

# **AWS 3-TIER WEB ARCHITECTURE DOCUMENTATION**

---

*Designed and implemented by Aditya Ghatol*

## Table of Contents

Index	contents	Page no
1	Introduction	2
2	Architecture Overview	3
3	Technologies Used	4
4	Step-by-Step Implementation	5
4.1	EC2 Instance Setup	6
4.2	Application Load Balancer (ALB) Setup	10
4.3	Auto Scaling Group Setup	11
4.4	Amazon RDS Setup	20
5	Conclusion	24

## 1. Introduction

In today's landscape, where web applications are crucial for business operations and user engagement, building scalable and resilient applications is more important than ever. This project focuses on designing and implementing a 3-tier architecture for a web application using Amazon Web Services (AWS), a leading cloud computing platform renowned for its reliability, flexibility, and extensive service offerings.

- **Presentation Layer:** Hosted on Amazon EC2, this layer manages the user interface and experience.
- **Application Layer:** Utilizes Auto Scaling Groups to handle application logic and dynamically adjust capacity based on demand.
- **Database Layer:** Managed by Amazon RDS, it provides reliable data storage with high availability and automated backups.

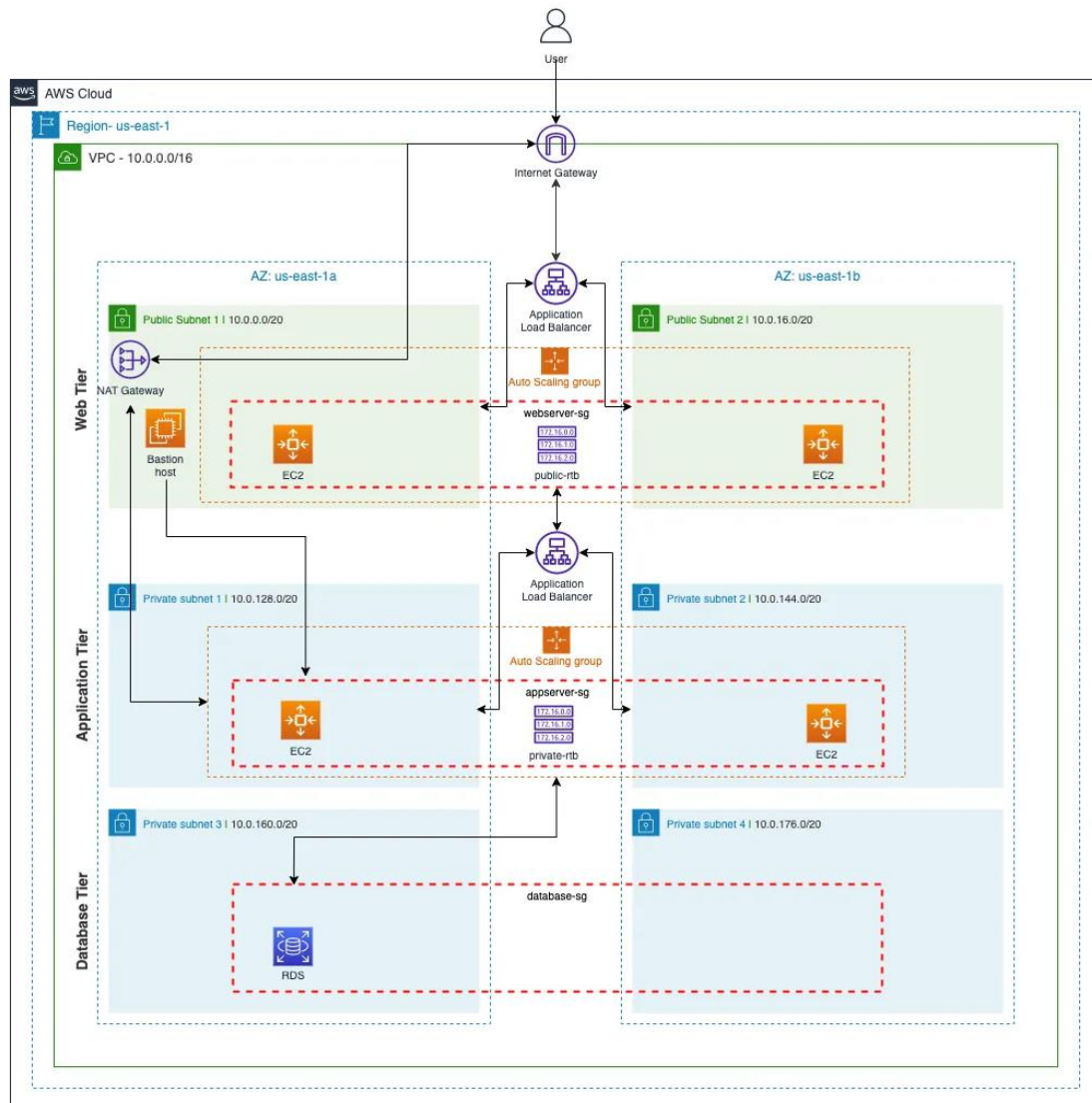
An Application Load Balancer (ALB) is employed to distribute traffic across multiple instances, preventing any single instance from becoming a bottleneck. This ensures the application remains responsive even under high traffic conditions.

