**Project Report: Deploying an Apache Web Server on AWS with Custom VPC Configuration**

**Executive Summary**

This report details the implementation of an Apache Web Server deployment on AWS within a custom Virtual Private Cloud (VPC). The project was meticulously documented using GitHub to ensure transparency and version control. The tasks involved configuring a secure and scalable cloud infrastructure, launching an EC2 instance, and setting up a web server. Each step demonstrated proficiency in cloud resource management, networking, and software deployment.

**Introduction**

The primary objective of this project was to design and deploy a scalable web application environment on AWS. It involved:

1. Setting up version control using GitHub for documentation and collaboration.

2. Configuring a custom VPC with subnets, an Internet Gateway, and security policies.

3. Launching an EC2 instance and installing Apache Web Server.

This report outlines the implementation process, key configurations, and final outcomes.

**Project Objectives**

1. Utilize GitHub for collaborative documentation and version control.

2. Create a custom VPC to manage networking and resource allocation.

3. Deploy and configure an EC2 instance with Apache Web Server.

**Implementation Details**

**1. Setting Up GitHub for Documentation and Version Control**

1.1 GitHub Account Creation

* Registered a GitHub account using a professional username (‘ayoolakehinde92@gmail.com and onabusola’).

1.2 Git Installation and Configuration

- Installed Git locally from [GitHub Desktop](<https://github.com/apps/desktop/>).

1.3 Repository Creation and Cloning

- Created a new public repository named ‘AWS-Web-Application’.

- Initialized the repository with a ‘onabusola/README.md’ file.

- and commit to save changes on the repository

- and keep adding my file

**2. Configuring a Custom VPC**

2.1 VPC Creation

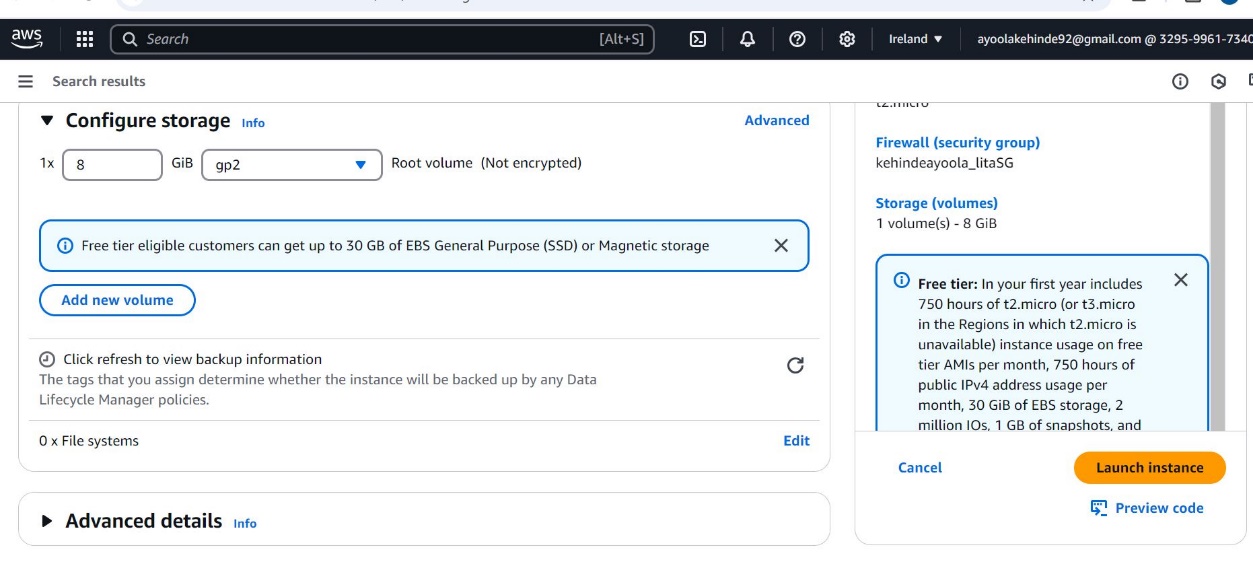
* I used the created VPC named ‘Lita\_project-vpc’ with CIDR block ‘10.0.0.0/16’.

2.2 Subnets Configuration

- two subnets was created:

- Public Subnet: CIDR block ‘10.0.1.0/24’. Using [subnet-0ecd659b88e6ad1d3 (Lita\_project-subnet-public1-eu-west-1](https://eu-west-1.console.aws.amazon.com/vpc/home?region=eu-west-1#subnets:subnetId=subnet-0ecd659b88e6ad1d3)

- Private Subnet: CIDR block ‘10.0.2.0/24’.



2.3 Internet Gateway and Route Table

- Created and attached an Internet Gateway to ‘Lita\_project’.

- Configured the route table for the public subnet to allow internet access (‘0.0.0.0/0’).

2.4 Network Access Control Lists (NACLs)

* Configured NACLs to allow HTTP (port 80) and SSH (port 22) traffic for the public subnet.

2.5 Security Groups

-I created a Security Group to:

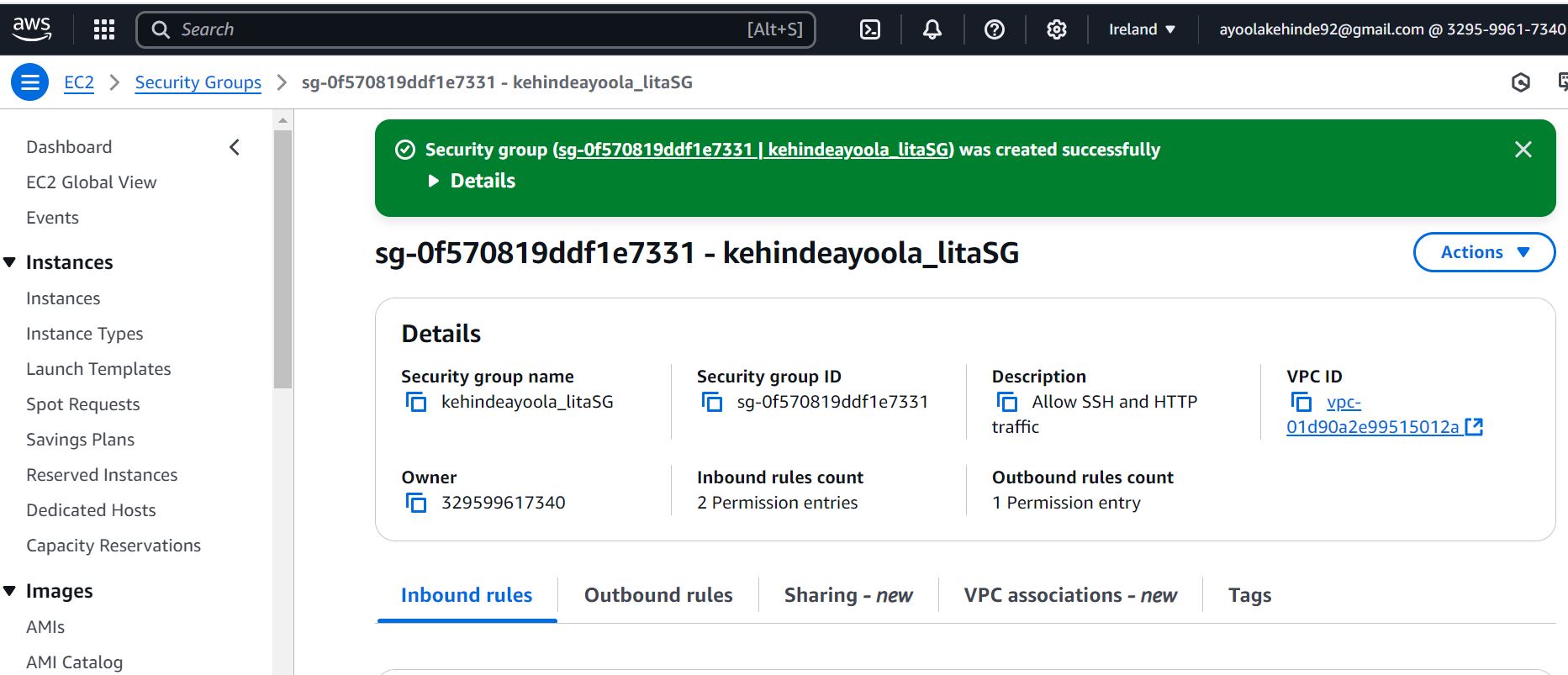
-I allowed HTTP traffic (port 80) from all IP addresses.

-I allowed SSH traffic (port 22) to all IP address.

A screenshot of a computer

Description automatically generated

Created a name for my security group and allowed SSH and HTTP traffic



Security group created successfully

**3. Launching an EC2 Instance and Deploying Apache**

3.1 EC2 Instance Launch

- Launched an EC2 instance named ‘kehindeayoola\_lita’ with the following configurations:

- AMI: Amazon Linux 2.

- Instance Type:t2. micro.

- Key Pair: I Created and downloaded security key as ‘kehindeayoolaona\_litakp’.

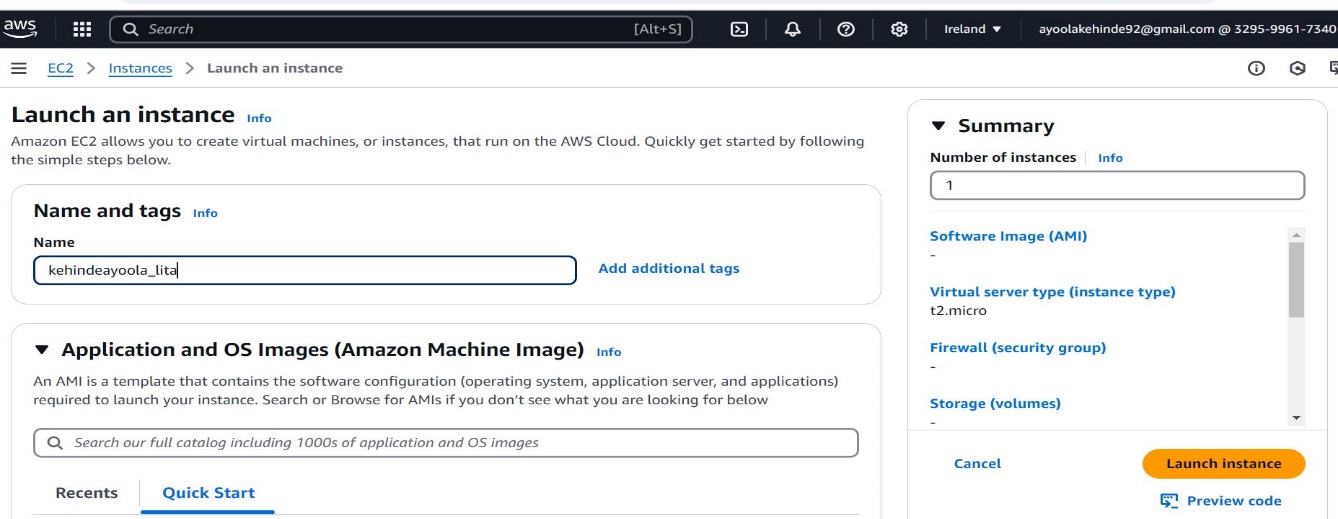
- Networking:

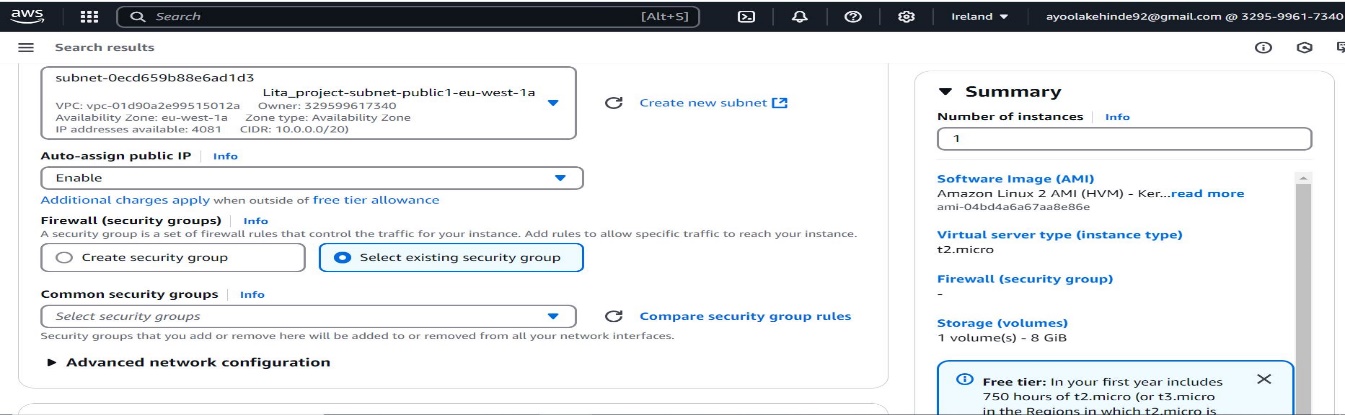
- Associated the instance with ‘kehindeayoola\_lita’ and the Lita project public subnet.

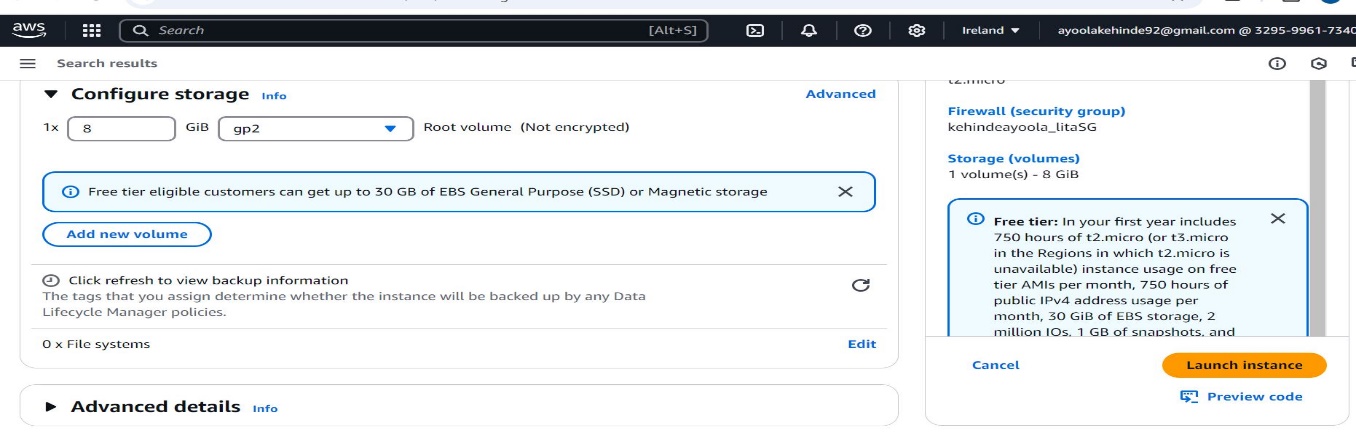
- Enabled auto-assign Public IP.

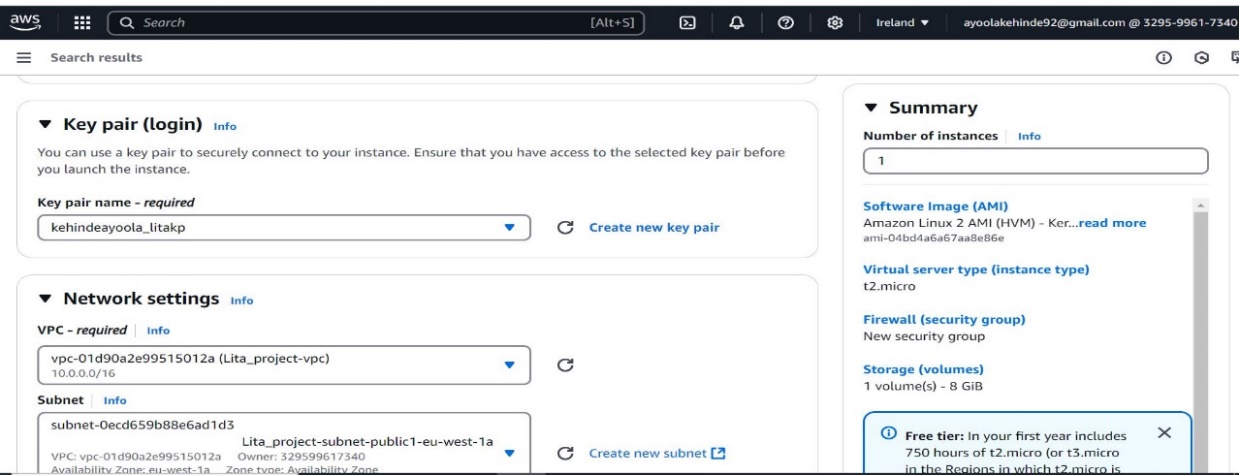
- Attached the previously created Security Group.

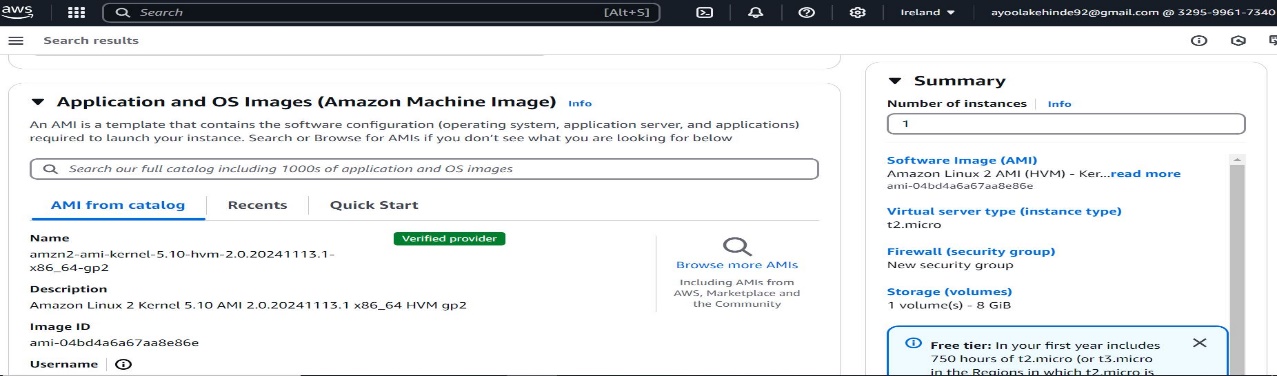
The diagrams below shows the process involved in launching EC2 instance ;

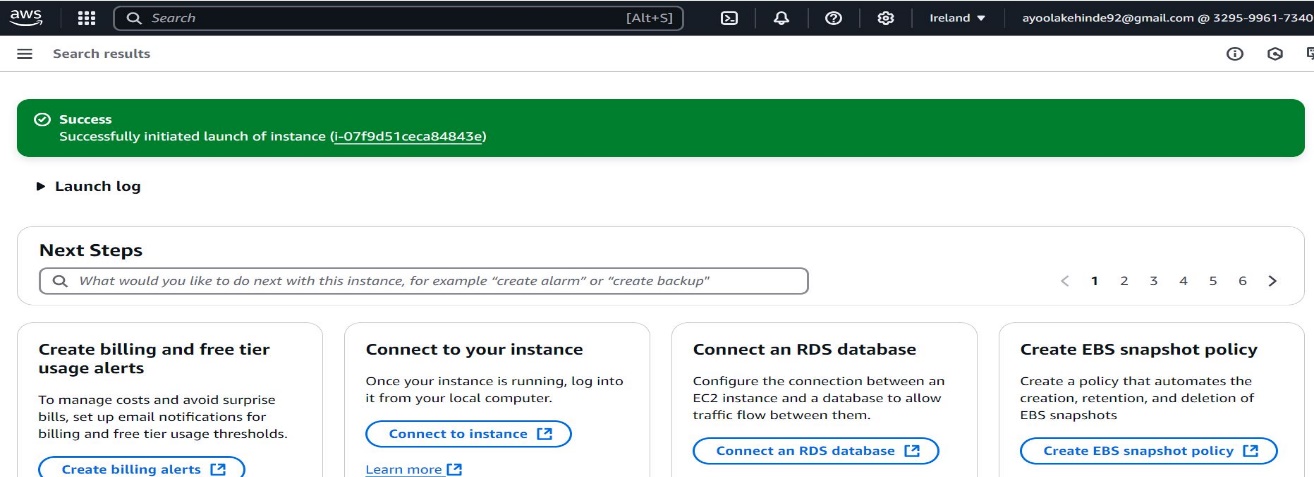


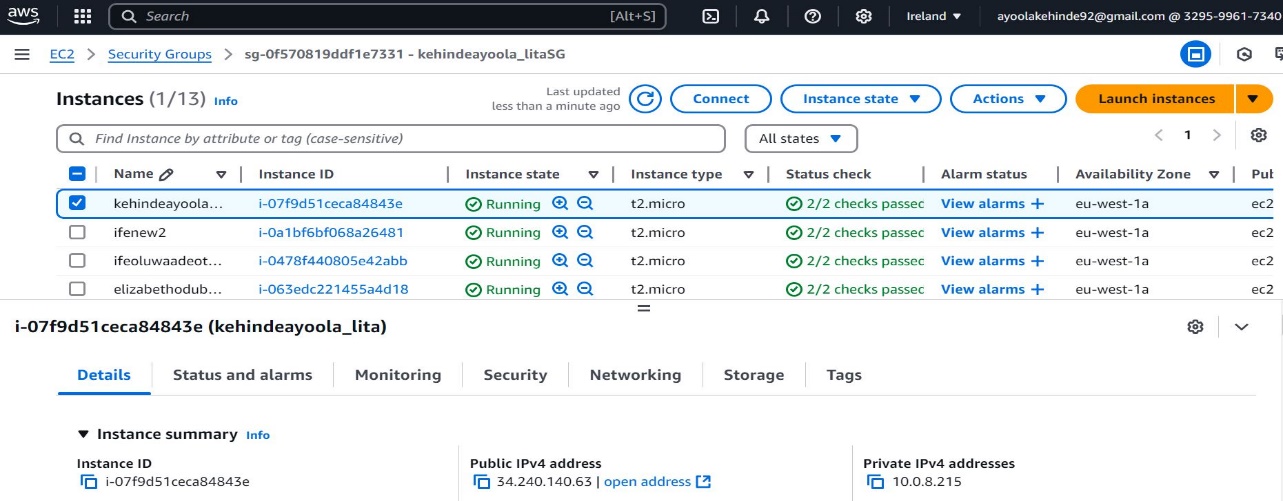












3.2 Apache Web Server Installation

