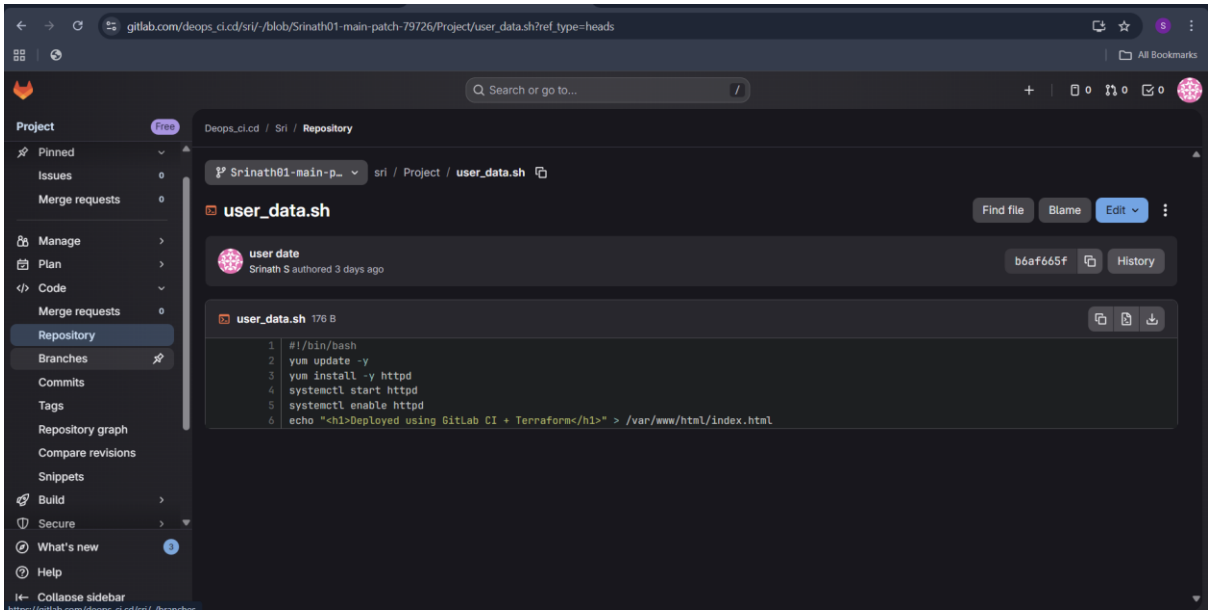
Create Terraform main file:



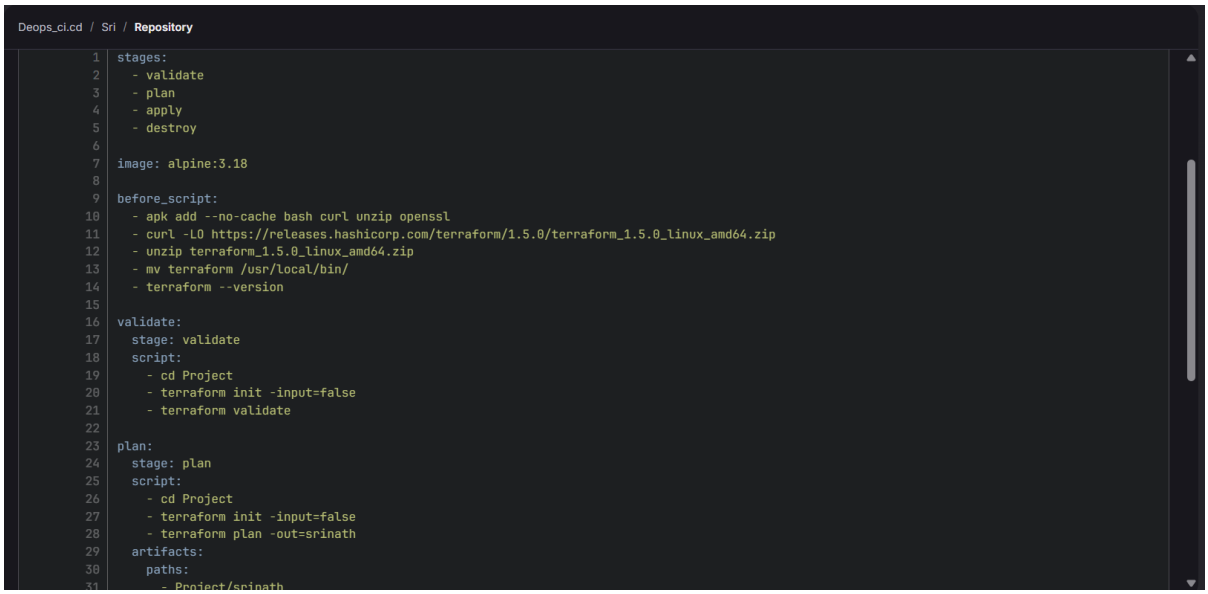
Create Variable File:



Create User_date:



Write a yml file as .gitlab-ci.yml:



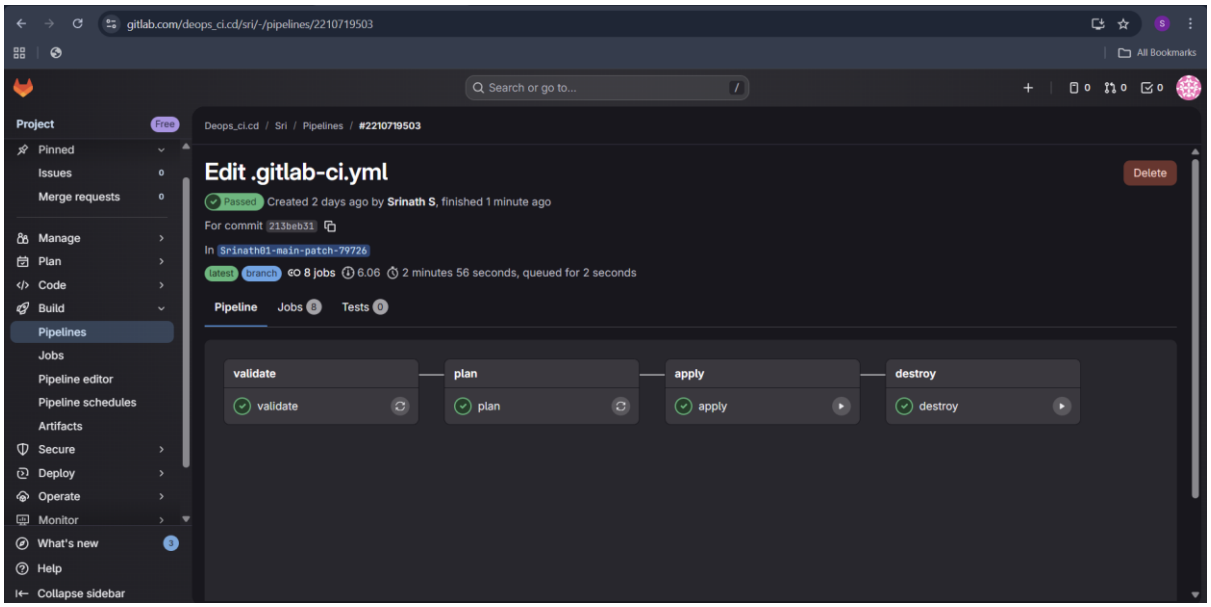
Code:

```
stages:
  - validate
  - plan
  - apply
  - destroy

image: alpine:3.18

before_script:
  - apk add --no-cache bash curl unzip openssl
  - curl -LO https://releases.hashicorp.com/terraform/1.5.0/terraform_1.5.0_linux_amd64.zip
  - unzip terraform_1.5.0_linux_amd64.zip
  - mv terraform /usr/local/bin/
  - terraform --version

validate:
  stage: validate
  script:
    - cd Project
    - terraform init -input=false
    - terraform validate

plan:
  stage: plan
  script:
    - cd Project
    - terraform init -input=false
    - terraform plan -out=srinath
  artifacts:
    paths:
```

