Hackathon Presentation (DevOps)

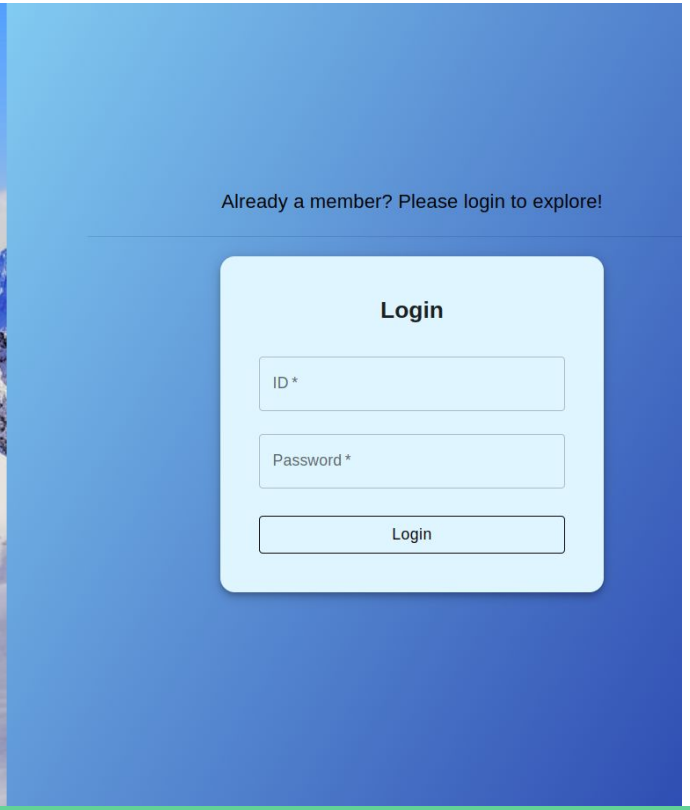
BUET DareDevils

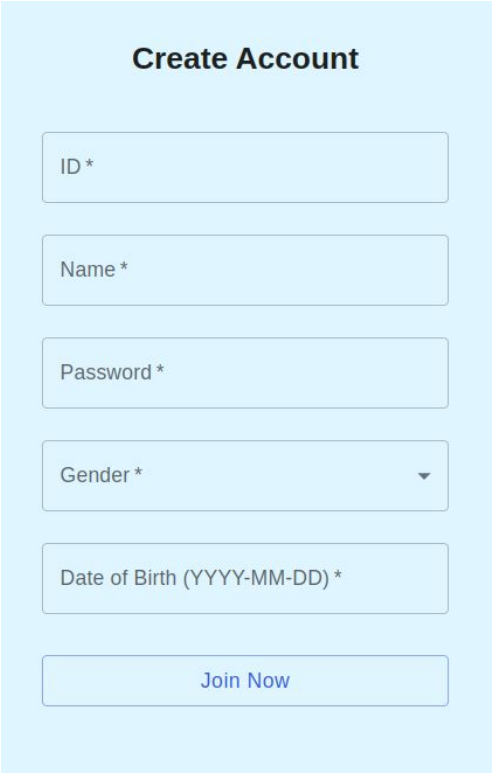
Md. Asif Haider
Zannatul Naim
Mahdee Mushfique Kamal

Challenge 0: Basic Application Development

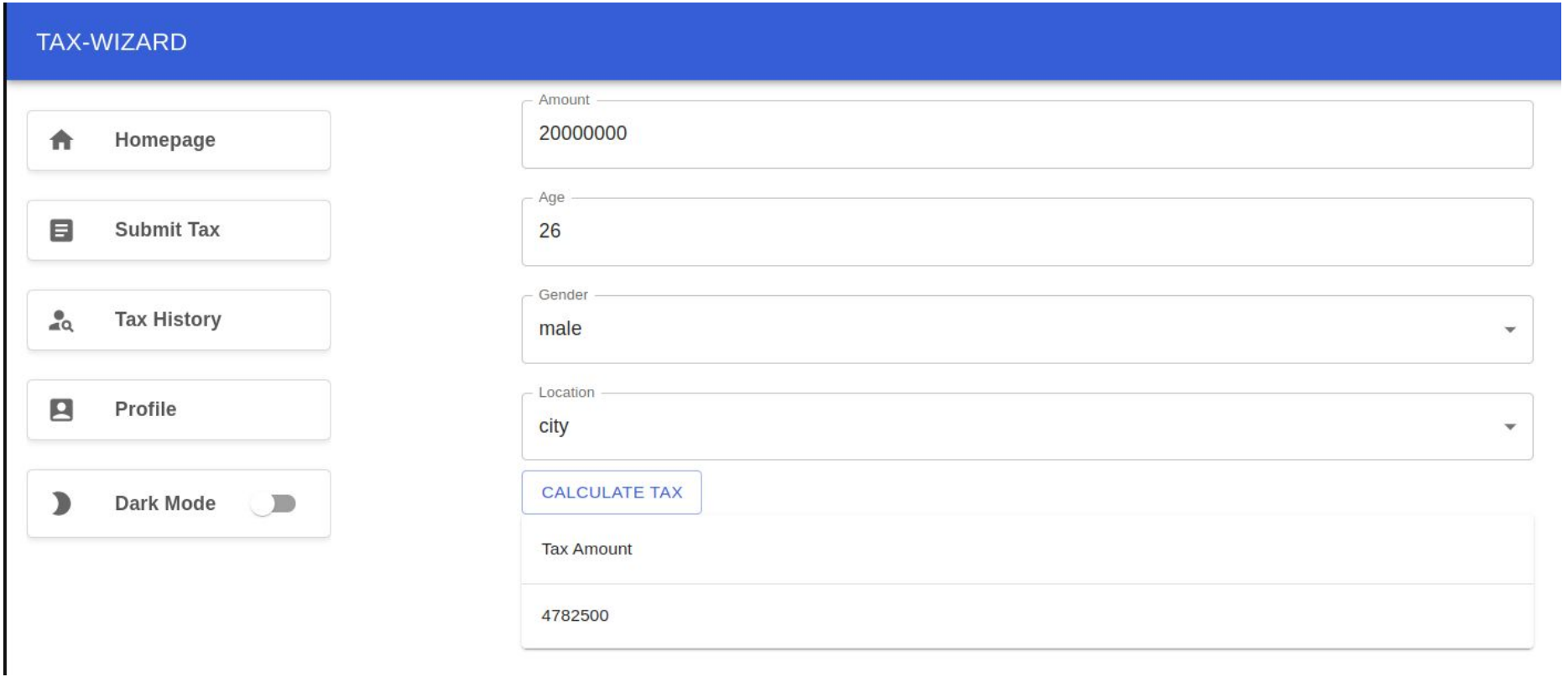
- Backend
  - FastAPI (Python)
- Database
  - PostgreSQL (SQLAlchemy)
- Frontend
  - NextJS (JavaScript)

User Interface





Functionalities



TAX-WIZARD

Homepage

Submit Tax

Tax History

Profile

Dark Mode

Year

1999

Income

1000000

Location

dhaka

SUBMIT TAX

Tax ID	Year	Location	Total Income	Total Taxable Amount	Total Payable Tax
03c75dd0-7d6b-49c9-a25f-e95411fa24de	1999	dhaka	1000000 Tk	650000 Tk	72500 Tk