By integrating these AWS services, the project designs a resilient and scalable web application that meets modern standards for availability and security. It addresses key challenges such as managing fluctuating workloads, maintaining performance, and protecting user data. Following AWS best practices, this project demonstrates how AWS's robust cloud services can be leveraged to build a powerful 3-tier architecture that exceeds contemporary web application expectations. This version keeps the essential details while improving readability.

## 2. Architecture Overview

The AWS 3-Tier architecture consists of the following components:

1. Web Tier: Amazon EC2 instance to host the frontend.
2. Application Tier: Application Load Balancer (ALB) to route traffic to instances.
3. Database Tier: Amazon RDS MySQL for persistent data storage.



### 3. Technologies Used

- **Amazon EC2 (Elastic Compute Cloud):** Provides scalable compute capacity in the cloud, enabling the hosting of web applications. EC2 instances run the application code and handle the presentation layer, offering flexibility to choose different instance types based on performance needs.
- **Auto Scaling Groups:** Automatically adjusts the number of EC2 instances in response to varying traffic loads. This technology ensures that the application can scale up to handle high traffic and scale down during lower demand periods, optimizing cost and maintaining performance.
- **Application Load Balancer (ALB):** Distributes incoming application traffic across multiple EC2 instances. ALB improves the application's availability and fault tolerance by ensuring that no single instance bears too much load. It also supports advanced routing features, such as path-based routing, to direct traffic to specific instances based on URL paths.
- **Amazon RDS (Relational Database Service):** A managed database service that simplifies database management tasks such as backups, patching, and scaling. Amazon RDS supports MySQL, providing a reliable and secure database solution for storing application data. It ensures high availability through Multi-AZ deployments and automated backups, reducing administrative overhead and enhancing data durability.

## 4. Step-by-Step Implementation

### 4.1 EC2 Instance Setup

1. Launch an EC2 instance with Amazon Linux 2 AMI.

### 4.2 Application Load Balancer (ALB) Setup

1. Create an Application Load Balancer on the AWS console.
2. Configure it to listen on port 80 and distribute traffic across EC2 instances.
3. Test ALB by accessing the DNS name in a browser.

### 4.3 Auto Scaling Group Setup

1. Create an Auto Scaling Group for EC2 instances.
2. Set scaling policies to handle traffic spikes by adding or removing instances based on CPU utilization.

### 4.4 Amazon RDS Setup

1. Create a MySQL database on RDS

## 5. Best Practices

- Security: Configured security groups to allow only necessary traffic.
- High Availability: Auto Scaling ensures the system handles load efficiently.
- Fault Tolerance: ALB distributes traffic across healthy instances.
- Backup and Recovery: Enabled automated backups for RDS.

## 6. Screenshots

Insert relevant screenshots here for each step (EC2 setup, ALB, Auto Scaling, RDS).

aws

Services

Search

VPC > Your VPCs > Create VPC

## Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances, Amazon S3 buckets, and Amazon ElastiCache instances.

### VPC settings

#### Resources to create [Info](#)

Create only the VPC resource or the VPC and other networking resources.

☐ VPC only

☒ VPC and more

#### Name tag auto-generation [Info](#)

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☒ Auto-generate

Aditya-webApp

#### IPv4 CIDR block [Info](#)

Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/1665,536 IPs

CIDR block size must be between /16 and /28.

#### IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block

☐ Amazon-provided IPv6 CIDR block

CloudShellFeedback

aws

Services

Search

Number of Availability Zones (AZs) [Info](#)

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

123

Customize AZs

Number of public subnets [Info](#)

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

02

Number of private subnets [Info](#)

The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

024

Customize subnets CIDR blocks

NAT gateways (\$) [Info](#)

Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway

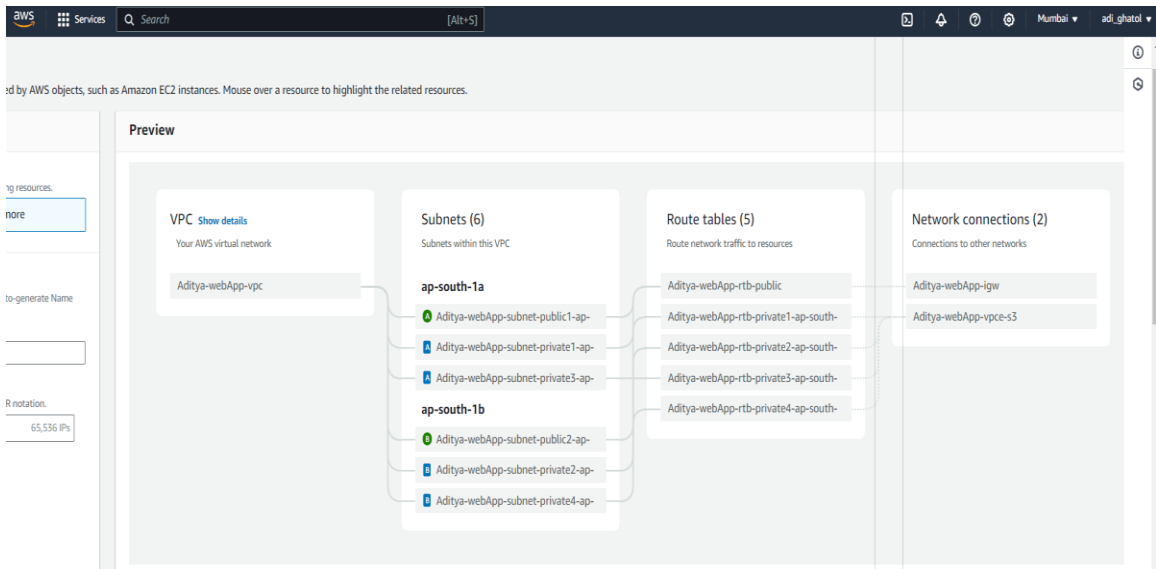
NoneIn 1 AZ1 per AZ

VPC endpoints [Info](#)

Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

NoneS3 Gateway

CloudShellFeedback



The screenshot shows the 'Edit subnet settings' page in the AWS VPC console. The breadcrumb trail is: VPC > Subnets > subnet-0c178908f6c01cbdf > Edit subnet settings.