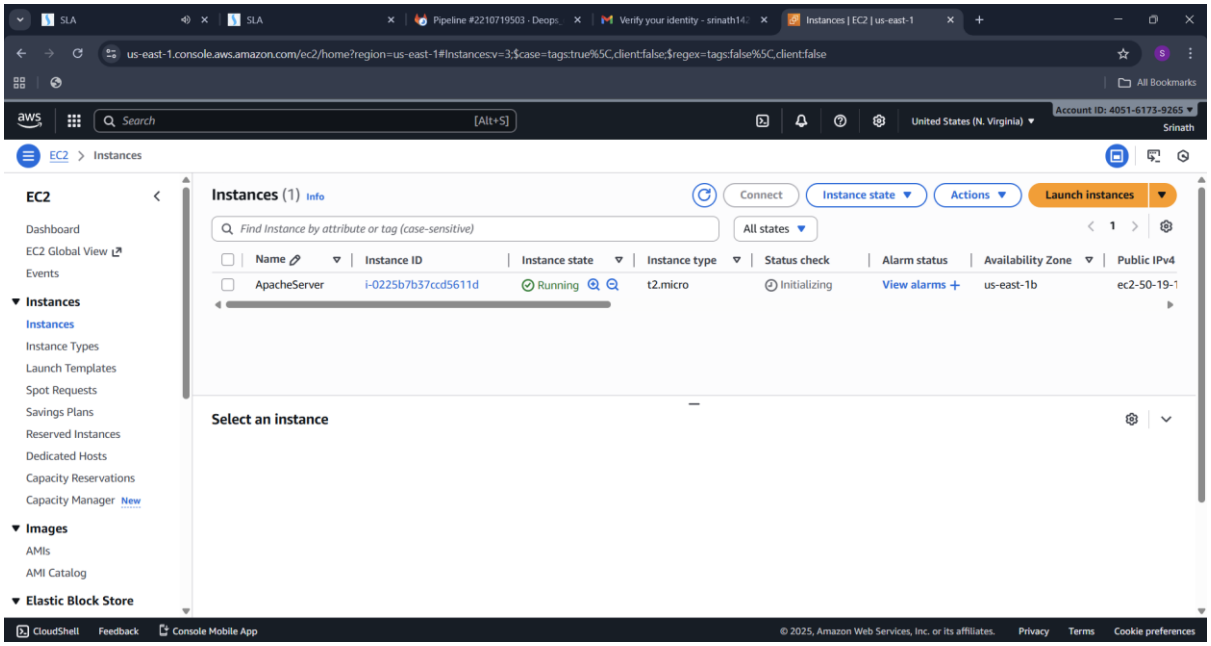
```
  - destroy

image: alpine:3.18

before_script:
  - apk add --no-cache bash curl unzip openssl
  - curl -LO
https://releases.hashicorp.com/terraform/1.5.0/terraform_1.5.0_linux_amd64.zip
  - unzip terraform_1.5.0_linux_amd64.zip
  - mv terraform /usr/local/bin/
  - terraform --version

validate:
  stage: validate
  script:
    - cd Project
    - terraform init -input=false
    - terraform validate

plan:
  stage: plan
  script:
    - cd Project
    - terraform init -input=false
    - terraform plan -out=srinath
  artifacts:
    paths:
      - Project/srinath
```

```
apply:
  stage: apply
  when: manual
  dependencies:
    - plan
  script:
    - cd Project
    - terraform init -input=false
    - terraform apply -auto-approve srinath
  artifacts:
    paths:
      - Project/terraform.tfstate
      - Project/terraform.tfstate.backup
    expire_in: 1 hour

destroy:
  stage: destroy
  when: manual
  dependencies:
    - apply
```

Pipe Line:



Instance Created



Instanced Deleted: