AWS PROJECT

📌 **Project Documentation: Full-Stack Project Deployment on AWS**

🧩 **Project Description**

This project focuses on deploying a **Full-Stack Project** using **AWS cloud services** following industry-level deployment standards.
The application codebase is already developed — our primary objective is to **configure**, **integrate**, and **deploy** the full architecture on AWS.

🛠️ **Tech Stack Summary**

✅ **Frontend**

- **Next.js** – React framework for production-ready UI
- **Tailwind CSS** – utility-first styling
- **Material UI Data Grid** – advanced tabular data management
- **Redux Toolkit + RTK Query** – state and API data handling

✅ **Backend**

- **Node.js + Express.js** – REST API services
- **PostgreSQL** – relational database
- **Prisma ORM** – database mapping and migrations
- **PgAdmin** – database GUI for monitoring

🎯 **Project Goal**

To deploy a **secure, scalable, and maintainable** web application on AWS, using serverless principles and managed authentication.
The deployment process includes detailed, step-by-step guidance with screenshots to avoid misconfigurations.

🧹 **Setup & Installation:** Follow these steps before deploying to AWS:

✅ 1️⃣ **Clone & Initialize the Repository(https://github.com/ITS-NAYAN/project-management.git)**

Download .zip — If you received a ZIP folder

1. Create a folder named **project-management**
2. Extract ZIP contents into the folder
3. Open the folder in **VS Code**

Then initialize Git:

git init

git add .

git commit -m "Initial commit"

Now, create a new GitHub repository:
➡️ https://github.com/new

After creating the repo, connect and push code:

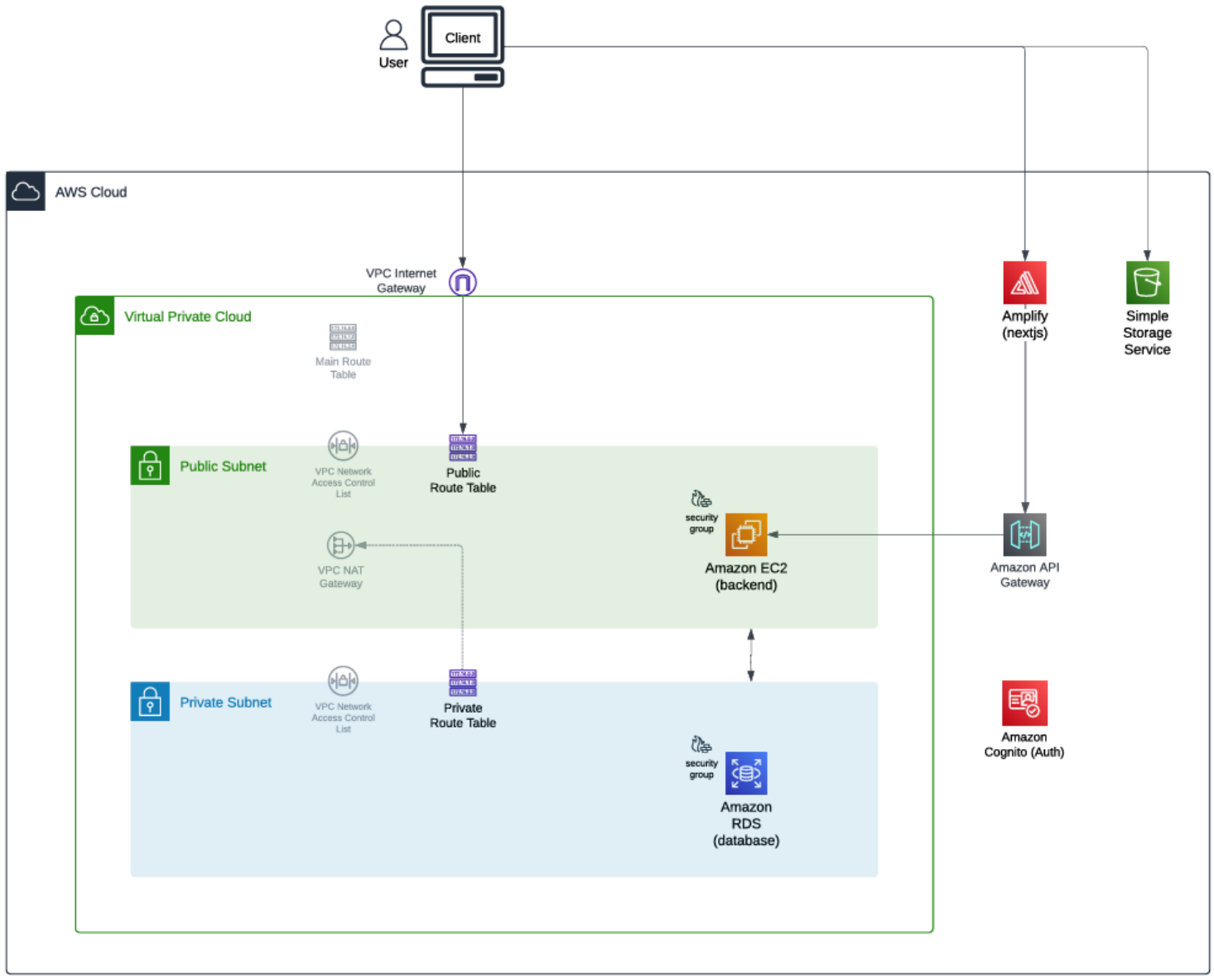
git remote add origin https://github.com/your-username/project-management.git

git branch -M main

git push -u origin main

✅ **Repository successfully stored on GitHub**

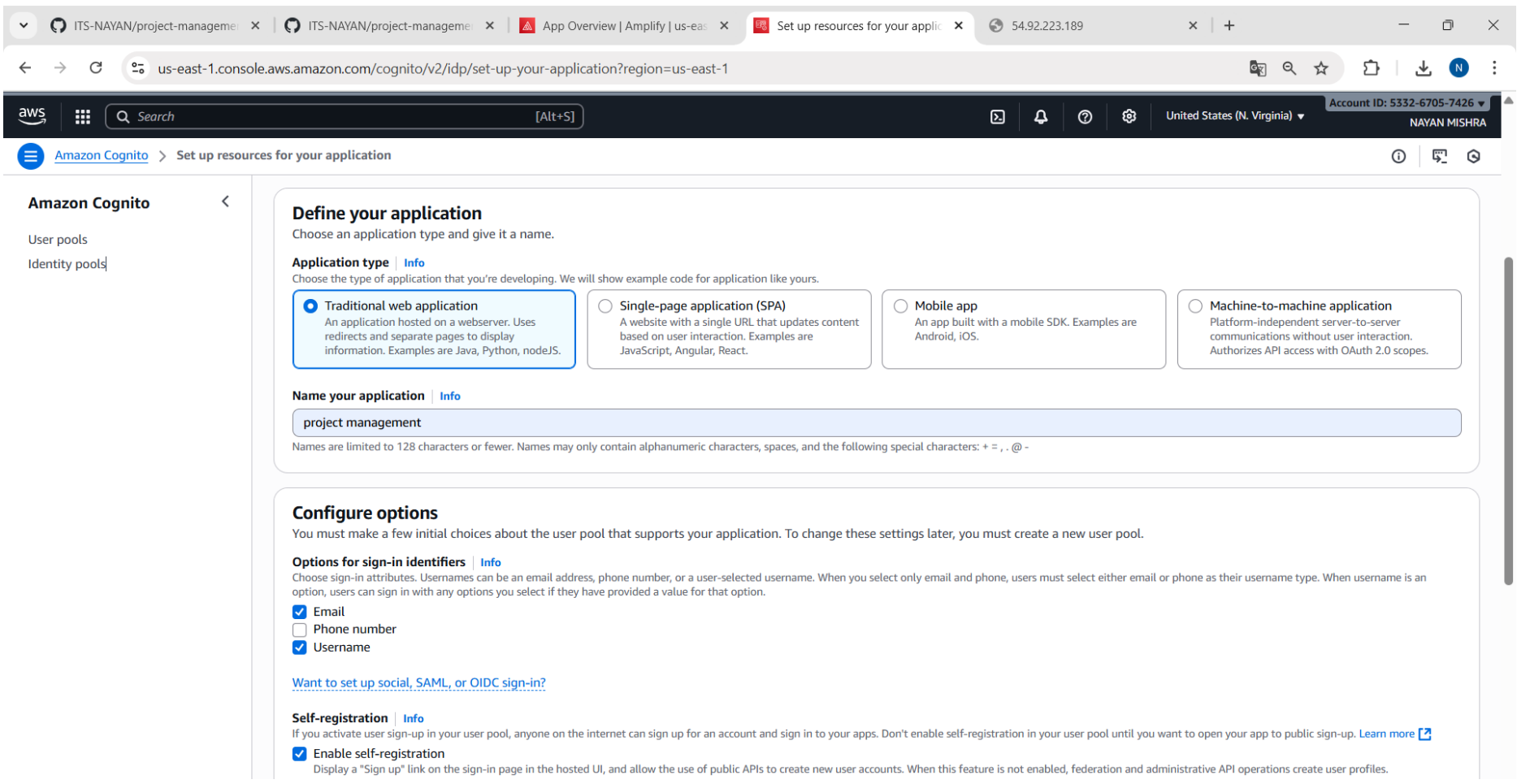
AWS DEPLOYMENT ARCHITECTURE

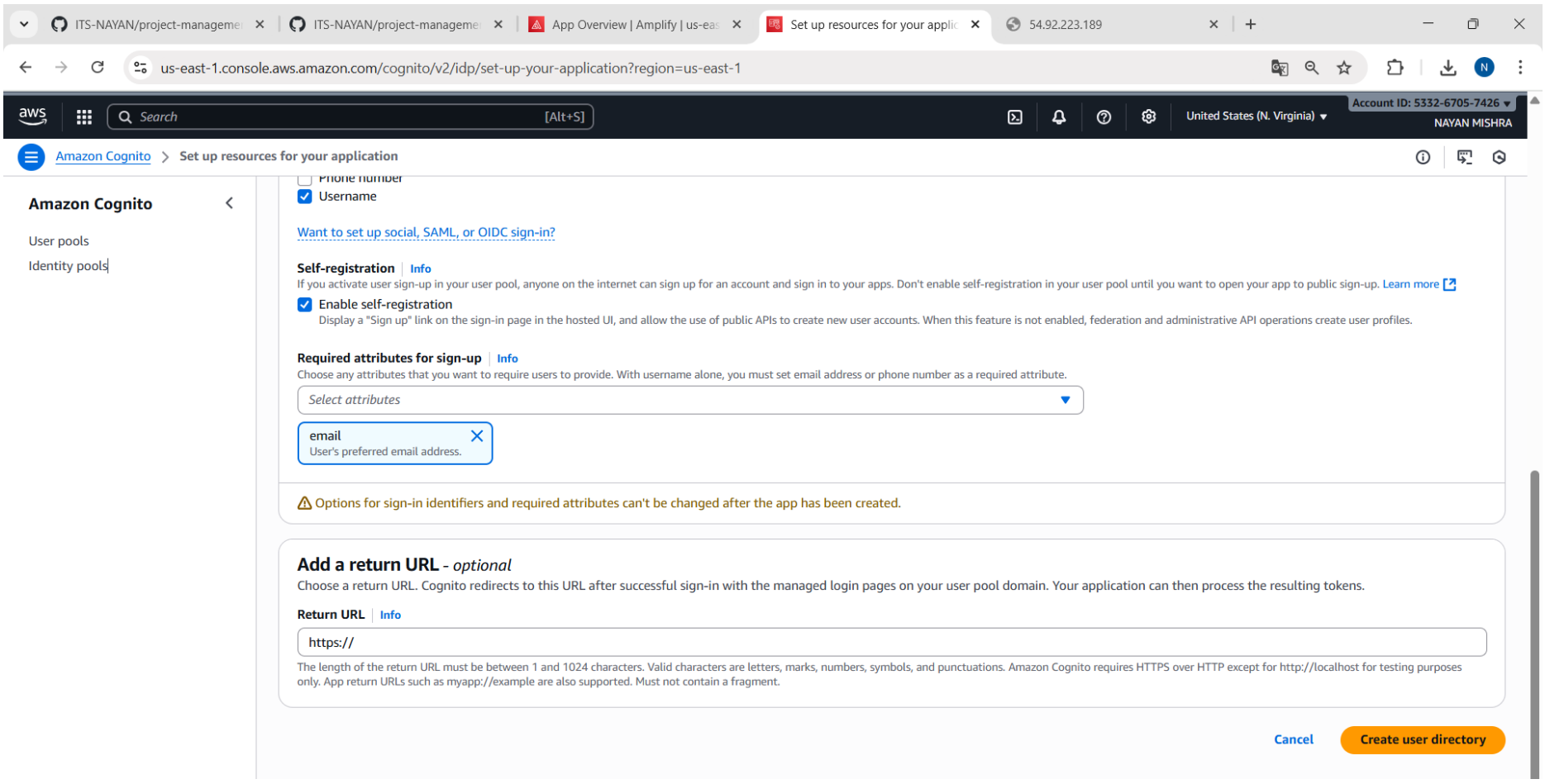


🔐 Step 2 — Configure AWS Cognito Authentication

✅ Create User Pool

Use AWS Console → **Cognito** → Create **User Pool**

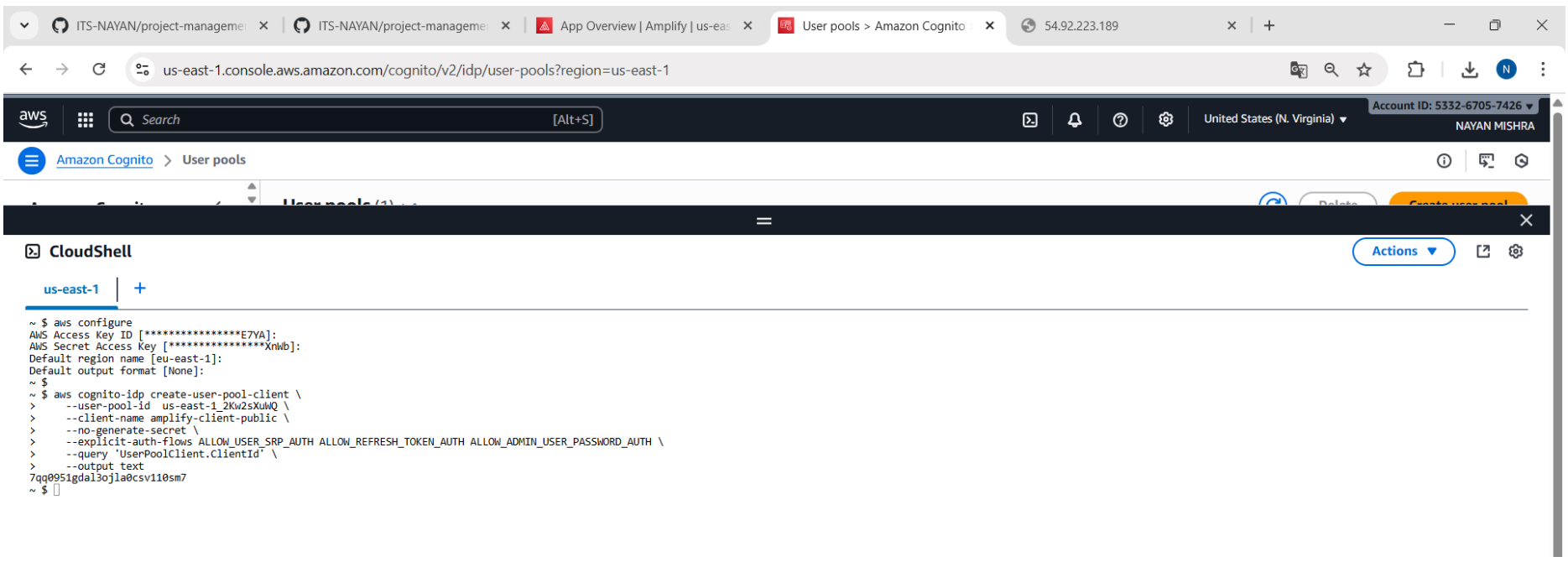




✓ Note down the **User Pool ID**
Example: us-east-1_2Kw2sXuWQ

✅ Create User Pool Client (via CloudShell)

Run the following command in AWS CloudShell:




```
aws cognito-idp create-user-pool-client \
  --user-pool-id us-east-1_2Kw2sXuWQ \
  --client-name amplify-client-public \
  --no-generate-secret \
  --explicit-auth-flows ALLOW_USER_SRP_AUTH ALLOW_REFRESH_TOKEN_AUTH ALLOW_ADMIN_USER_PASSWORD_AUTH \
  --query 'UserPoolClient.ClientId' \
  --output text
```

Example output:

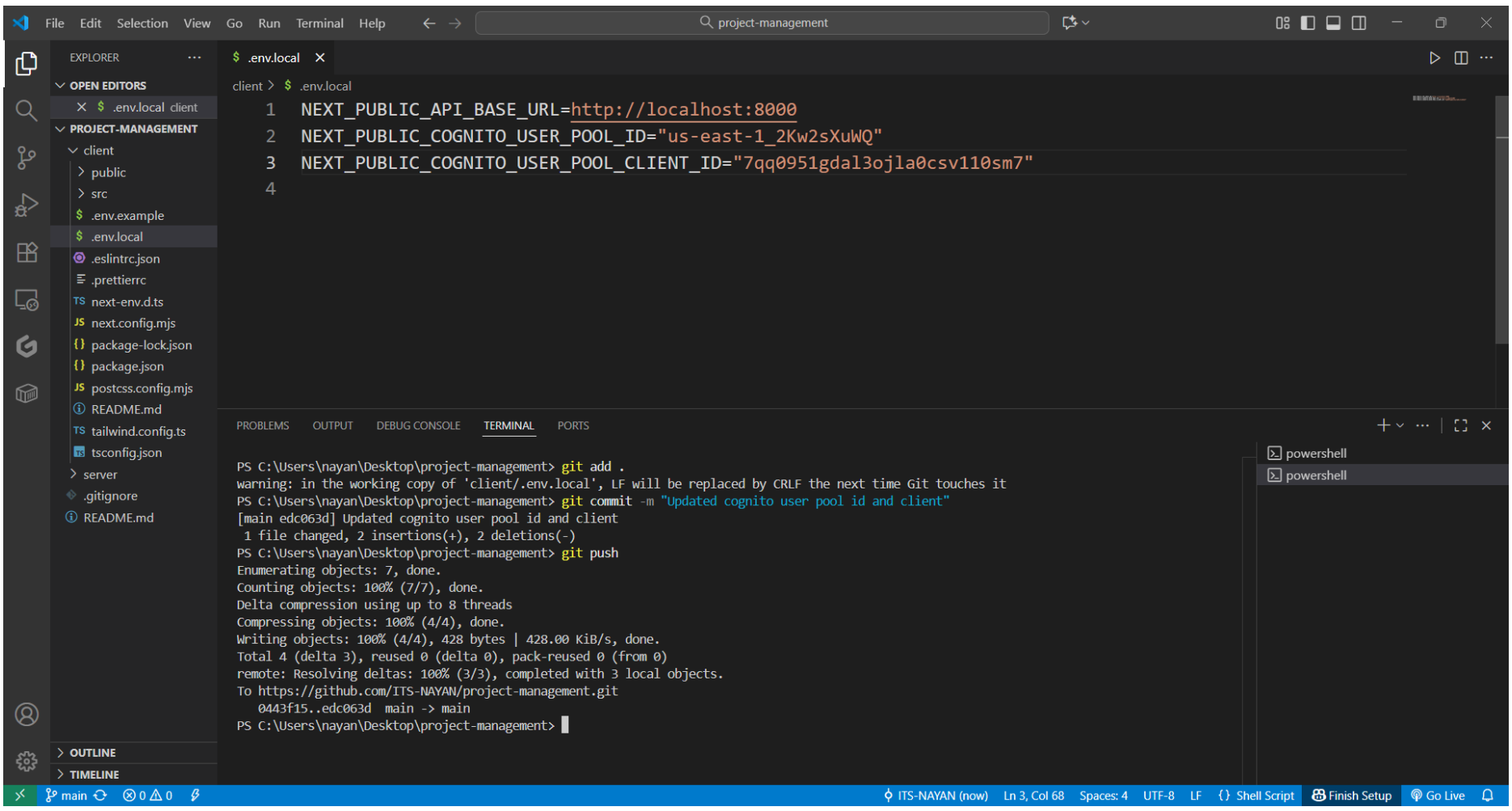
7qq0951gdal3ojla0csv110sm7

📌 Important IDs to Store

ID Type	Value
Cognito User Pool ID	us-east-1_2Kw2sXuWQ
Cognito Client ID	7qq0951gdal3ojla0csv110sm7

💡 Keep these safe — required for authentication integration.