### Edit subnet settings

**Subnet**

Subnet ID	Name
subnet-0c178908f6c01cbdf	-

**Auto-assign IP settings**

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

- ☒ Enable auto-assign public IPv4 address
- ☐ Enable auto-assign customer-owned IPv4 address

**Resource-based name (RBN) settings**

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

- ☐ Enable resource name DNS A record on launch
- ☐ Enable resource name DNS AAAA record on launch

Hostname type

☐ Resource name

☒ IP name



The screenshot shows the AWS VPC dashboard with the 'Route tables' section active. The 'Aditya-webApp-rtb-public' route table is selected, and the 'Set main route table' action is highlighted in the 'Actions' menu.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
-	rtb-0311a5aa83f44209b	-	-	Yes	vpc-059
Aditya-webApp-rtb-private4-ap-south-1b	rtb-0082a7464ab1af6d5	subnet-04f5c33c9ebe496...	-	No	vpc-0ed
<b>Aditya-webApp-rtb-public</b>	<b>rtb-06deb7393c3d36773</b>	<b>2 subnets</b>	-	No	vpc-0ed
Aditya-webApp-rtb-private2-ap-south-1b	rtb-0d9f22682980f1c95	subnet-08f7b08a8ccbf5...	-	No	vpc-0ed
Aditya-webApp-rtb-private1-ap-south-1a	rtb-017ee90d8b09aade	subnet-0d754cb9364aba...	-	No	vpc-0ed
-	rtb-08a0a73ac4691a8be	-	-	Yes	vpc-0ed298f895034a050   Adit...
Aditya-webApp-rtb-private3-ap-south-1a	rtb-0ca7d8da730d52f62	subnet-09b8be998eef15...	-	No	vpc-0ed298f895034a050   Adit...

✓ Elastic IP address 13.202.206.180 (eipalloc-0dd2ff6de7951ac3d) allocated.

[VPC](#) > [NAT gateways](#) > Create NAT gateway

## Create NAT gateway [Info](#)

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

### NAT gateway settings

#### Name - *optional*

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Subnet

Select a subnet in which to create the NAT gateway.

#### Connectivity type

Select a connectivity type for the NAT gateway.

- ☒ Public  
☐ Private

#### Elastic IP allocation ID [Info](#)

Assign an Elastic IP address to the NAT gateway.

[Allocate Elastic IP](#)

► [Additional settings](#) [Info](#)

### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter

Services Search [Alt+S] Number ad\_ghatol

VPC > Route tables > rtb-017ee90d8b909aade > Edit routes

### Edit routes

Destination	Target	Status	Propagated
pl-78a54011 10.0.0.0/16	vpce-0be223220c9f04e7	Active	No
	local	Active	No
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="local"/>		
	NAT Gateway		No
	<input type="text" value="nat-017689e13a6b75af6d"/>		

[Add route](#)

[Cancel](#) [Preview](#) [Save changes](#)



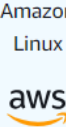
## ▼ Application and OS Images (Amazon Machine Image) - required [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below


 Search our full catalog including 1000s of application and OS images

Recents


Quick Start




Amazon Linux




macOS



Ubuntu




Windows



Red Hat



SUSE Li



Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

### Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

Free tier eligible ▼

ami-08718895af4dfa033 (64-bit (x86), uefi-preferred) / ami-0083e0c040551216d (64-bit (Arm), uefi)

Virtualization: hvm    ENA enabled: true    Root device type: ebs

### Description

Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

▼ Instance type [Info](#) | [Get advice](#) Advanced

Instance type

t2.micro

Family: t2    1 vCPU    1 GiB Memory    Current generation: true    Free tier eligible

On-Demand Linux base pricing: 0.0124 USD per Hour

On-Demand Windows base pricing: 0.017 USD per Hour

On-Demand RHEL base pricing: 0.0268 USD per Hour

On-Demand SUSE base pricing: 0.0124 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name

MyKeyPair ▼ [Create new key pair](#)

▼ Network settings [Info](#)

Subnet [Info](#)

Don't include in launch template ▼ [Create new subnet](#)

When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your

▼ Summary

Software Image (AMI)

Amazon Linux 2023 AMI 2023.5.2...[read more](#)

ami-08718895af4dfa033

Virtual server type (instance type)



t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

 **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. 

Cancel

Create launch template

11

### Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 152.57.235.161/32)

Remove

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>
ssh	TCP	22
Source type <a href="#">Info</a>	Name <a href="#">Info</a>	Description - optional <a href="#">Info</a>
My IP	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>
	<input type="text" value="152.57.235.161/32"/>	

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>
HTTP	TCP	80
Source type <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>
Anywhere	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>
	<input type="text" value="0.0.0.0/0"/>	

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Add security group rule

### ▼ Summary

#### Software Image (AMI)

Amazon Linux 2023 AMI 2023.5.2...[read more](#)  
ami-08718895af4dfa033

#### Virtual server type (instance type)

t2.micro

#### Firewall (security group)

New security group

#### Storage (volumes)

1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Create launch template

### Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 152.57.235.161/32)

Remove

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>
ssh	TCP	22
Source type <a href="#">Info</a>	Name <a href="#">Info</a>	Description - optional <a href="#">Info</a>
My IP	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>
	<input type="text" value="152.57.235.161/32"/>	

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>
HTTP	TCP	80
Source type <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>
Anywhere	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>
	<input type="text" value="0.0.0.0/0"/>	

▼ Security group rule 3 (TCP, 443, 0.0.0.0/0)

Remove

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>
HTTPS	TCP	443
Source type <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>
Anywhere	<input type="text" value="Add CIDR, prefix list or security"/>	<input type="text" value="e.g. SSH for admin desktop"/>
	<input type="text" value="0.0.0.0/0"/>	

#### Software Image (AMI)

Amazon Linux 2023 AMI 2023.5.2...[read more](#)  
ami-08718895af4dfa033

#### Virtual server type (instance type)

t2.micro

#### Firewall (security group)

New security group

#### Storage (volumes)

1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Create launch template

### Automatic scaling - optional

Choose whether to use a target tracking policy [Info](#)

You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

☐ No scaling policies

Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

☒ Target tracking scaling policy

Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name

Target Tracking Policy

Metric type [Info](#)

Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.

Average CPU utilization ▼

Target value

50

Instance warmup [Info](#)

300

seconds

☐ Disable scale in to create only a scale-out policy

[EC2](#) > [Auto Scaling groups](#) > Create Auto Scaling group

Step 1  
[Choose launch template](#)

Step 2  
[Choose instance launch options](#)

Step 3 - optional  
[Configure advanced options](#)

Step 4 - optional  
[Configure group size and scaling](#)

Step 5 - optional  
[Add notifications](#)

Step 6 - optional  
[Add tags](#)

Step 7  
**Review**

## Review [Info](#)

### Step 1: Choose launch template [Edit](#)

#### Group details

Auto Scaling group name  
Aditya\_ASG

#### Launch template

Launch template	Version	Description
<a href="#">Aditya-WebServer</a> <a href="#">🔗</a>	Default	Aditya Web Server
<a href="#">lt-0abe7499aaf4e447a</a>		

### Step 2: Choose instance launch options [Edit](#)

#### Network

##### Network

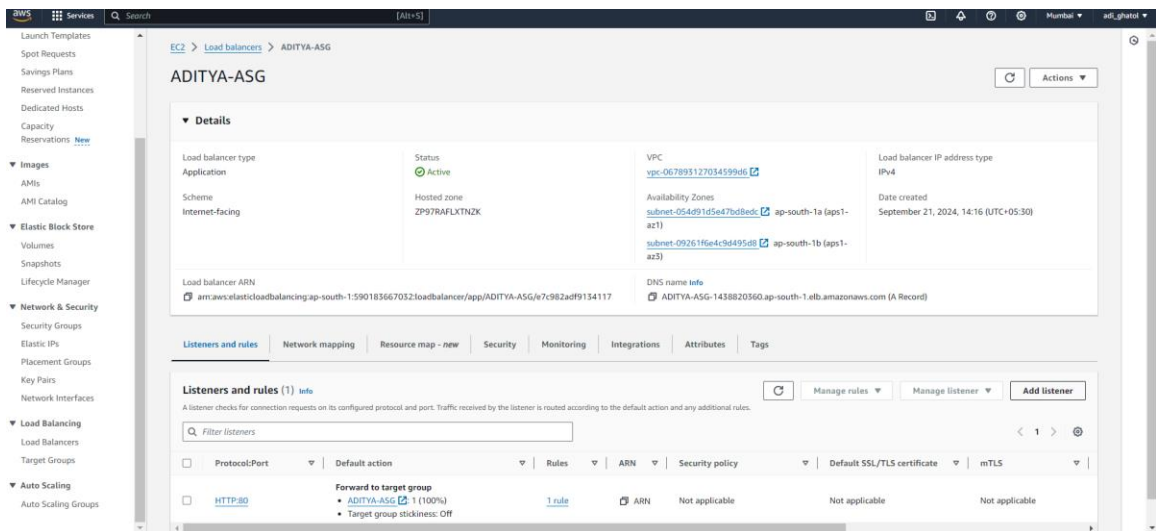
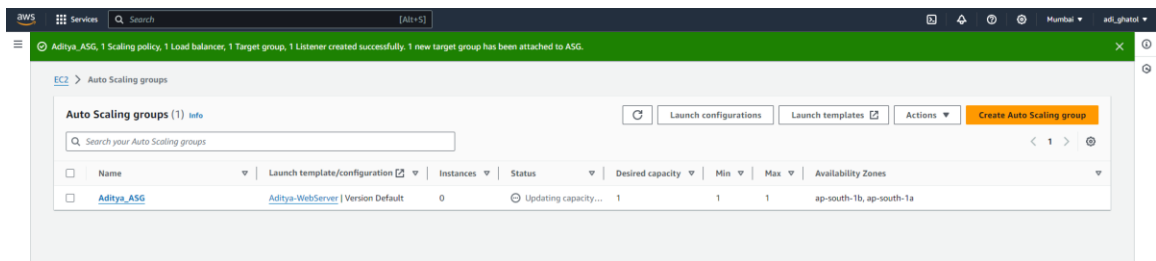
VPC

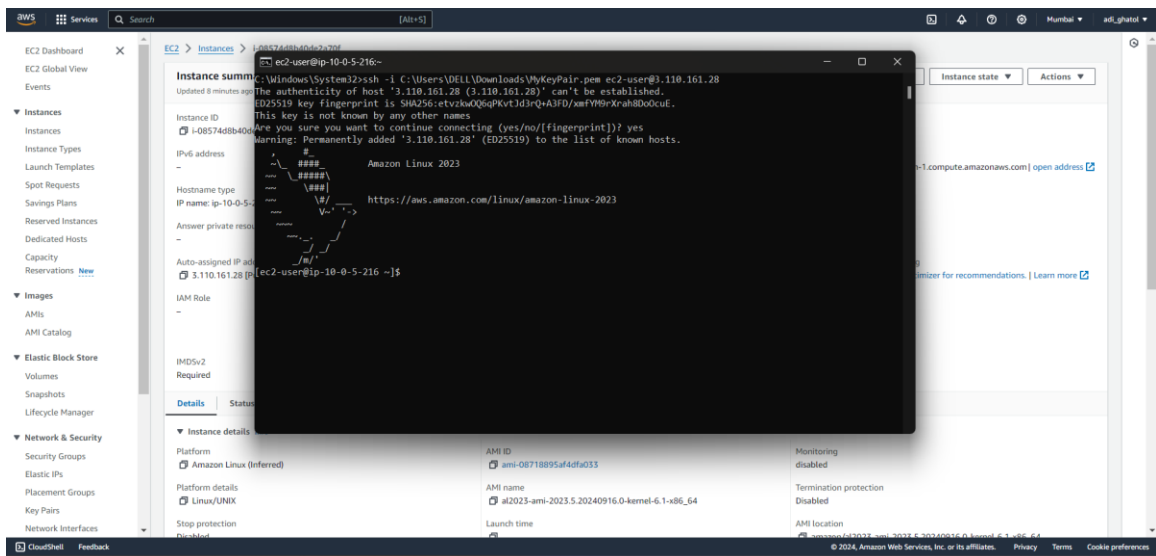
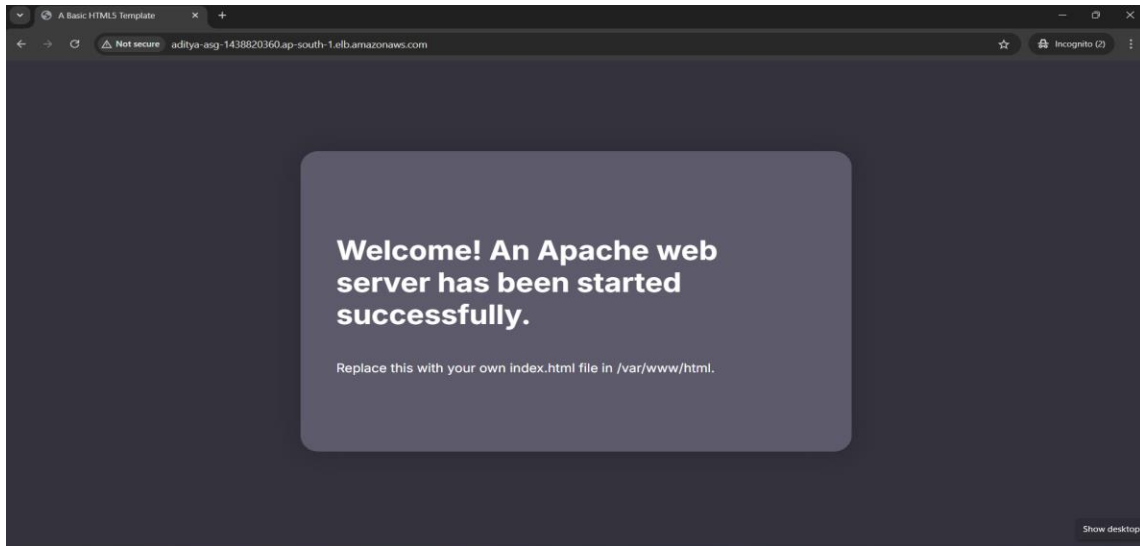
[vpc-067893127034599d6](#) [🔗](#)

Availability Zone

Subnet

ap-south-1a	<a href="#">subnet-054d91d5e47bd8edc</a> <a href="#">🔗</a>	10.0.0.0/20
ap-south-1b	<a href="#">subnet-09261f6e4c9d495d8</a> <a href="#">🔗</a>	10.0.16.0/20





Success!

We've successfully built the architecture for the Web Tier for our Brainiac application! Remember, this is the 'Presentation' layer, where our users will directly interact with our app.



## Tier 2: Application tier (Backend)

The Application Tier is essentially where the heart of our Brainiac app lives. This is where the source code and core operations send/retrieve data to/from the Web and Database tiers.

The structure is very similar to the Web Tier but with some minor additions and considerations.

What we will build:

A launch template to define the type of EC2 instances.

An Auto Scaling Group (ASG) to dynamically provision EC2 instances.

An Application Load Balancer (ALB) to route traffic from the Web tier.

A Bastion host to securely connect to our application servers.

The screenshot displays the AWS Management Console interface for creating a launch template. The 'Network settings' section is active, showing options for Subnet, Firewall (security groups), Security group name, Description, and VPC. A dropdown menu for 'Security groups' is open, listing various groups including 'launch-wizard-2' through 'launch-wizard-10' and 'Aditya-webserver-SG'. The 'Summary' section on the right provides an overview of the configuration: Software Image (AMI) as 'Amazon Linux 2023.5.2', Virtual server type (instance type) as 't2.micro', Firewall (security group) as 'New security group', and Storage (volumes) as '1 volume(s) - 8 GiB'. A 'Free tier' notification box is also visible, stating that the first year includes 750 hours of t2.micro (or t3.micro) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. At the bottom, there are 'Cancel' and 'Create launch template' buttons.





aws

Services

Search

[Alt+S]

EC2

Instances

Launch an instance

## Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

### Name and tags Info

Name

Aditya-bastionHost

Add additional tags

### Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUS

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

Free tier eligible

### Summary

Number of instances

Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.5.2...read more

ami-08718895af4dfa033

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Review commands

VPC - required Info

vpc-067893127034599d6 (Aditya\_webApp-vpc)

10.0.0.0/16

Subnet Info

subnet-054d91d5e47bd8edc Aditya\_webApp-subnet-public1-ap-south-1a

VPC: vpc-067893127034599d6 Owner: 590183667032

Availability Zone: ap-south-1a Zone type: Availability Zone

IP addresses available: 4088 CIDR: 10.0.0.0/20

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

Security group name - required

Aditya-bastionHost-sg

Description - required Info

launch-wizard-11 created 2024-09-21T10:33:04.419Z

Inbound Security Group Rules

Security group rule 1 (TCP, 22, 152.57.235.161/32)

Remove

Type Info

ssh

Protocol Info

TCP

Port range Info

22

Source type Info

My IP

Name Info

152.57.235.161/32

Description - optional Info

e.g. SSH for admin desktop

### Summary

Number of instances

Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.5.2...read more

ami-08718895af4dfa033

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

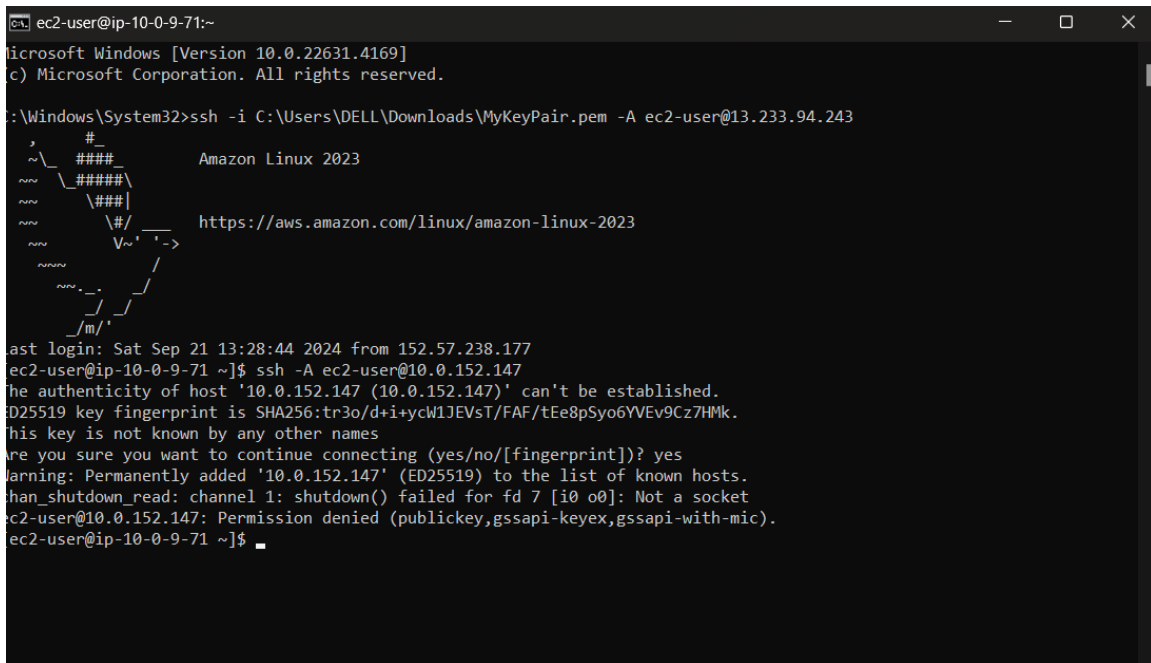
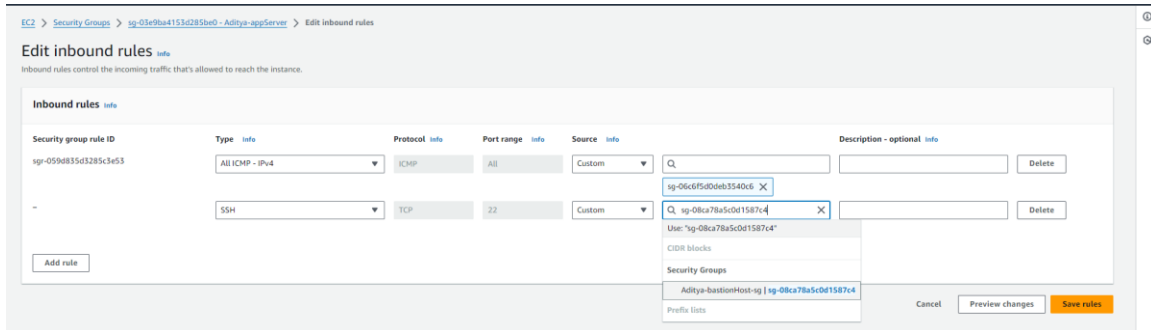
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Review commands



An Auto Scaling Group (ASG) that will dynamically provision EC2 instances.

An Application Load Balancer (ALB) to help route incoming traffic to the proper targets.

2. Set up a security group to allow SSH (port 22) and HTTP (port 80).

3. Install and start Apache Web Server on the instance:

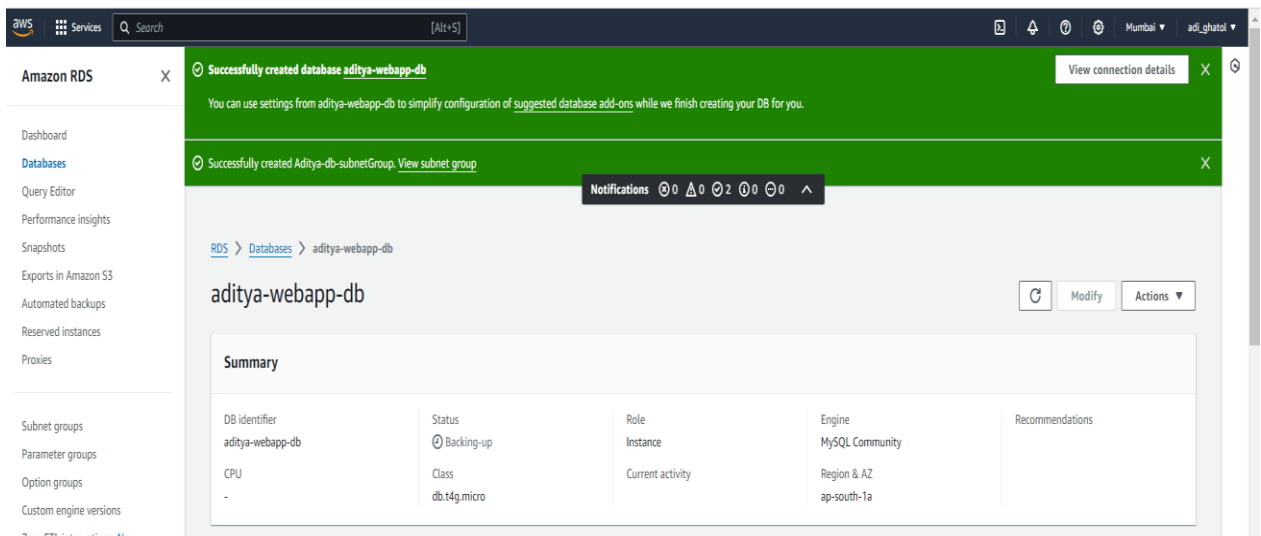
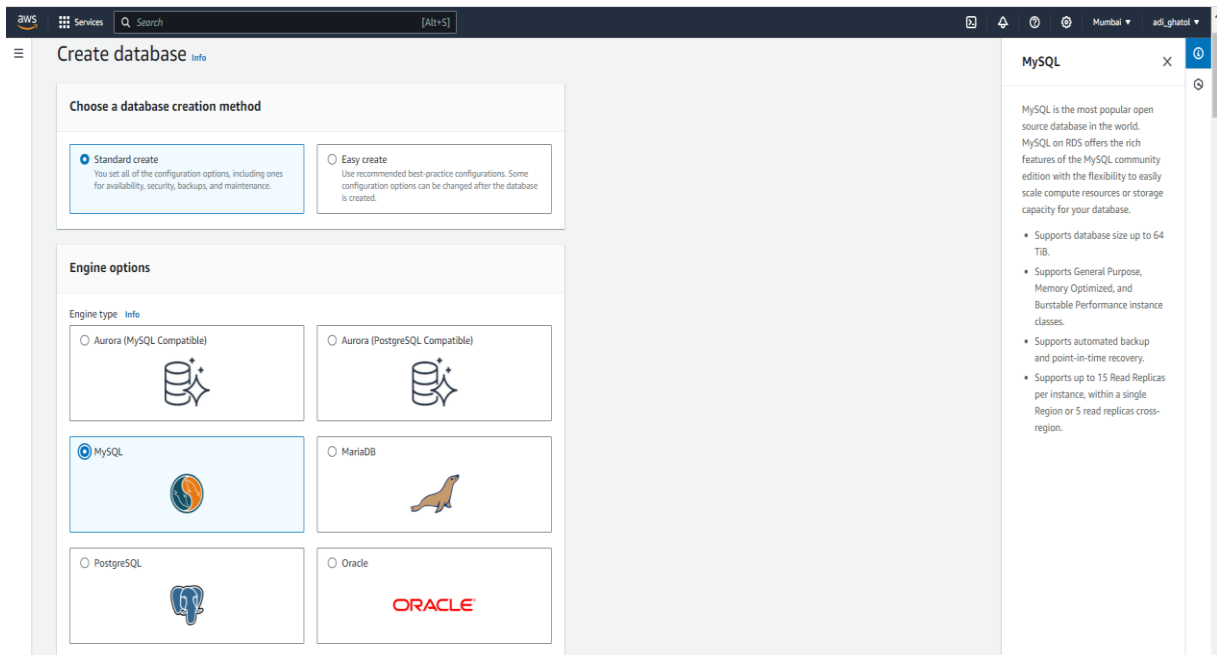
```
sudo yum update -y  
sudo yum install httpd -y  
sudo service httpd start
```