TAX-WIZARD

Homepage

Submit Tax

Tax History

Profile

Dark Mode

SHOW TAX DETAILS

Tax History Report

Year	Location	Total Income	Taxable Amount	Tax
1999	dhaka	1000000 Tk	650000 Tk	72500 Tk
2011	dhaka	8000000 Tk	7650000 Tk	1782500 Tk
2014	dhaka	800000 Tk	450000 Tk	42500 Tk
2015	dhaka	1000000 Tk	650000 Tk	72500 Tk
2016	dhaka	1000000 Tk	650000 Tk	72500 Tk
2017	dhaka	1000000 Tk	650000 Tk	72500 Tk
2018	dhaka	1000000 Tk	650000 Tk	72500 Tk
2019	dhaka	1000000 Tk	650000 Tk	72500 Tk
2020	dhaka	1000000 Tk	650000 Tk	72500 Tk
2022	dhaka	1000000 Tk	650000 Tk	72500 Tk
2023	dhaka	1000000 Tk	650000 Tk	72500 Tk
2025	dhaka	80000 Tk	0 Tk	0 Tk
2026	city	787878789 Tk	787528789 Tk	196752197 Tk
2027	city	787878789 Tk	787528789 Tk	196752197 Tk
2028	city	787878789 Tk	787528789 Tk	196752197 Tk
2029	dhaka	1000000 Tk	650000 Tk	72500 Tk
2030	city	800000 Tk	450000 Tk	42500 Tk

DOWNLOAD PDF

Tax History Report

Year	Location	Total Income	Taxable Amount	Tax
1999	dhaka	1000000 Tk	650000 Tk	72500 Tk
2011	dhaka	8000000 Tk	7650000 Tk	1782500 Tk
2014	dhaka	800000 Tk	450000 Tk	42500 Tk
2015	dhaka	1000000 Tk	650000 Tk	72500 Tk
2016	dhaka	1000000 Tk	650000 Tk	72500 Tk
2017	dhaka	1000000 Tk	650000 Tk	72500 Tk
2018	dhaka	1000000 Tk	650000 Tk	72500 Tk
2019	dhaka	1000000 Tk	650000 Tk	72500 Tk
2020	dhaka	1000000 Tk	650000 Tk	72500 Tk
2022	dhaka	1000000 Tk	650000 Tk	72500 Tk
2023	dhaka	1000000 Tk	650000 Tk	72500 Tk
2025	dhaka	80000 Tk	0 Tk	0 Tk
2026	city	787878789 Tk	787528789 Tk	196752197 Tk
2027	city	787878789 Tk	787528789 Tk	196752197 Tk
2028	city	787878789 Tk	787528789 Tk	196752197 Tk
2029	dhaka	1000000 Tk	650000 Tk	72500 Tk
2030	city	800000 Tk	450000 Tk	42500 Tk

Challenge 1: CI/CD Pipeline

- Git Strategy
- CI/CD Pipeline
  - Build
  - Test
  - Deploy
- Cloud Tools

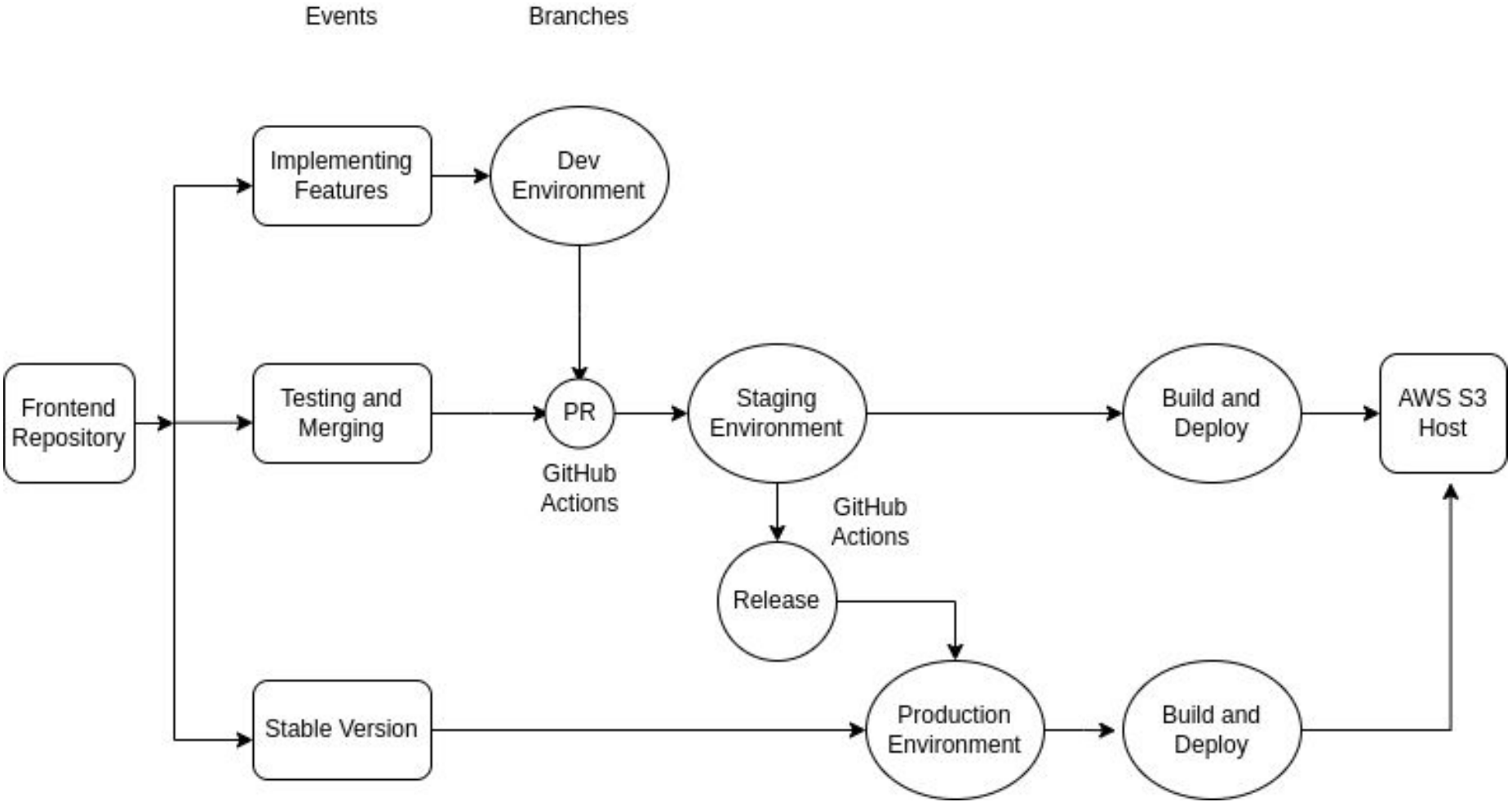
Git (Version Control) Strategy

- GitHub
- 3 different environments
  - Development
  - Staging
  - Production
- Different Repositories
- GitHub Actions for CI

Deployment (AWS Cloud)

- AWS Elastic Container (ECS)
  - Scalable and flexible deploying infrastructure
- AWS RDS (Relational Database)
  - Easy to use postgres db
- AWS S3 (Frontend)
  - Reliable, cost-effective static storage

Frontend Pipeline


Events
Branches
Implementing Features
Dev Environment
Frontend Repository
Testing and Merging
PR
GitHub Actions
Staging Environment
Build and Deploy
AWS S3 Host
GitHub Actions
Release
Stable Version
Production Environment
Build and Deploy

Backend Pipeline

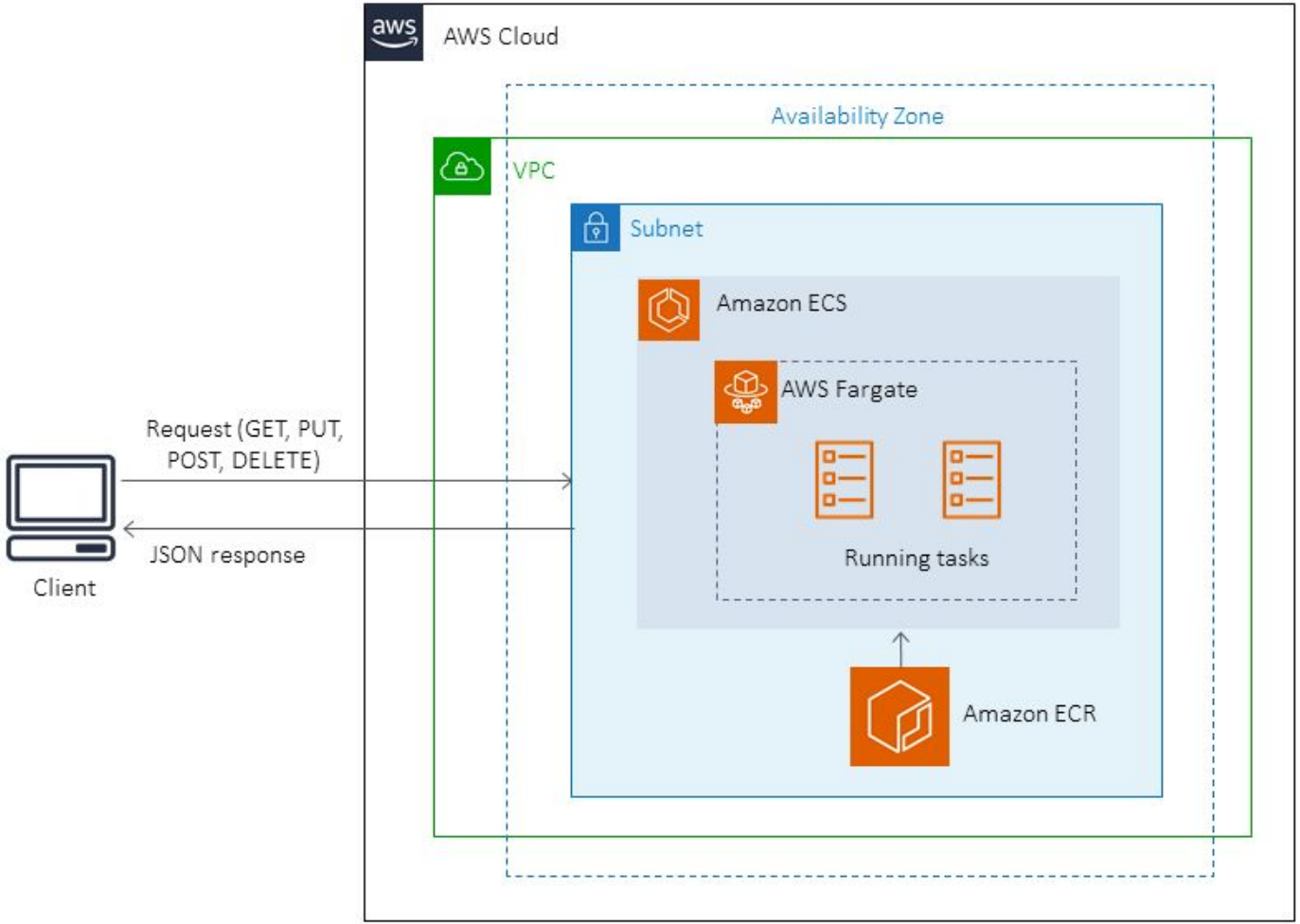
Example Test Scripts

```python
from app.tax_calculation import calculate_final_tax


def test_calculate_tax():
    assert (
        calculate_final_tax(amount=1000000, gender="male", age=40, location="major")
        == 72500
    )
    assert (
        calculate_final_tax(amount=400000, gender="male", age=25, location="non_city")
        == 3000
    )

    assert (
        calculate_final_tax(amount=400000, gender="female", age=25, location="non_city")
        == 0
    )
```

```python
def test_index():
    response = client.get("/")
    response_json = response.json()
    assert response.status_code == 200
    assert response_json["success"] is True


def test_calculate_tax():
    body = {"amount": 1000000, "gender": "male", "age": 45, "location": "city"}
    response = client.post("/calculate_tax", json=body)
    response_json = response.json()
    assert response.status_code == 200
    assert response_json["tax_amount"] == 72500
```

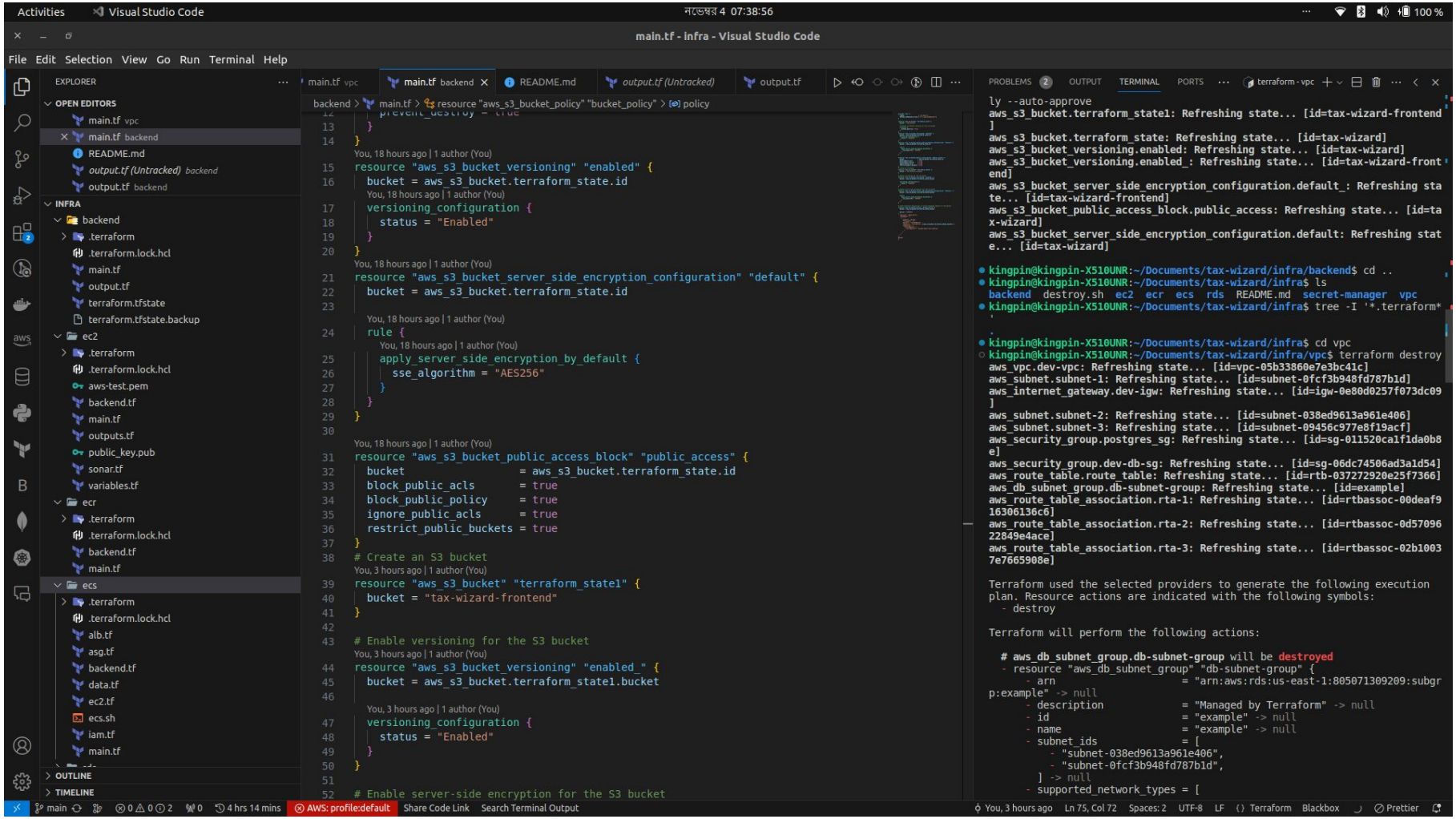
Backend Cloud Tools

- Amazon ECR (Elastic Cloud Registry)
  - Store, manage and employ docker container images
- Amazon ECS (Elastic Container Service)
  - Scalable, high-performance container orchestration service
- Amazon Fargate
  - Serverless container compute engine


AWS Cloud
Availability Zone
VPC
Subnet
Amazon ECS
AWS Fargate
Running tasks
Amazon ECR
Request (GET, PUT, POST, DELETE)
JSON response
Client

Challenge 2: Scaling

- Infrastructure Management
  - Terraform (Infrastructure As Code)
- Scaling
  - Automatic Resource Scale Up and Down
- Load Balancing


Activities
Visual Studio Code
নভেম্বর 4 07:38:56
100 %
main.tf - infra - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
main.tf vpc
main.tf backend
README.md
output.tf (Untracked) backend
output.tf backend
INFRA
backend
.terraform
.terraform.lock.hcl
main.tf
output.tf
terraform.tfstate
terraform.tfstate.backup
ec2
.terraform
.terraform.lock.hcl
aws-test.pem
backend.tf
main.tf
outputs.tf
public_key.pub
sonar.tf
variables.tf
ecr
.terraform
.terraform.lock.hcl
backend.tf
main.tf
ecs
.terraform
.terraform.lock.hcl
alb.tf
asg.tf
backend.tf
data.tf
ec2.tf
ecs.sh
iam.tf
main.tf
OUTLINE
TIMELINE
main.tf vpc
main.tf backend
README.md
output.tf (Untracked)
output.tf
backend > main.tf > resource "aws_s3_bucket_policy" "bucket_policy" > policy
prevent_destroy = true
}
}
You, 18 hours ago | 1 author (You)
resource "aws_s3_bucket_versioning" "enabled" {
bucket = aws_s3_bucket.terraform_state.id
You, 18 hours ago | 1 author (You)
versioning_configuration {
status = "Enabled"
}
}
You, 18 hours ago | 1 author (You)
resource "aws_s3_bucket_server_side_encryption_configuration" "default" {
bucket = aws_s3_bucket.terraform_state.id
You, 18 hours ago | 1 author (You)
rule {
You, 18 hours ago | 1 author (You)
apply_server_side_encryption_by_default {
sse_algorithm = "AES256"
}
}
}
You, 18 hours ago | 1 author (You)
resource "aws_s3_bucket_public_access_block" "public_access" {
bucket = aws_s3_bucket.terraform_state.id
block_public_acls = true
block_public_policy = true
ignore_public_acls = true
restrict_public_buckets = true
}
# Create an S3 bucket
You, 3 hours ago | 1 author (You)
resource "aws_s3_bucket" "terraform_state1" {
bucket = "tax-wizard-frontend"
}
# Enable versioning for the S3 bucket
You, 3 hours ago | 1 author (You)
resource "aws_s3_bucket_versioning" "enabled_" {
bucket = aws_s3_bucket.terraform_state1.bucket
You, 3 hours ago | 1 author (You)
versioning_configuration {
status = "Enabled"
}
}
# Enable server-side encryption for the S3 bucket
PROBLEMS 2
OUTPUT
TERMINAL
PORTS
terraform - vpc
ly --auto-approve
aws_s3_bucket.terraform_state1: Refreshing state... [id=tax-wizard-frontend]
aws_s3_bucket.terraform_state: Refreshing state... [id=tax-wizard]
aws_s3_bucket_versioning.enabled: Refreshing state... [id=tax-wizard]
aws_s3_bucket_versioning.enabled_: Refreshing state... [id=tax-wizard-frontend]
aws_s3_bucket_server_side_encryption_configuration.default_: Refreshing state... [id=tax-wizard-frontend]
aws_s3_bucket_public_access_block.public_access: Refreshing state... [id=tax-wizard]
aws_s3_bucket_server_side_encryption_configuration.default: Refreshing state... [id=tax-wizard]
kingpin@kingpin-X510UNR:~/Documents/tax-wizard/infra/backend$ cd ..
kingpin@kingpin-X510UNR:~/Documents/tax-wizard/infra$ ls
backend destroy.sh ec2 ecr ecs rds README.md secret-manager vpc
kingpin@kingpin-X510UNR:~/Documents/tax-wizard/infra$ tree -I '*.terraform*'
.
kingpin@kingpin-X510UNR:~/Documents/tax-wizard/infra$ cd vpc
kingpin@kingpin-X510UNR:~/Documents/tax-wizard/infra/vpc$ terraform destroy
aws_vpc.dev-vpc: Refreshing state... [id=vpc-05b33860e7e3bc41c]
aws_subnet.subnet-1: Refreshing state... [id=subnet-0fcf3b948fd787b1d]
aws_internet_gateway.dev-igw: Refreshing state... [id=igw-0e80d0257f073dc09]
aws_subnet.subnet-2: Refreshing state... [id=subnet-038ed9613a961e406]
aws_subnet.subnet-3: Refreshing state... [id=subnet-09456c977e8f19acf]
aws_security_group.postgres_sg: Refreshing state... [id=sg-011520ca1f1da0b8e]
aws_security_group.dev-db-sg: Refreshing state... [id=sg-06dc74506ad3a1d54]
aws_route_table.route_table: Refreshing state... [id=rtb-037272920e25f7366]
aws_db_subnet_group.db-subnet-group: Refreshing state... [id=example]
aws_route_table_association.rta-1: Refreshing state... [id=rtbassoc-00deaf916306136c6]
aws_route_table_association.rta-2: Refreshing state... [id=rtbassoc-0d5709622849e4ace]
aws_route_table_association.rta-3: Refreshing state... [id=rtbassoc-02b10037e7665908e]
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
- destroy
Terraform will perform the following actions:
# aws_db_subnet_group.db-subnet-group will be destroyed
- resource "aws_db_subnet_group" "db-subnet-group" {
- arn = "arn:aws:rds:us-east-1:805071309209:subgrp:example" -> null
- description = "Managed by Terraform" -> null
- id = "example" -> null
- name = "example" -> null
- subnet_ids = [
- "subnet-038ed9613a961e406",
- "subnet-0fcf3b948fd787b1d",
] -> null
- supported_network_types = [
main
0 0 0 2
0
4 hrs 14 mins
AWS: profile:default
Share Code Link
Search Terminal Output
You, 3 hours ago
Ln 75, Col 72
Spaces: 2
UTF-8
LF
Terraform
Blackbox
Prettier

Service auto scaling - *optional*

Automatically adjust your service's desired count up and down within a specified range in response to CloudWatch alarms. You can modify your service auto scaling configuration at any time to meet the needs of your application.

☑ Use service auto scaling

Configure service auto scaling to adjust your service's desired count

Minimum number of tasks

The lower boundary to which service auto scaling can adjust the desired count of the service.

1

Maximum number of tasks

The upper boundary to which service auto scaling can adjust the desired count of the service.

5

Scaling policy

Remove

Scaling policy type

Target tracking

Policy name

tw-policy

ECS service metric

ECSServiceAverageCPUUtilization

Target value

70

Scale-out cooldown period

300

Scale-in cooldown period

300

☐ Turn off scale-in

Challenge 3: Security

- Continuous Code Quality Evaluation
- Security Vulnerability Inspection
- SonarQube (SAST Tool)
- Cloud Inspection Tools

Main Branch Status

Quality Gate

Passed

Enjoy your sparkling clean code!

See Full Analysis

Main Branch Evolution since 1 hour ago

11 Findings =

Findings | Coverage | Duplications

Bugs

0 =

Code Smells

5 =

Vulnerabilities

1 =

Security Hotspots

5 =

— Number of findings ☐ New Code

See full history

Latest Activity

NEW ANALYSIS Main Branch — Passed

November 4 at 7:20 AM `1902b05d` Add Sonarlint workflow (#11)

0 Fixed Issues · 0 New Issues · 0.0% Coverage · 0.0% Duplications · 0 Lines of Code

FIRST ANALYSIS sonarlint — Passed

November 4 at 6:16 AM `5360b850` add sonar workflow

0 Issues · 0.0% Coverage · 0.0% Duplications · 0 Lines of Code

Quality Gates

Create

Sonar way

DEFAULT BUILT-IN

Sonar way

DEFAULT BUILT-IN

Copy

Conditions

Conditions on New Code

Conditions on New Code apply to all branches and to Pull Requests.

Metric	Operator	Value
Coverage	is less than	80.0%
Duplicated Lines (%)	is greater than	3.0%
Maintainability Rating	is worse than	A
Reliability Rating	is worse than	A
Security Hotspots Reviewed	is less than	100%
Security Rating	is worse than	A

Projects

Every project not specifically associated to a Quality Gate will be associated to this one by default.

Overview

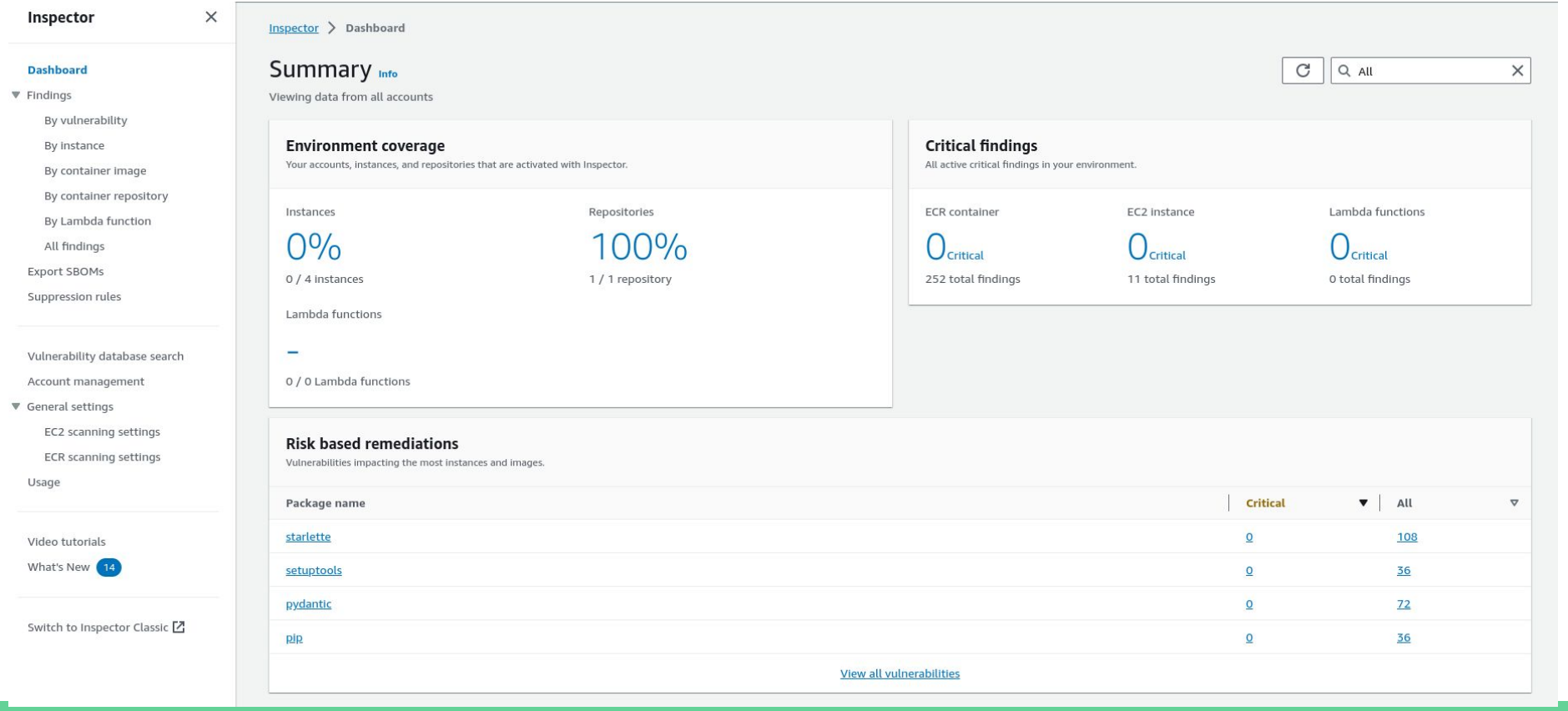
Critical	High	Medium	Low	Informational	Undefined
0	2	4	1	0	0

Vulnerabilities (7)

1

Name	Package	Severity	Description	Status	Remediation
CVE-2023-30798	starlette, starlette, starlette	HIGH	There MultipartParser usage in Encode's Starlette python framework before versions 0.25.0 allows an unauthenticated and remote attacker to specify any number of form fields or files which can cause excessive memory usage resulting in denial of service of the HTTP service.	ACTIVE	None Provided
CVE-2023-29159	starlette, starlette, starlette	HIGH	Directory traversal vulnerability in Starlette versions 0.13.5 and later and prior to 0.27.0 allows a remote unauthenticated attacker to view files in a web service which was built using Starlette.	ACTIVE	None Provided
SNYK-PYTHON-PYDANTIC-5926694	pydantic, pydantic, pydantic	MEDIUM	## Overview [pydantic](https://pypi.org/project/pydantic) is a Data validation and settings management using python 3.6 type hinting Affected versions of this package are vulnerable to Regular Expression Denial of Service (ReDoS) via the `validate_email` function due to improper limitation of mail length and the usage of an insecure regular expression. ## PoC ```py import time from pydantic import networks from pydantic.networks import validate_email start = time.time() try: exploit_string = '<' + ' ' * 3000 validate_email(exploit_string) except: pass print(f"Time elapsed: {time.time() - start}") ``` ## Details Denial of Service (DoS) describes a family of attacks, all aimed at making a system inaccessible to its original and legitimate users. There are many types of DoS attacks,	ACTIVE	None Provided

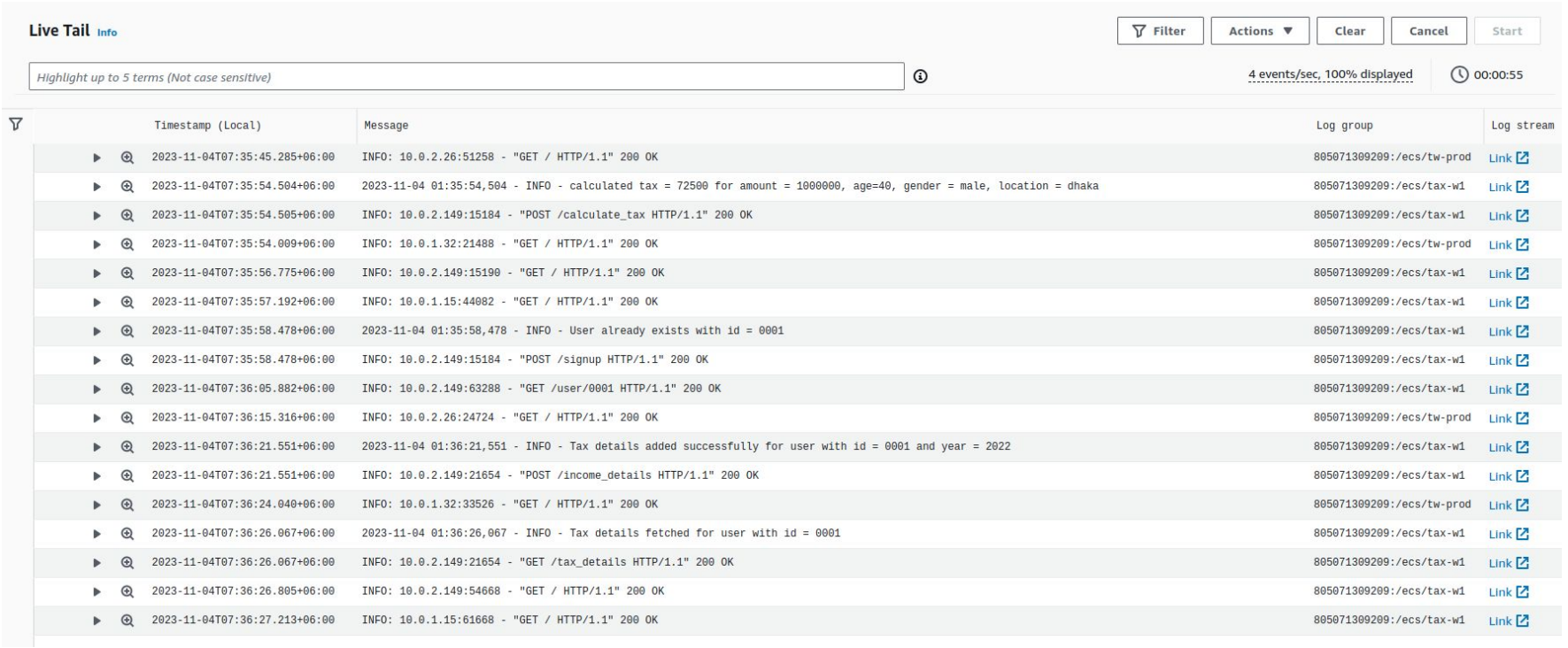
AWS Inspector



Challenge 4: Monitoring, Logging

- Real Time Monitoring (CloudWatch)
  - Logging
  - Visualization
  - Alert system

Logging



Live Tail Info

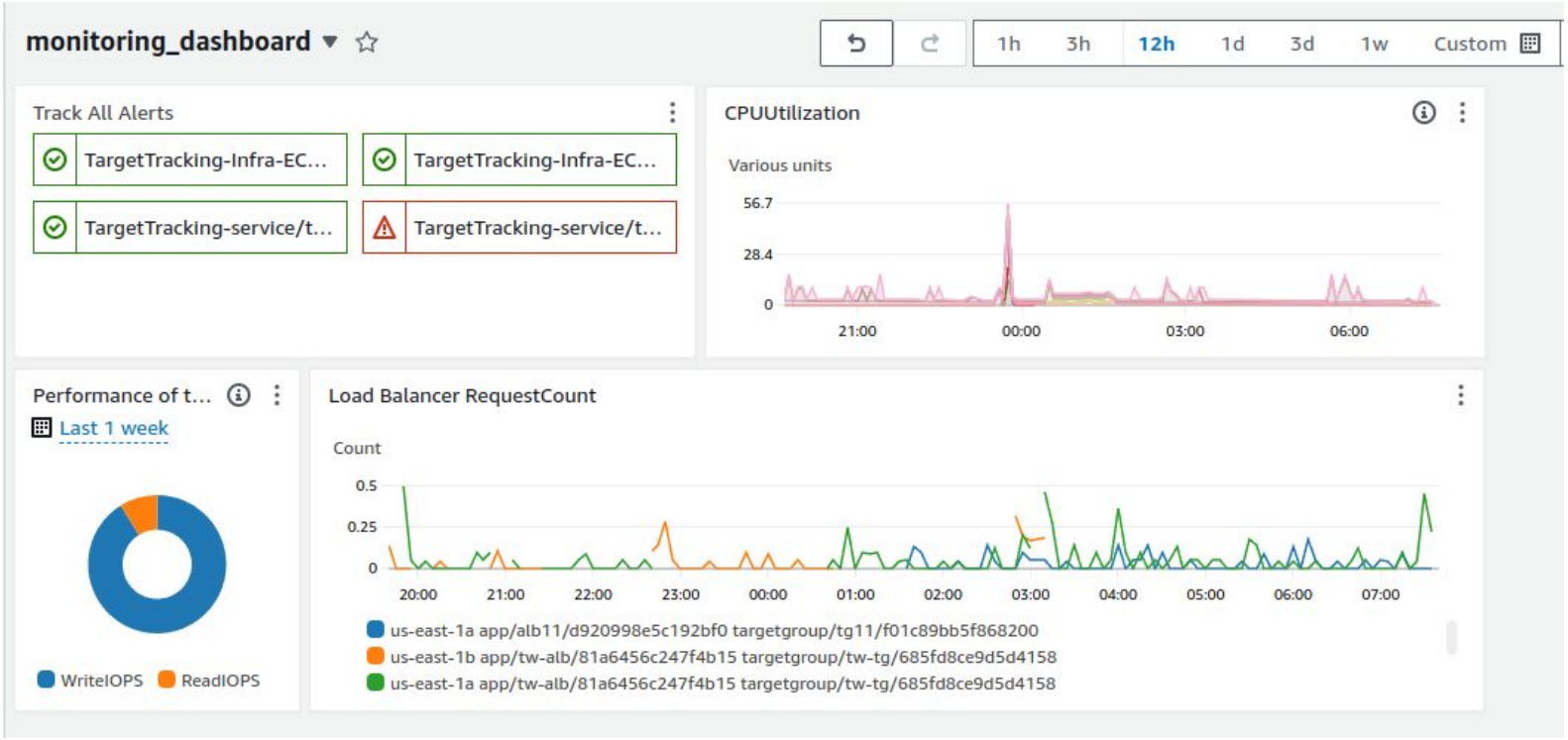
Filter | Actions | Clear | Cancel | Start

Highlight up to 5 terms (Not case sensitive)

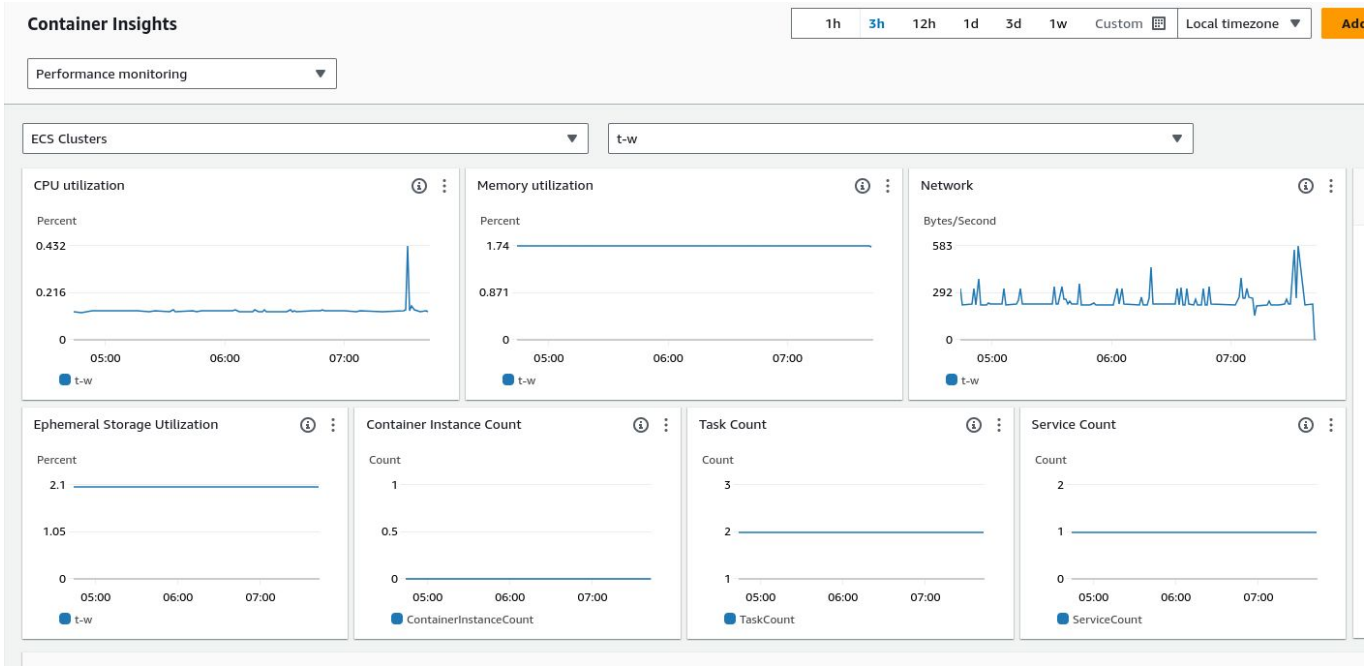
4 events/sec, 100% displayed | 00:00:55

Timestamp (Local)	Message	Log group	Log stream
2023-11-04T07:35:45.285+06:00	INFO: 10.0.2.26:51258 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tw-prod	Link
2023-11-04T07:35:54.504+06:00	2023-11-04 01:35:54,504 - INFO - calculated tax = 72500 for amount = 1000000, age=40, gender = male, location = dhaka	805071309209:/ecs/tax-w1	Link
2023-11-04T07:35:54.505+06:00	INFO: 10.0.2.149:15184 - "POST /calculate_tax HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:35:54.009+06:00	INFO: 10.0.1.32:21488 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tw-prod	Link
2023-11-04T07:35:56.775+06:00	INFO: 10.0.2.149:15190 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:35:57.192+06:00	INFO: 10.0.1.15:44082 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:35:58.478+06:00	2023-11-04 01:35:58,478 - INFO - User already exists with id = 0001	805071309209:/ecs/tax-w1	Link
2023-11-04T07:35:58.478+06:00	INFO: 10.0.2.149:15184 - "POST /signup HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:05.882+06:00	INFO: 10.0.2.149:63288 - "GET /user/0001 HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:15.316+06:00	INFO: 10.0.2.26:24724 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tw-prod	Link
2023-11-04T07:36:21.551+06:00	2023-11-04 01:36:21,551 - INFO - Tax details added successfully for user with id = 0001 and year = 2022	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:21.551+06:00	INFO: 10.0.2.149:21654 - "POST /income_details HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:24.040+06:00	INFO: 10.0.1.32:33526 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tw-prod	Link
2023-11-04T07:36:26.067+06:00	2023-11-04 01:36:26,067 - INFO - Tax details fetched for user with id = 0001	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:26.067+06:00	INFO: 10.0.2.149:21654 - "GET /tax_details HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:26.805+06:00	INFO: 10.0.2.149:54668 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link
2023-11-04T07:36:27.213+06:00	INFO: 10.0.1.15:61668 - "GET / HTTP/1.1" 200 OK	805071309209:/ecs/tax-w1	Link

Dashboard


monitoring_dashboard
1h
3h
12h
1d
3d
1w
Custom
Track All Alerts
TargetTracking-Infra-EC...
TargetTracking-Infra-EC...
TargetTracking-service/t...
TargetTracking-service/t...
CPUUtilization
Various units
56.7
28.4
0
21:00
00:00
03:00
06:00
Performance of t...
Last 1 week
WriteIOPS
ReadIOPS
Load Balancer RequestCount
Count
0.5
0.25
0
20:00
21:00
22:00
23:00
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
us-east-1a app/alb11/d920998e5c192bf0 targetgroup/tg11/f01c89bb5f868200
us-east-1b app/tw-alb/81a6456c247f4b15 targetgroup/tw-tg/685fd8ce9d5d4158
us-east-1a app/tw-alb/81a6456c247f4b15 targetgroup/tw-tg/685fd8ce9d5d4158

Container Insights



Things we tried to do

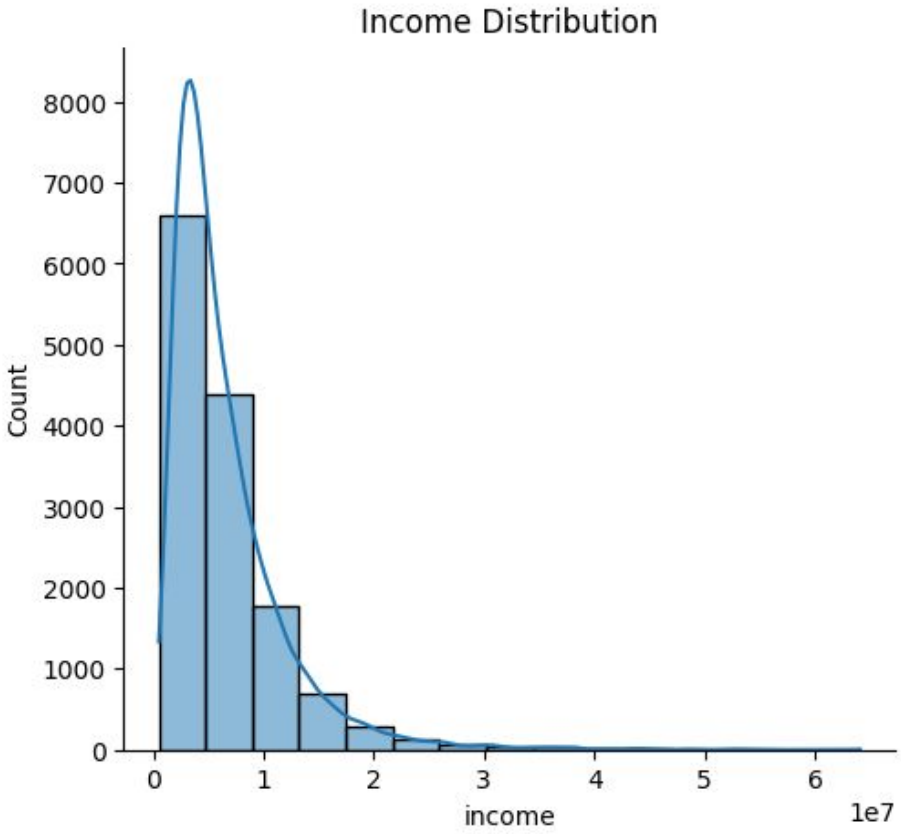
- Create Metrics from Regex Pattern, Create alerts from those and send notification through SNS.