✅ Update Environment Variables in Frontend



Go to:

📁 client/src/ → Create / Edit → .env.local

Add:

NEXT_PUBLIC_COGNITO_USER_POOL_ID=us-east-1_2Kw2sXuWQ

NEXT_PUBLIC_COGNITO_CLIENT_ID=7qq0951gdal3ojla0csv110sm7

✓ Also push changes in git using commands shown in screenshot. This connects your frontend to AWS Cognito authentication

---

🪜 PHASE 1 — CORE NETWORKING INFRASTRUCTURE

Components: VPC | Subnets | Internet Gateway | Route Tables
Objective:
To build a secure and isolated network environment in AWS where:

- Public-facing resources get controlled internet access
- Private resources remain fully protected and isolated
- High availability and scalability are supported

✅ 1️⃣ VPC Creation — PM-VPC

Setting	Value
Name	PM-VPC
CIDR Block	10.0.0.0/16

Purpose & Benefits:

- Creates a **logically isolated** network from the AWS public cloud
- /16 CIDR provides **65,536 private IPs**, allowing future scaling
- Forms the **foundation** for all networking in this project

---

✅ 2️⃣ Subnet Design & Configuration

To properly segment resources and secure sensitive workloads:

Subnet Name	CIDR	Type	Availability Zone	Purpose
PM-public-subnet-1	10.0.0.0/24	Public	us-east-2a	EC2 / API resources requiring internet access
PM-private-subnet-1	10.0.1.0/24	Private	us-east-2a	Backend + Database (sensitive components)
PM-private-subnet-2	10.0.2.0/24	Private	us-east-2b	High availability + RDS Multi-AZ support

Why two private subnets?

- Ensures **fault tolerance**
- Required for **RDS Multi-AZ** deployments
- Supports future auto-scaling in multiple zones

✅ 3️⃣ Internet Gateway — PM-Internet-Gateway

Element	Description
Internet Gateway	Created and attached to **PM-VPC**

Purpose:

- Provides **internet connectivity** for resources in public subnets
- Enables inbound/outbound web traffic routing

📌 *Private subnets remain isolated — no IGW route*

✅ 🔽 Route Tables — Public & Private Segregation

Route Table Name	Associated Subnets	Internet Route	Purpose
PM-public-route-table-1	PM-public-subnet-1	0.0.0.0/0 → Internet Gateway	Allows public resources to access the internet
PM-private-route-table-1	PM-private-subnet-1	❌ No internet route	Full isolation for sensitive resources
PM-private-route-table-2	PM-private-subnet-2	❌ No internet route	Supports HA database deployment

Routing Benefits:

- Public subnets handle internet-facing workloads securely
- Private subnets restrict exposure and reduce attack surface

---

🚀 Phase 2 — Backend Deployment

Components: EC2 | Security Groups | Node.js Backend
Objective: Deploy the backend application on an EC2 instance and connect it securely with a PostgreSQL database hosted on AWS.

✅ 1️⃣ EC2 Instance Deployment — pm-ec2-backend

Configuration	Details
Instance Type	t2.micro (Free Tier Eligible)
AMI	Amazon Linux 2
Network Placement	PM-VPC → PM-public-subnet-1
Public IP	Enabled (for internet access)
Key Pair	Created for secured SSH login

🛡️ Security Group — pm-ec2-sg

Port	Protocol	Purpose
22	SSH	Remote admin access
80	HTTP	Public access to backend APIs
443	HTTPS	Secure traffic support

i Only required and safe communication ports are allowed → **enhanced security**

🛠️ Software & Backend Setup (SSH Deployment Steps)

Once the EC2 instance is live:
✅ Connect using SSH and configure: And run all commands in **aws-ec2-instructions.md**

✅ THIS IS FROM EC2 INSTRUCTION.MD Create a pm2 ecosystem configuration file (inside server directory):

```
[root@ip-10-0-0-176 server]#
[root@ip-10-0-0-176 server]# ls
aws-ec2-instructions.md  dist  ecosystem.config.js  node_modules  package-lock.json  package.json  prisma  src  tsconfig.json
[root@ip-10-0-0-176 server]# nano ecosystem.config.js
[root@ip-10-0-0-176 server]#
```

```
root@ip-10-0-0-176:~/project
 GNU nano 8.3                                    ecosystem.config.js
module.exports = {
  apps: [
    {
      name: "project-management",
      script: "npm",
      args: "run dev",
      env: {
        NODE_ENV: "development",
      },
    },
  ],
};
```

🗄️ RDS DATABASE SETUP — PM-RDS

Objective: Deploy a secured, private PostgreSQL database accessible **only** by the backend EC2 instance.

✅ Database Configuration

Setting	Value
Engine	PostgreSQL
DB Identifier	pm-rds
Instance Class	Free-Tier Eligible
Public Access	❌ Disabled
Storage Autoscaling	❌ Disabled
Backups	Disabled (cost optimization)
Encryption	Disabled (free-tier compliance)

🔐 Master Username & Password
➞ Stored securely and **not exposed publicly**

🌐 Network Placement

Component	Configuration
VPC	PM-VPC
Subnet Group	Private — includes PM-private-subnet-1 & PM-private-subnet-2
High Availability	✅ Multi-AZ enabled via subnet group placement

Benefits:

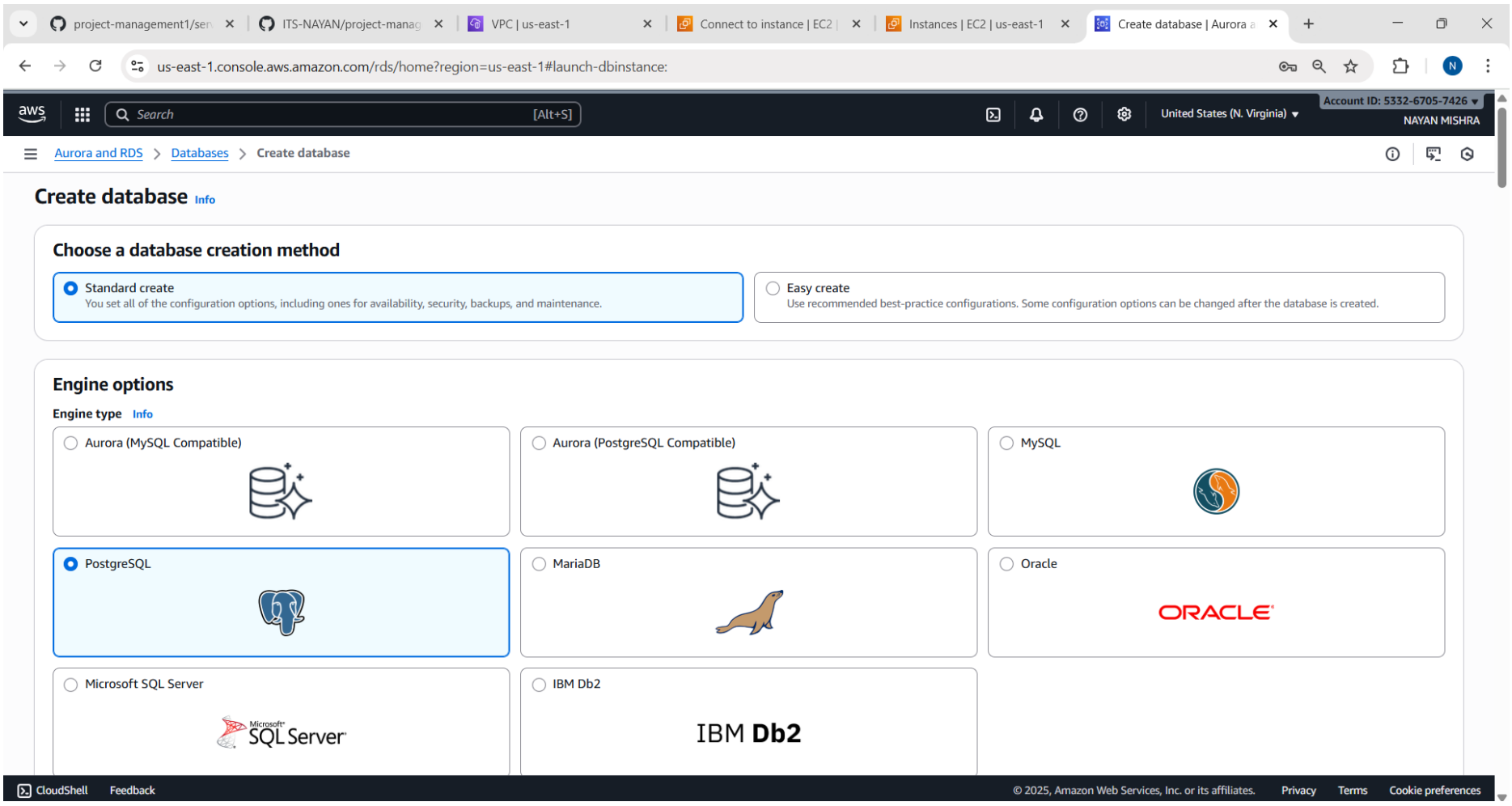
- Private subnets → **No internet exposure**
- High availability → **AZ failure resilience**
- Backend-only access → **Zero unwanted traffic**

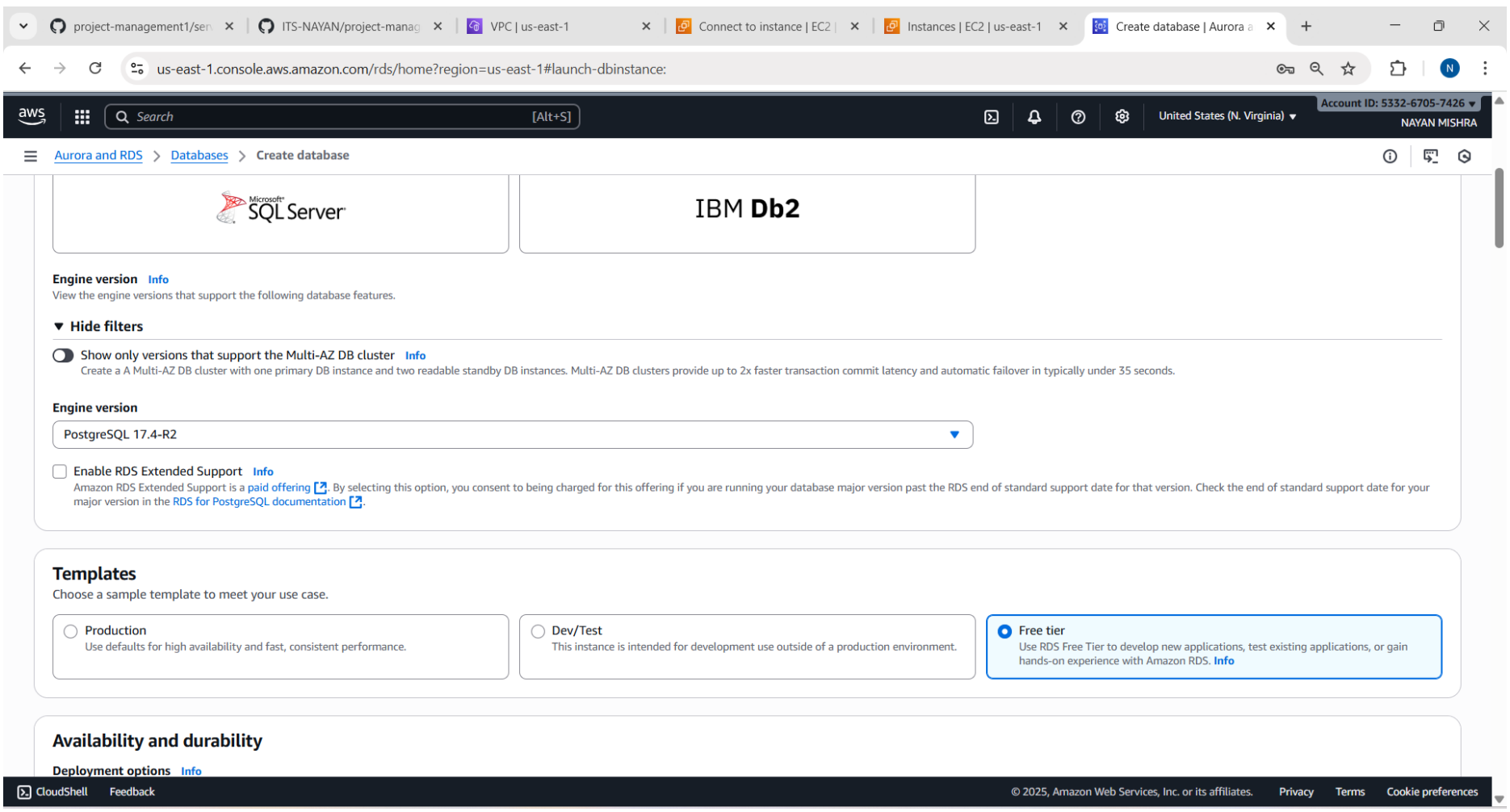
🛡️ Security Group — PM-RDS-SG(SCREENSHOT AT END)

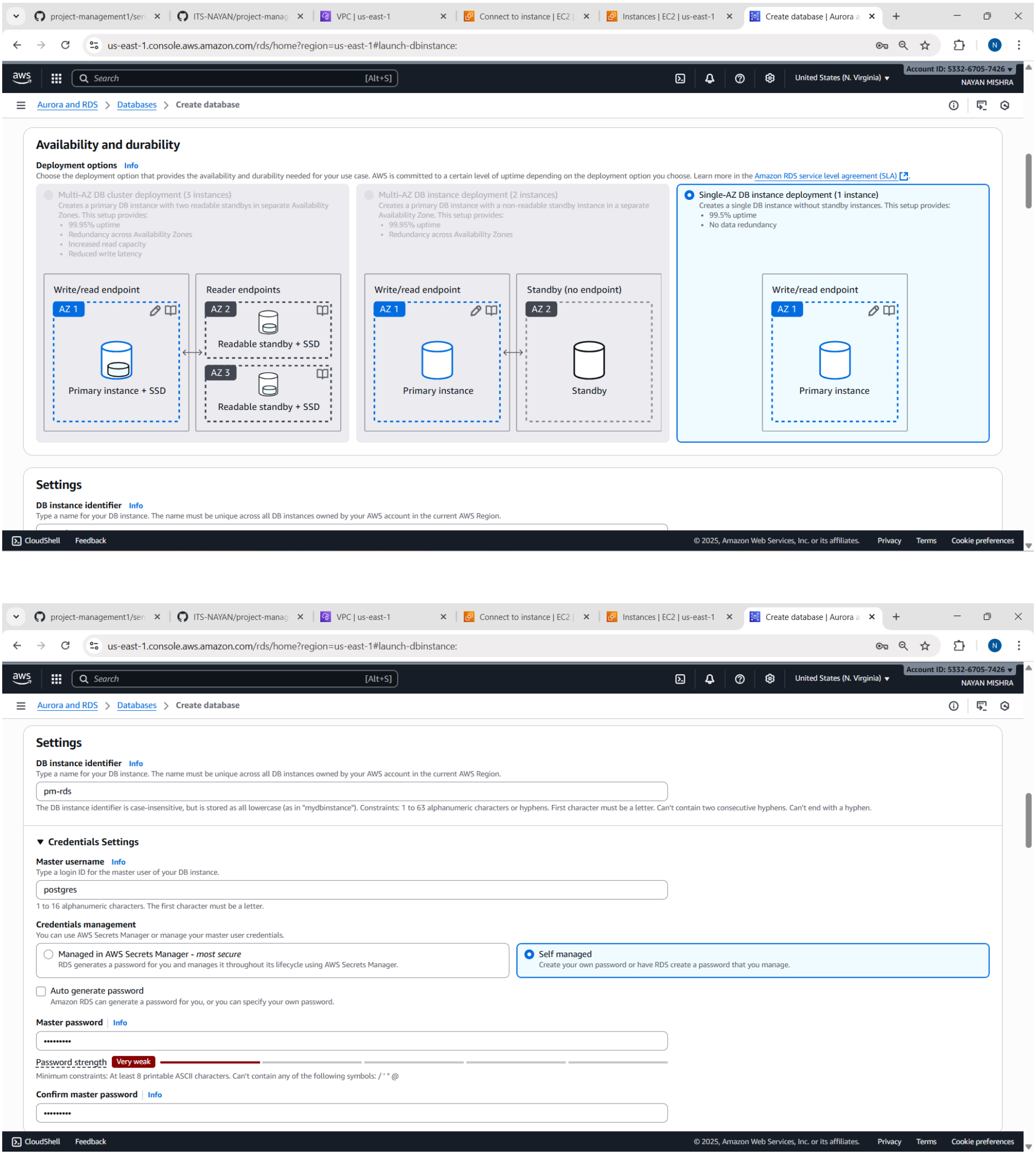
Rule Type	Source	Port	Purpose
Inbound	pm-ec2-sg	5432	Allow database access **only** from backend
Outbound	pm-ec2-sg allowed	5432	Ensures return traffic to backend

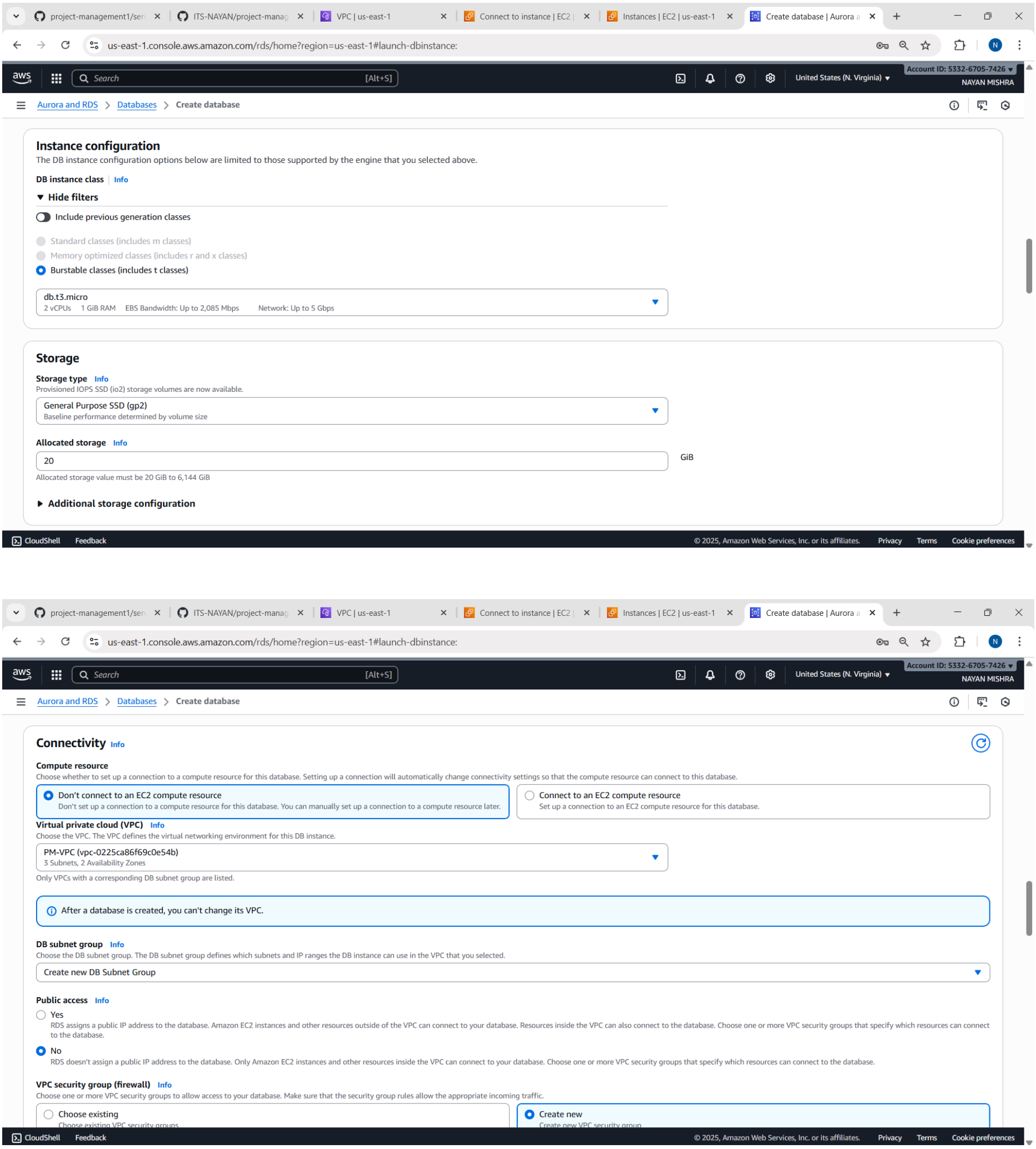
✅ Strong security posture:
Database invisible to the outside world and other AWS services

---

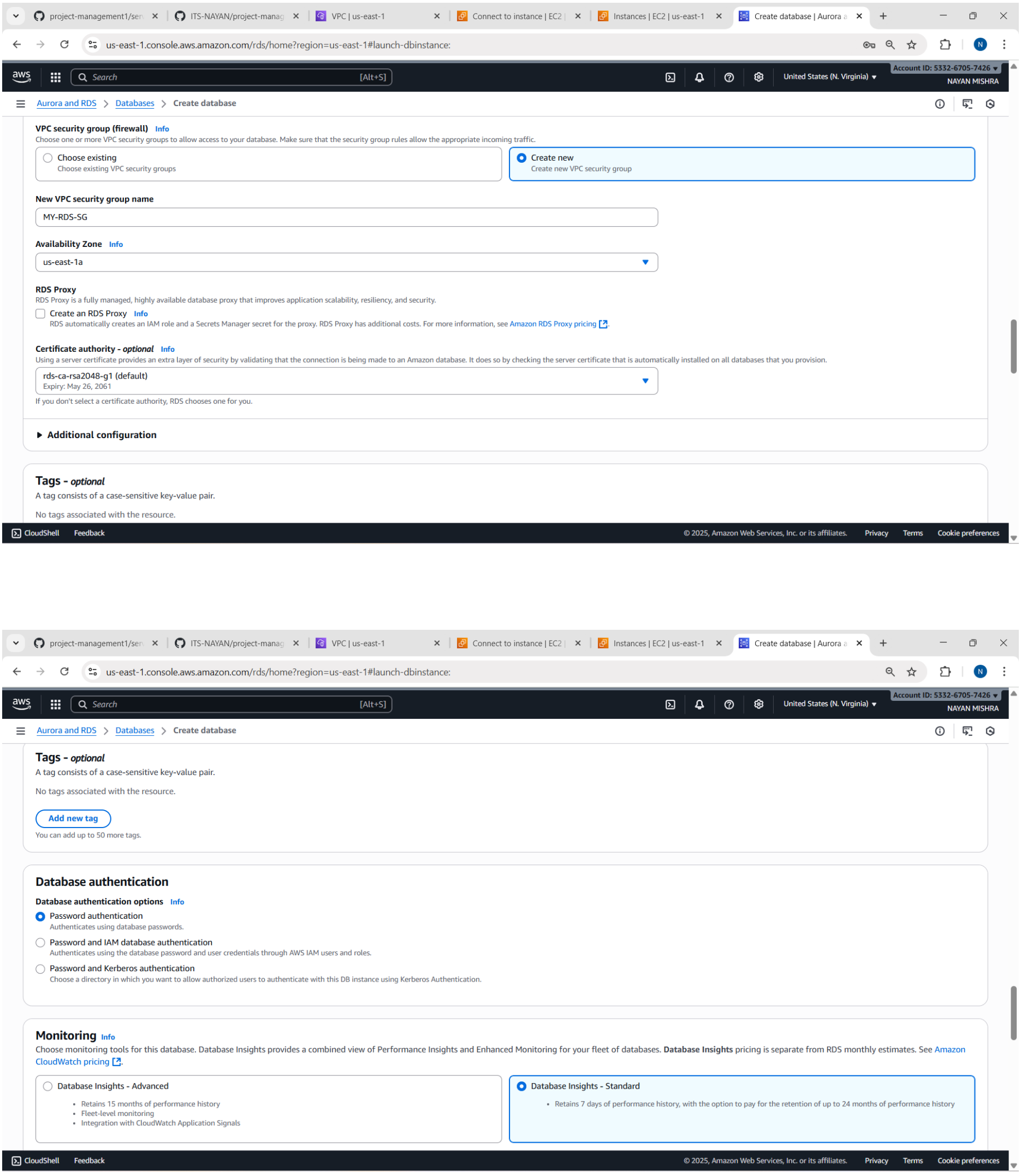

project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Connect to instance | EC2
Instances | EC2 | us-east-1
Create database | Aurora a
us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS
Databases
Create database
Create database Info
Choose a database creation method
Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.
Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.
Engine options
Engine type Info
Aurora (MySQL Compatible)
Aurora (PostgreSQL Compatible)
MySQL
PostgreSQL
MariaDB
Oracle
ORACLE
Microsoft SQL Server
IBM Db2
IBM Db2
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Connect to instance | EC2
Instances | EC2 | us-east-1
Create database | Aurora a
us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS
Databases
Create database
Microsoft SQL Server
IBM Db2
Engine version Info
View the engine versions that support the following database features.
Hide filters
Show only versions that support the Multi-AZ DB cluster Info
Create a A Multi-AZ DB cluster with one primary DB instance and two readable standby DB instances. Multi-AZ DB clusters provide up to 2x faster transaction commit latency and automatic failover in typically under 35 seconds.
Engine version
PostgreSQL 17.4-R2
Enable RDS Extended Support Info
Amazon RDS Extended Support is a paid offering. By selecting this option, you consent to being charged for this offering if you are running your database major version past the RDS end of standard support date for that version. Check the end of standard support date for your major version in the RDS for PostgreSQL documentation.
Templates
Choose a sample template to meet your use case.
Production
Use defaults for high availability and fast, consistent performance.
Dev/Test
This instance is intended for development use outside of a production environment.
Free tier
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. Info
Availability and durability
Deployment options Info
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

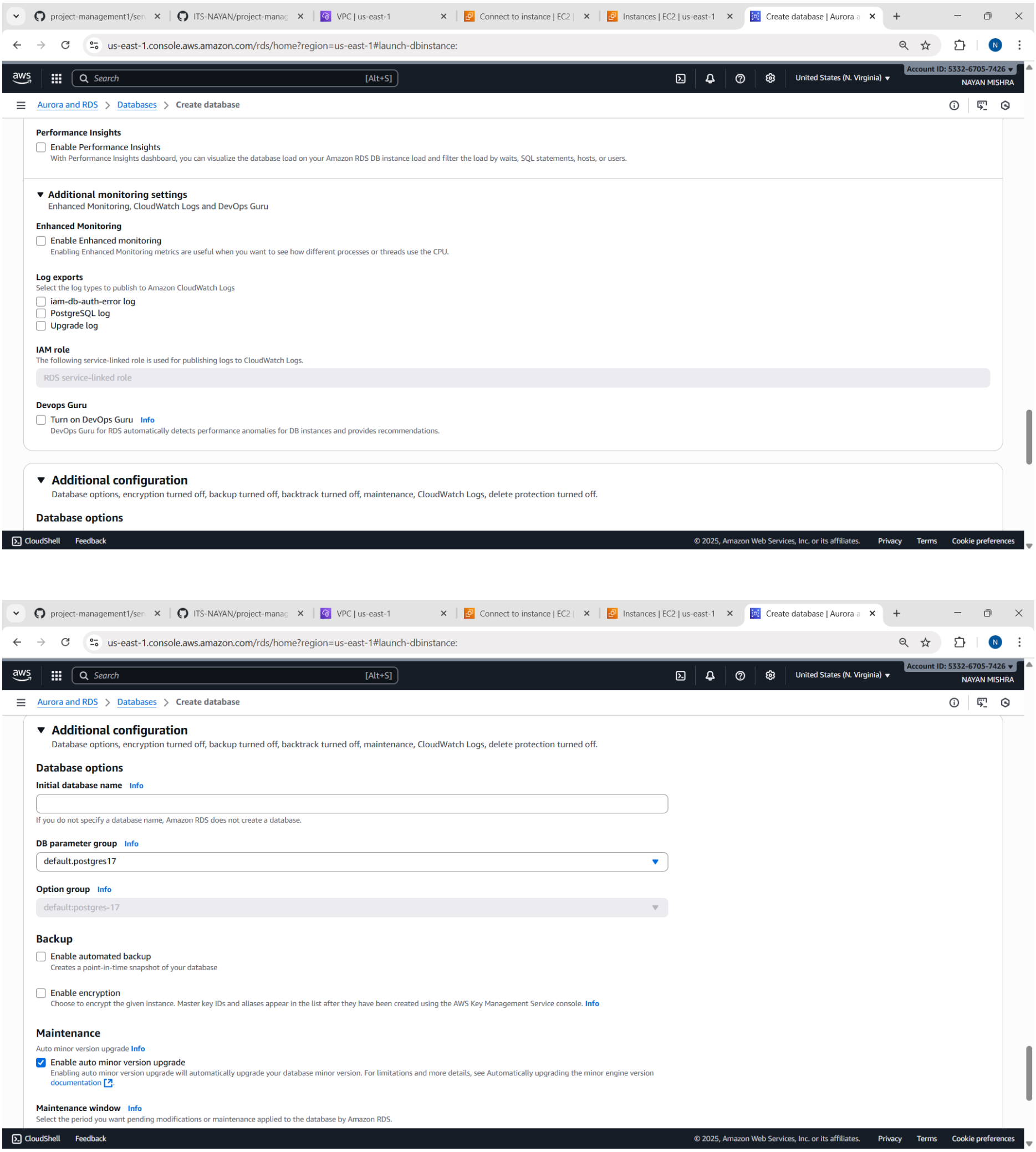

project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Connect to instance | EC2
Instances | EC2 | us-east-1
Create database | Aurora a
us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS > Databases > Create database
Availability and durability
Deployment options Info
Choose the deployment option that provides the availability and durability needed for your use case. AWS is committed to a certain level of uptime depending on the deployment option you choose. Learn more in the Amazon RDS service level agreement (SLA).
Multi-AZ DB cluster deployment (3 instances)
Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:
99.95% uptime
Redundancy across Availability Zones
Increased read capacity
Reduced write latency
Write/read endpoint
AZ 1
Primary instance + SSD
Reader endpoints
AZ 2
Readable standby + SSD
AZ 3
Readable standby + SSD
Multi-AZ DB instance deployment (2 instances)
Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:
99.95% uptime
Redundancy across Availability Zones
Write/read endpoint
AZ 1
Primary instance
Standby (no endpoint)
AZ 2
Standby
Single-AZ DB instance deployment (1 instance)
Creates a single DB instance without standby instances. This setup provides:
99.5% uptime
No data redundancy
Write/read endpoint
AZ 1
Primary instance
Settings
DB instance identifier Info
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
pm-rds
The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.
Credentials Settings
Master username Info
Type a login ID for the master user of your DB instance.
postgres
1 to 16 alphanumeric characters. The first character must be a letter.
Credentials management
You can use AWS Secrets Manager or manage your master user credentials.
Managed in AWS Secrets Manager - most secure
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.
Self managed
Create your own password or have RDS create a password that you manage.
Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.
Master password Info
Password strength
Very weak
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @
Confirm master password Info
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

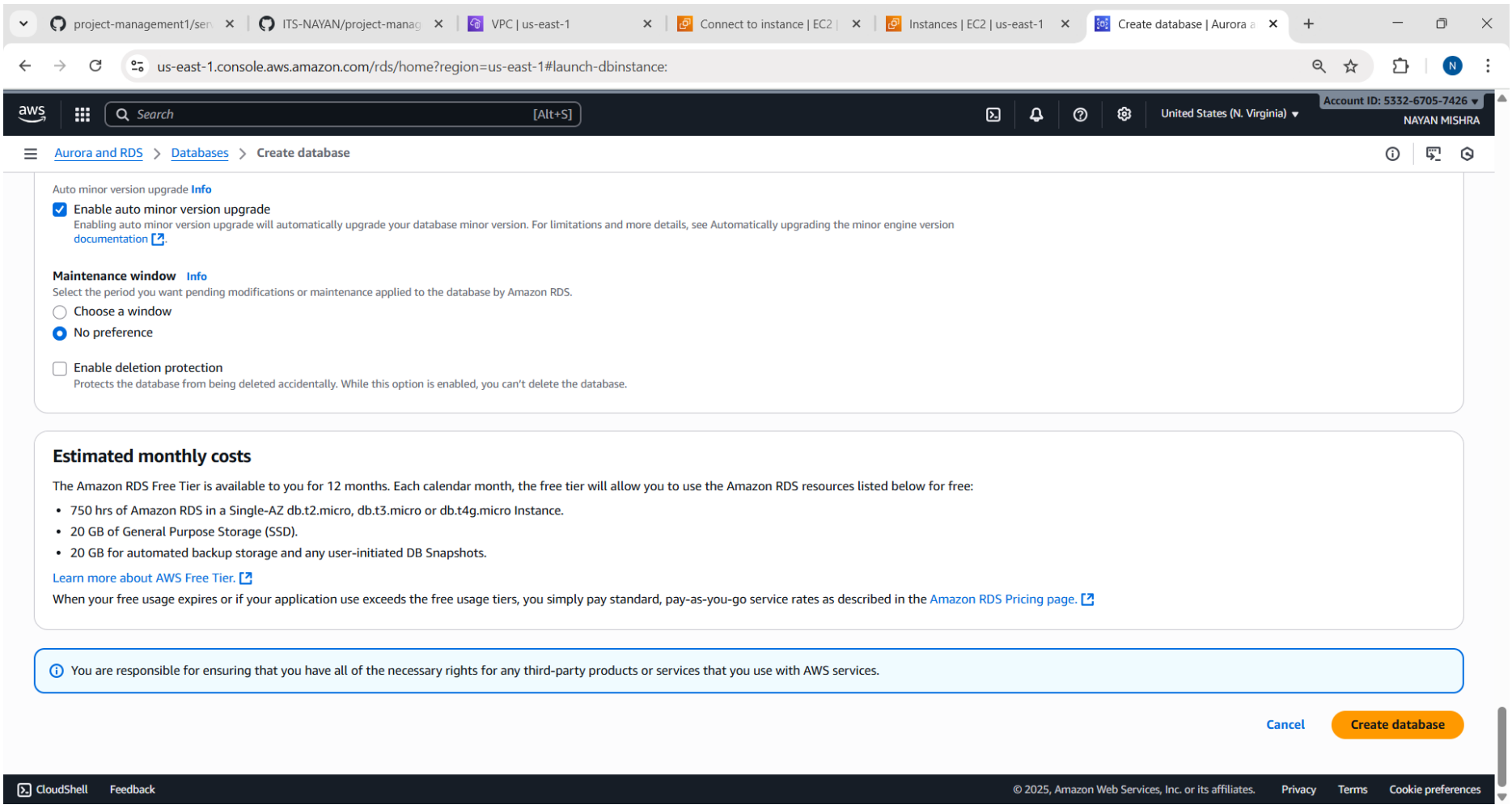

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS > Databases > Create database
Instance configuration
The DB instance configuration options below are limited to those supported by the engine that you selected above.
DB instance class Info
Hide filters
Include previous generation classes
Standard classes (includes m classes)
Memory optimized classes (includes r and x classes)
Burstable classes (includes t classes)
db.t3.micro
2 vCPUs 1 GiB RAM EBS Bandwidth: Up to 2,085 Mbps Network: Up to 5 Gbps
Storage
Storage type Info
Provisioned IOPS SSD (io2) storage volumes are now available.
General Purpose SSD (gp2)
Baseline performance determined by volume size
Allocated storage Info
20
GiB
Allocated storage value must be 20 GiB to 6,144 GiB
Additional storage configuration
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS > Databases > Create database
Connectivity Info
Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.
Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.
Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.
Virtual private cloud (VPC) Info
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.
PM-VPC (vpc-0225ca86f69c0e54b)
3 Subnets, 2 Availability Zones
Only VPCs with a corresponding DB subnet group are listed.
After a database is created, you can't change its VPC.
DB subnet group Info
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.
Create new DB Subnet Group
Public access Info
Yes
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.
No
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.
VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.
Choose existing
Choose existing VPC security groups
Create new
Create new VPC security group
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS > Databases > Create database
VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.
Choose existing
Choose existing VPC security groups
Create new
Create new VPC security group
New VPC security group name
MY-RDS-SG
Availability Zone Info
us-east-1a
RDS Proxy
RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.
Create an RDS Proxy Info
RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see Amazon RDS Proxy pricing.
Certificate authority - optional Info
Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.
rds-ca-rsa2048-g1 (default)
Expiry: May 26, 2061
If you don't select a certificate authority, RDS chooses one for you.
Additional configuration
Tags - optional
A tag consists of a case-sensitive key-value pair.
No tags associated with the resource.
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS > Databases > Create database
Tags - optional
A tag consists of a case-sensitive key-value pair.
No tags associated with the resource.
Add new tag
You can add up to 50 more tags.
Database authentication
Database authentication options Info
Password authentication
Authenticates using database passwords.
Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.
Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.
Monitoring Info
Choose monitoring tools for this database. Database Insights provides a combined view of Performance Insights and Enhanced Monitoring for your fleet of databases. Database Insights pricing is separate from RDS monthly estimates. See Amazon CloudWatch pricing.
Database Insights - Advanced
Retains 15 months of performance history
Fleet-level monitoring
Integration with CloudWatch Application Signals
Database Insights - Standard
Retains 7 days of performance history, with the option to pay for the retention of up to 24 months of performance history
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Connect to instance | EC2
Instances | EC2 | us-east-1
Create database | Aurora a
us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Aurora and RDS
Databases
Create database
Performance Insights
Enable Performance Insights
With Performance Insights dashboard, you can visualize the database load on your Amazon RDS DB instance load and filter the load by waits, SQL statements, hosts, or users.
Additional monitoring settings
Enhanced Monitoring, CloudWatch Logs and DevOps Guru
Enhanced Monitoring
Enable Enhanced monitoring
Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.
Log exports
Select the log types to publish to Amazon CloudWatch Logs
iam-db-auth-error log
PostgreSQL log
Upgrade log
IAM role
The following service-linked role is used for publishing logs to CloudWatch Logs.
RDS service-linked role
Devops Guru
Turn on DevOps Guru Info
DevOps Guru for RDS automatically detects performance anomalies for DB instances and provides recommendations.
Additional configuration
Database options, encryption turned off, backup turned off, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned off.
Database options
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Initial database name Info
If you do not specify a database name, Amazon RDS does not create a database.
DB parameter group Info
default.postgres17
Option group Info
default:postgres-17
Backup
Enable automated backup
Creates a point-in-time snapshot of your database
Enable encryption
Choose to encrypt the given instance. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service console. Info
Maintenance
Auto minor version upgrade Info
Enable auto minor version upgrade
Enabling auto minor version upgrade will automatically upgrade your database minor version. For limitations and more details, see Automatically upgrading the minor engine version documentation.
Maintenance window Info
Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.