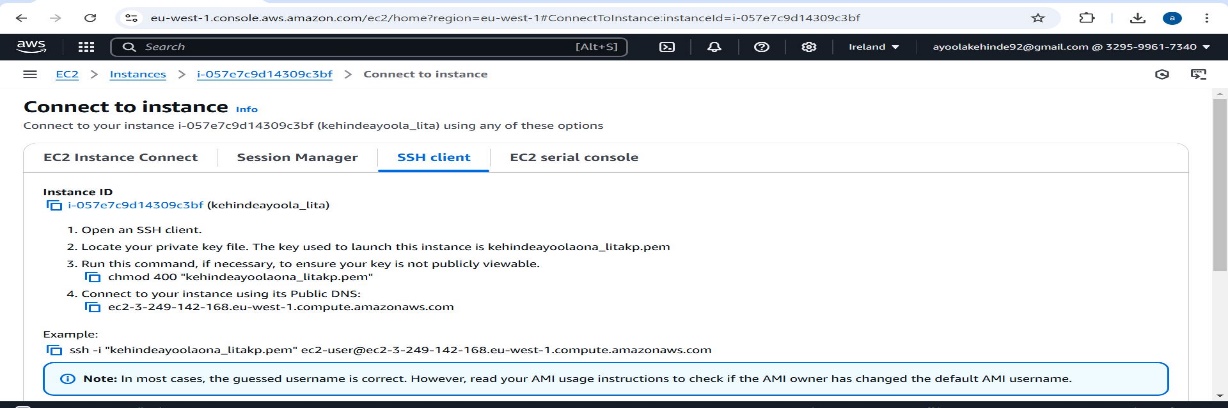
* Accessed the instance via SSH client
* Installed and enabled Apache using