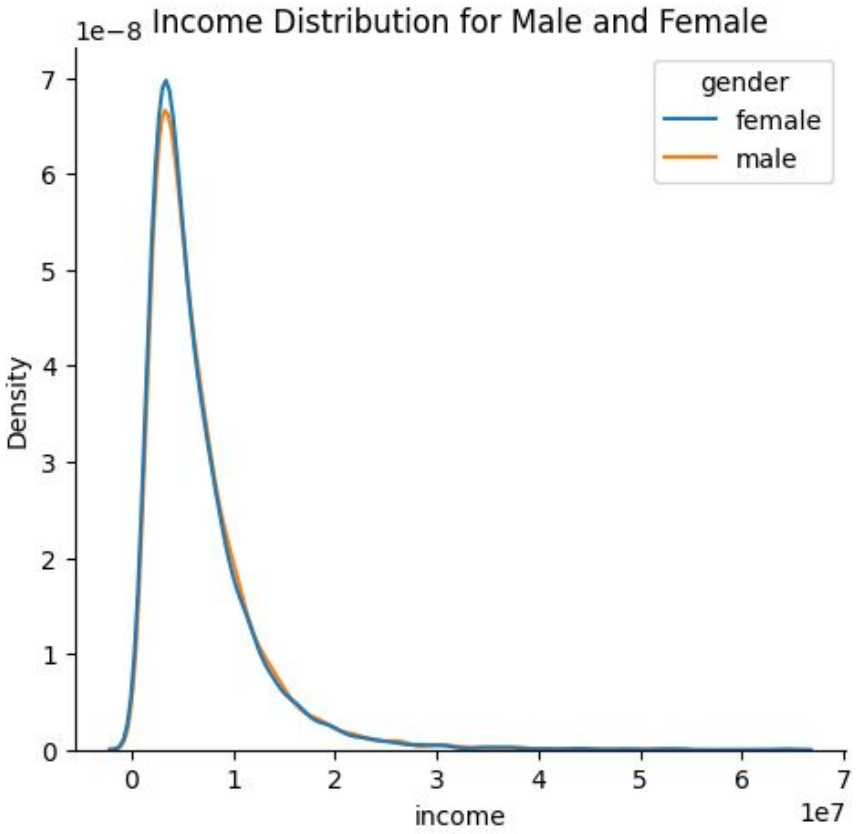
Bonus Task - Analytics and Forecasting using ML

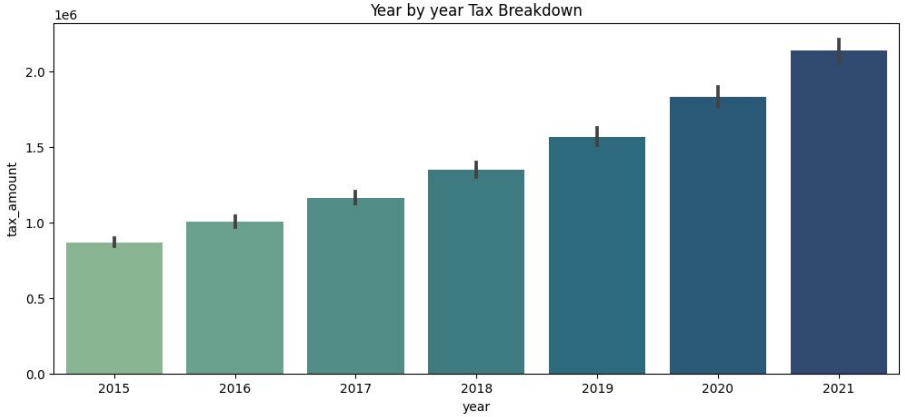
Motivation

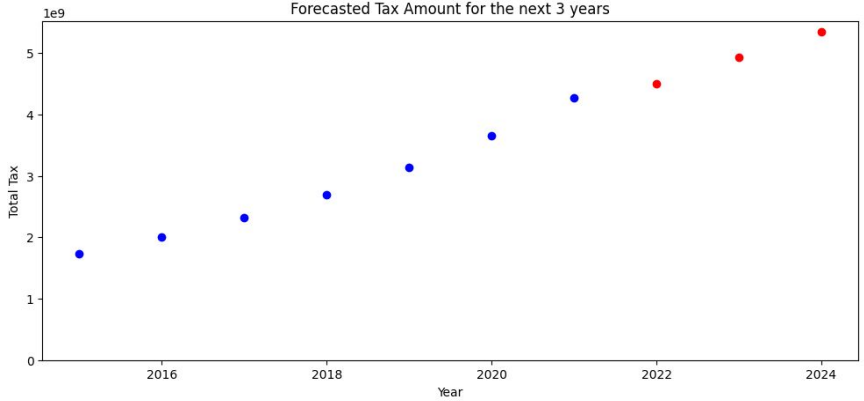
From our collected data, a lot of socio economic aspects can be analysed

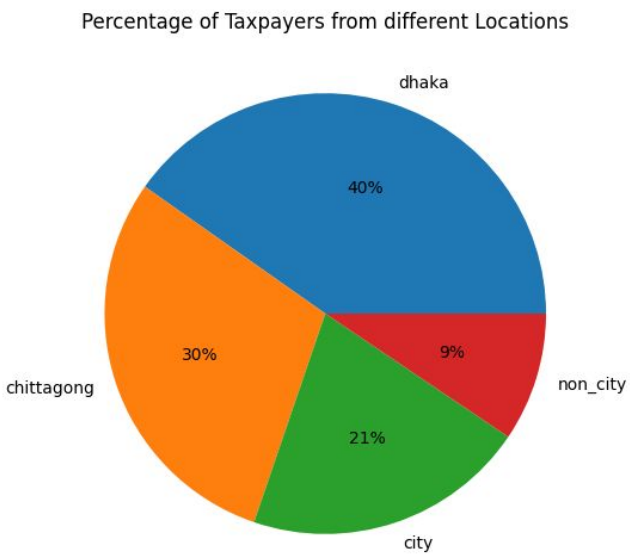
1. Income Distribution
2. Gender equality in payment
3. Economic Growth (Using Income)
4. Tax Forecasting using ML (Will help in planning)


Income Distribution
Count
income
1e7


Income Distribution for Male and Female
1e−8
Density
income
1e7
gender
female
male


Year by year Tax Breakdown
1e6
tax_amount
year
2015
2016
2017
2018
2019
2020
2021


Forecasted Tax Amount for the next 3 years
1e9
Total Tax
Year


Percentage of Taxpayers from different Locations
dhaka
40%
chittagong
30%
city
21%
non_city
9%

https://taxwizard-buet-daredevils.github.io/

Deployed using **Github pages**, updated using **Github actions**

Thanks! Any Question?