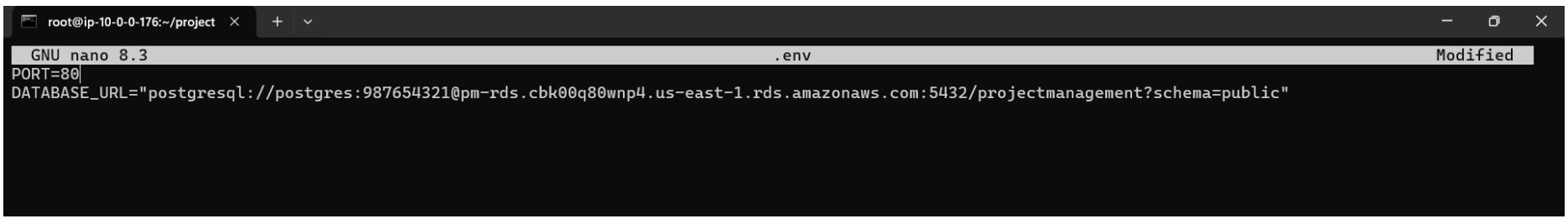
🔧 Backend Integration (EC2)

After RDS creation:

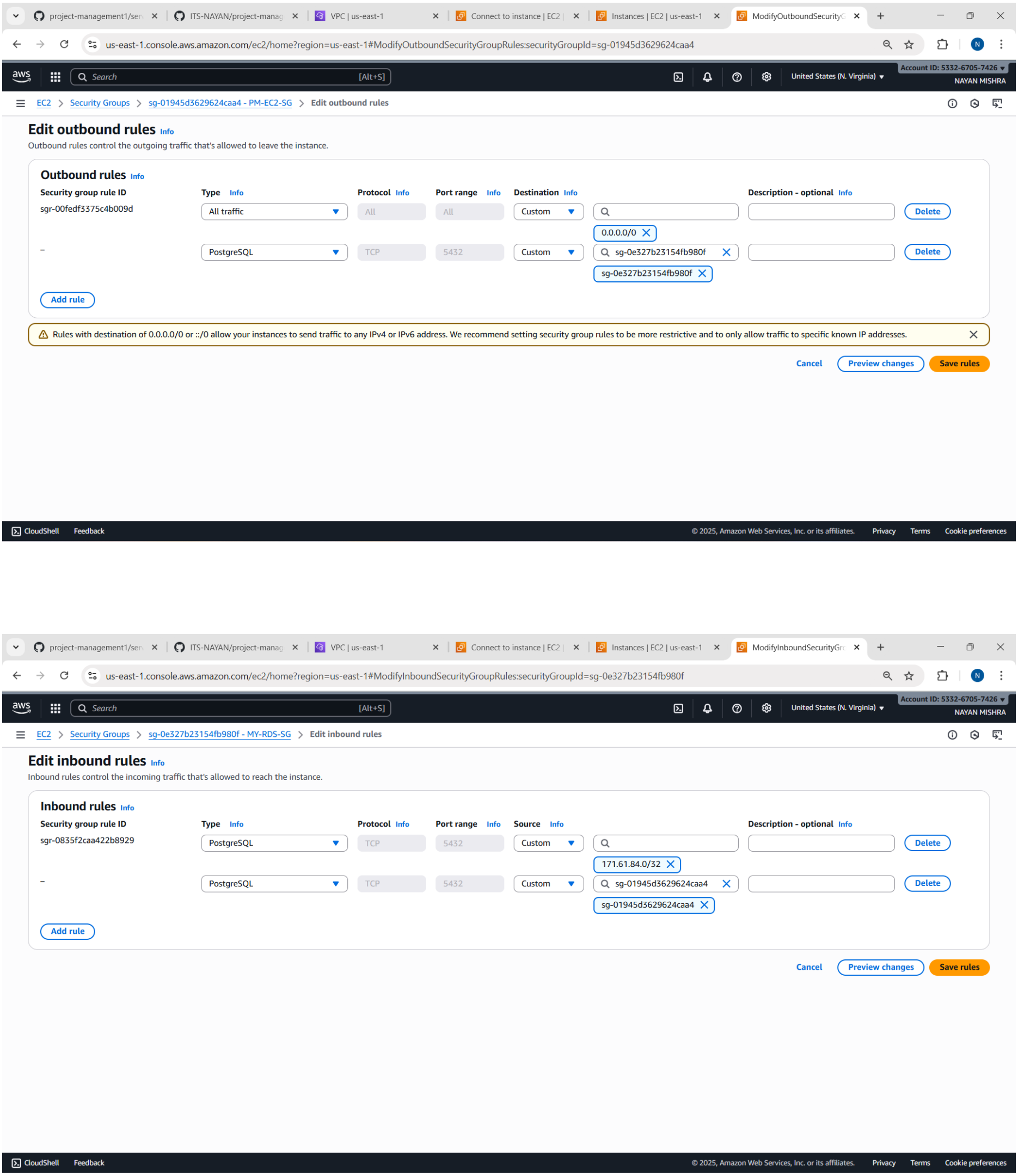
✅ 1️⃣ Update .env configuration on EC2

Added RDS connection parameters including: you add your own

- RDS Endpoint: pm-rds.cbk00q80wnp4.us-east-1.rds.amazonaws.com
- Database Name
- Master Username
- Master Password: 987654321



Paste this URL according to configuration: DATABASE_URL="postgresql://postgres:987654321@pm-rds.cbk00q80wnp4.us-east-1.rds.amazonaws.com:5432/projectmanagement?schema=public"


project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Connect to instance | EC2
Instances | EC2 | us-east-1
ModifyOutboundSecurityG
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ModifyOutboundSecurityGroupRules:securityGroupId=sg-01945d3629624caa4
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
EC2 > Security Groups > sg-01945d3629624caa4 - PM-EC2-SG > Edit outbound rules
Edit outbound rules Info
Outbound rules control the outgoing traffic that's allowed to leave the instance.
Outbound rules Info
Security group rule ID
Type Info
Protocol Info
Port range Info
Destination Info
Description - optional Info
sgr-00fedf3375c4b009d
All traffic
All
All
Custom
0.0.0.0/0
Delete
–
PostgreSQL
TCP
5432
Custom
sg-0e327b23154fb980f
sg-0e327b23154fb980f
Delete
Add rule
Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.
Cancel
Preview changes
Save rules
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Connect to instance | EC2
Instances | EC2 | us-east-1
ModifyInboundSecurityGr
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ModifyInboundSecurityGroupRules:securityGroupId=sg-0e327b23154fb980f
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
EC2 > Security Groups > sg-0e327b23154fb980f - MY-RDS-SG > Edit inbound rules
Edit inbound rules Info
Inbound rules control the incoming traffic that's allowed to reach the instance.
Inbound rules Info
Security group rule ID
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info
sgr-0835f2caa422b8929
PostgreSQL
TCP
5432
Custom
171.61.84.0/32
Delete
–
PostgreSQL
TCP
5432
Custom
sg-01945d3629624caa4
sg-01945d3629624caa4
Delete
Add rule
Cancel
Preview changes
Save rules
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

✅ 2️⃣ Run Prisma Commands: In EC2 run these commands

npx prisma generate

npx prisma migrate dev

npm run seed

Result:

- Database schema created
- Initial data inserted successfully

🧪 Final Connectivity Test

Backend API was accessed via EC2 public IP →
✅ Data retrieved from RDS confirming successful:

Check	Status
EC2 → RDS Connectivity	✅
Prisma Migration	✅
Application Data Response	✅

✅ RDS Deployment Completed!

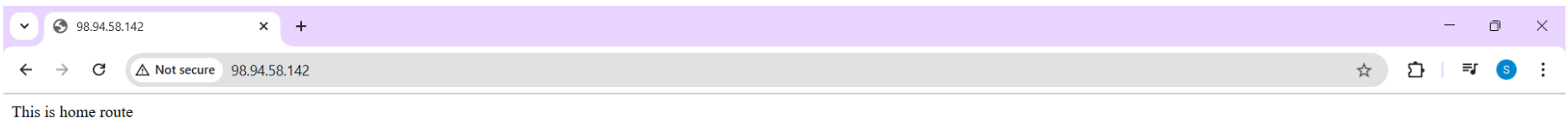
You now have:

- 🔹 Secure PostgreSQL running inside private AWS network
- 🔹 HA-ready architecture using multi-AZ design
- 🔹 Fully connected backend storing real data

🌎 Backend Public Access Test (EC2)

Once the backend deployment is complete:

✅ Copy the **Public IPv4 Address** of EC2
➞ Paste it into any browser
➞ Expected Result:



❌ **If the Backend is NOT Loading (Troubleshooting)**

Sometimes the backend may fail to start on **Port 80** due to a conflicting service already using it.