Sudo yum update -y

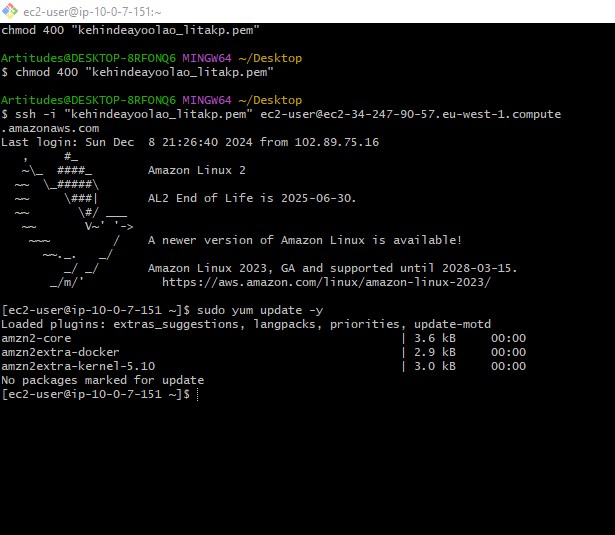
Sudo yum install httpd -y

Sudo systemctl start httpd

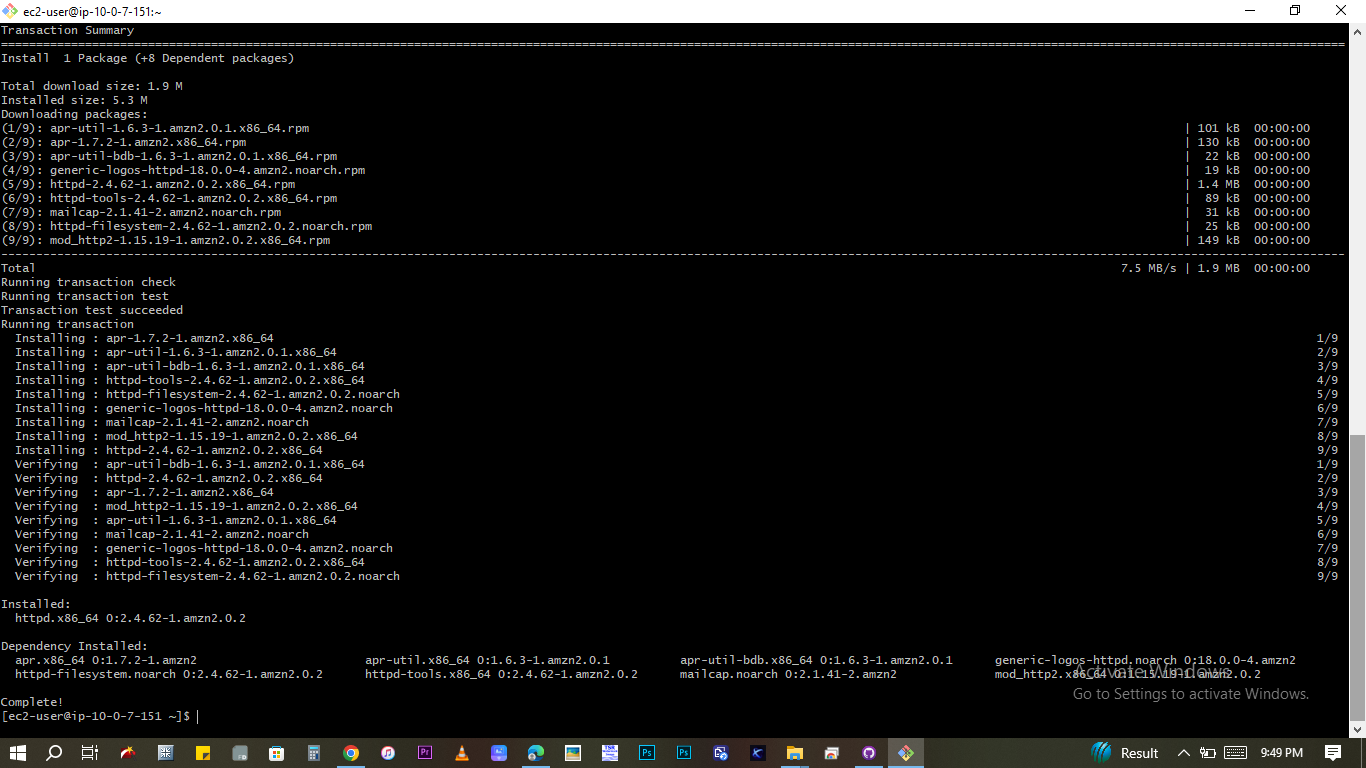
Sudo systemctl enable httpd



I accessed the instance via SSH client



I Installed and enabled Apache



3.3 Verification

* Verified the Apache installation by entering the public IP **3.249.142.168** of the instance in a web browser.

A screenshot of a computer

Description automatically generated

I verified that Apache has been installed

**4. Documenting and Pushing Final Results to GitHub**

4.1 Documentation Updates

- Updated ‘README.md’ with:

- Detailed project description.

- Screenshots of key configurations and outputs.

- Overview of AWS services used and their purpose.

- Committed and pushed final updates to GitHub

**5.0 Challenges and Lessons Learned**

* **Challenges:**

- Configuring the network for secure yet functional access.

- Ensuring all dependencies were installed correctly for Apache.

* **Lessons Learned:**

- Gained practical experience with GitHub for version control and documentation.

- Acquired foundational skills in AWS networking, resource allocation, and instance management.

**Conclusion**

The project was successfully completed, demonstrating proficiency in configuring cloud infrastructure, deploying web services, and maintaining detailed project documentation. The deployed Apache Web Server is fully functional and accessible. The documentation is available on GitHub for review and further improvement.

GitHub Repository

Access the complete project documentation here: [AWS-Web-Application Repository](<https://github.com/ayoolakehinde92/AWS-Web-Application>).