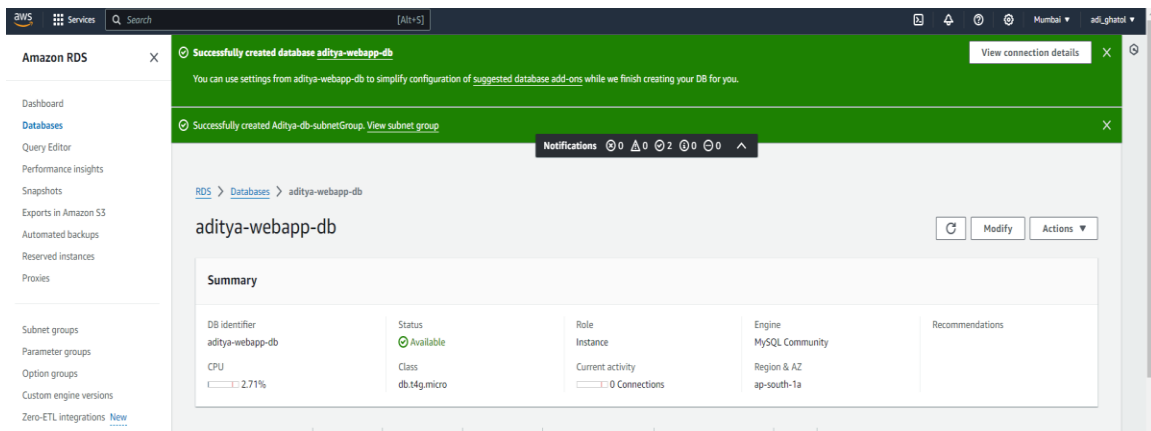
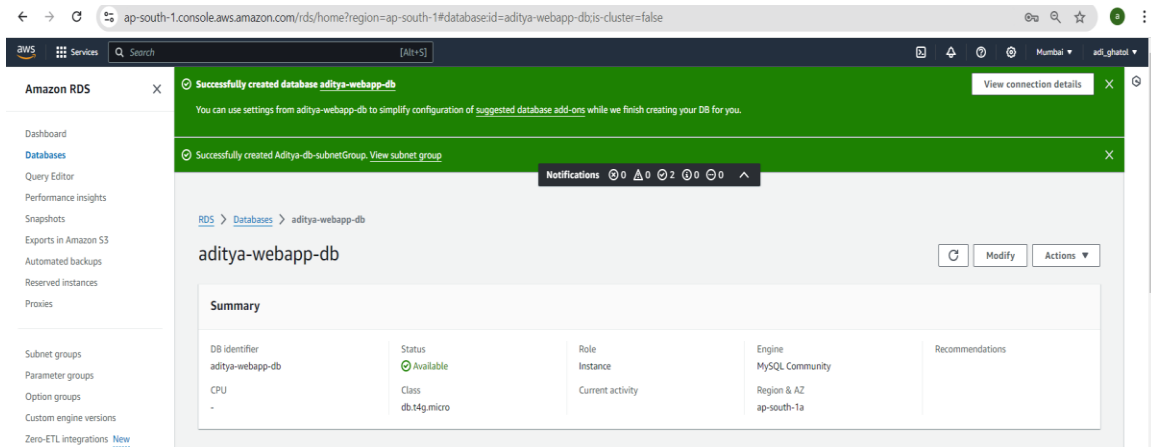
4. Access the instance via SSH using the public IP and verify the web server.

The screenshot shows the AWS Management Console interface for creating a DB subnet group. The breadcrumb navigation at the top indicates the path: **RDS** > **Subnet groups** > **Create DB subnet group**. The main heading is **Create DB subnet group**, followed by a descriptive paragraph: "To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC."

The form is divided into two main sections:

- Subnet group details**:
  - Name**: A text input field containing "Aditya-db-subnetGroup". A note states: "You won't be able to modify the name after your subnet group has been created." Below the field, it says: "Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed."
  - Description**: An empty text input field.
  - VPC**: A dropdown menu showing "Aditya\_webApp-vpc (vpc-067893127034599d6)". A note states: "Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created."
- Add subnets**:
  - Availability Zones**: A dropdown menu showing "Choose an availability zone". Below it, two tags are displayed: "ap-south-1a" and "ap-south-1b", each with a close button (X).
  - Subnets**: A dropdown menu showing "subnet-09410d957d3fedeb (10.0.160.0/20)". A note states: "Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones."





```
[ec2-user@ip-10-0-156-224 ~]$ mysql -h brainiac-webapp-db.cgwzuzkgdaxy.us-east-1
.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 115
Server version: 8.0.28 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> |
```



Great! We successfully connected to our database from our application server!

## **Success!**

Man, that was quite the journey! I know it wasn't easy, but we took it step-by-step and pulled through. We've successfully created a highly available, 3-tier application architecture that's ready for our Brainiac App!

## **7. Conclusion**

The AWS 3-tier web application project demonstrates the efficient use of cloud infrastructure for creating scalable, secure, and fault-tolerant applications. This architecture can be expanded further by incorporating additional AWS services such as S3, CloudFront, and Lambda for greater performance and flexibility.