🧩 **Step-by-Step Fix**

✅ 1️⃣ **Check Which Process is Using Port 80**

Run either command:

sudo lsof -i :80

or

sudo netstat -tulnp | grep :80

Sample response:

nginx 1234 root 6u IPv4 123456 0t0 TCP *:80 (LISTEN)

➡ Another web server such as **nginx**, **Apache**, or a previous **Node.js process** is using port 80

✅ 2️⃣ **Resolve the Port Conflict**

You have **two safe options**:

✅ **— Free up Port 80 (Recommended)**

If the service running on 80 is not needed: **sudo kill -9 <PID>**

Replace <PID> with the ID identified above.

Then restart the backend:

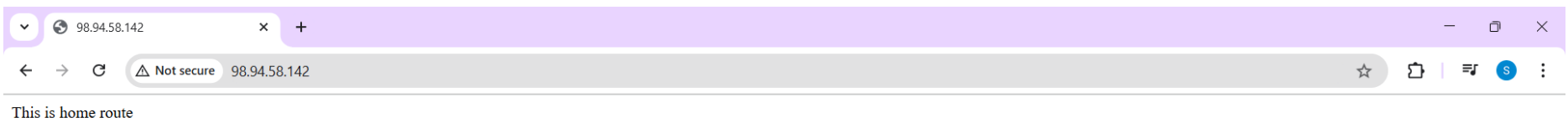
pm2 restart project-management

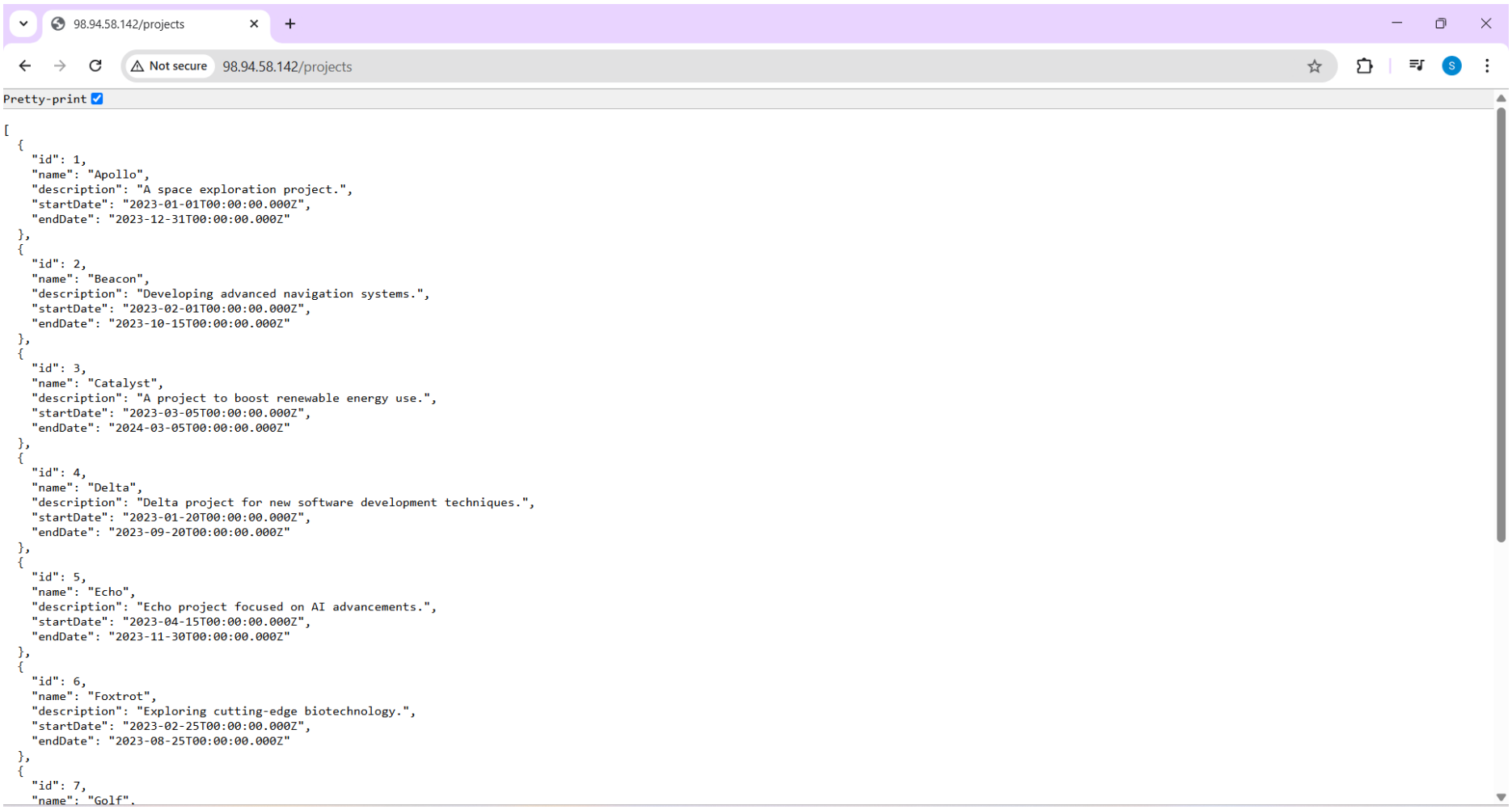
Check logs to confirm the app is running:

pm2 logs project-management

Expected logs: Server started on port 80 ✅

🧩 EXPECTED SCREENSHOT





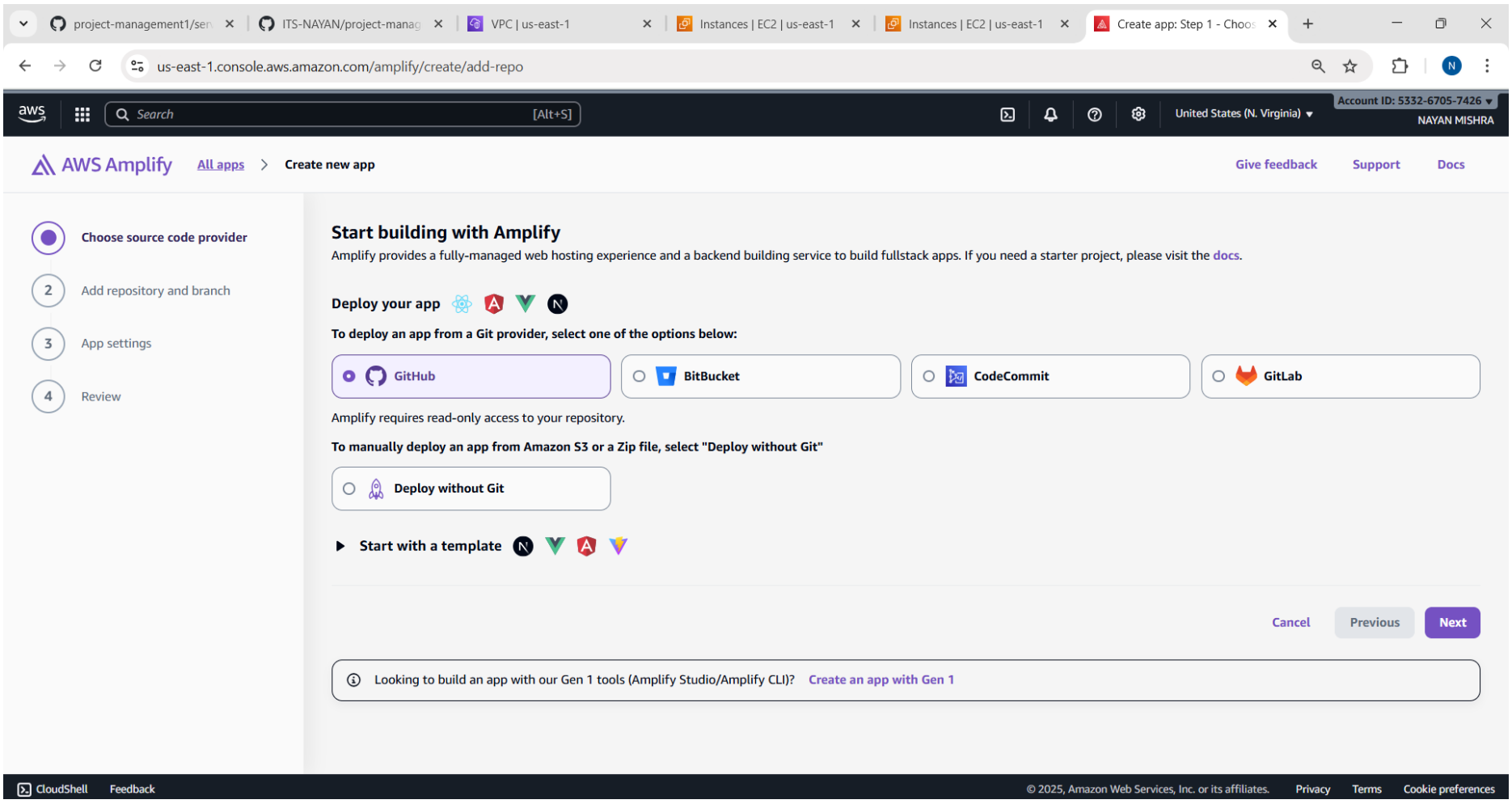
REFRENCE SCREENSHOT

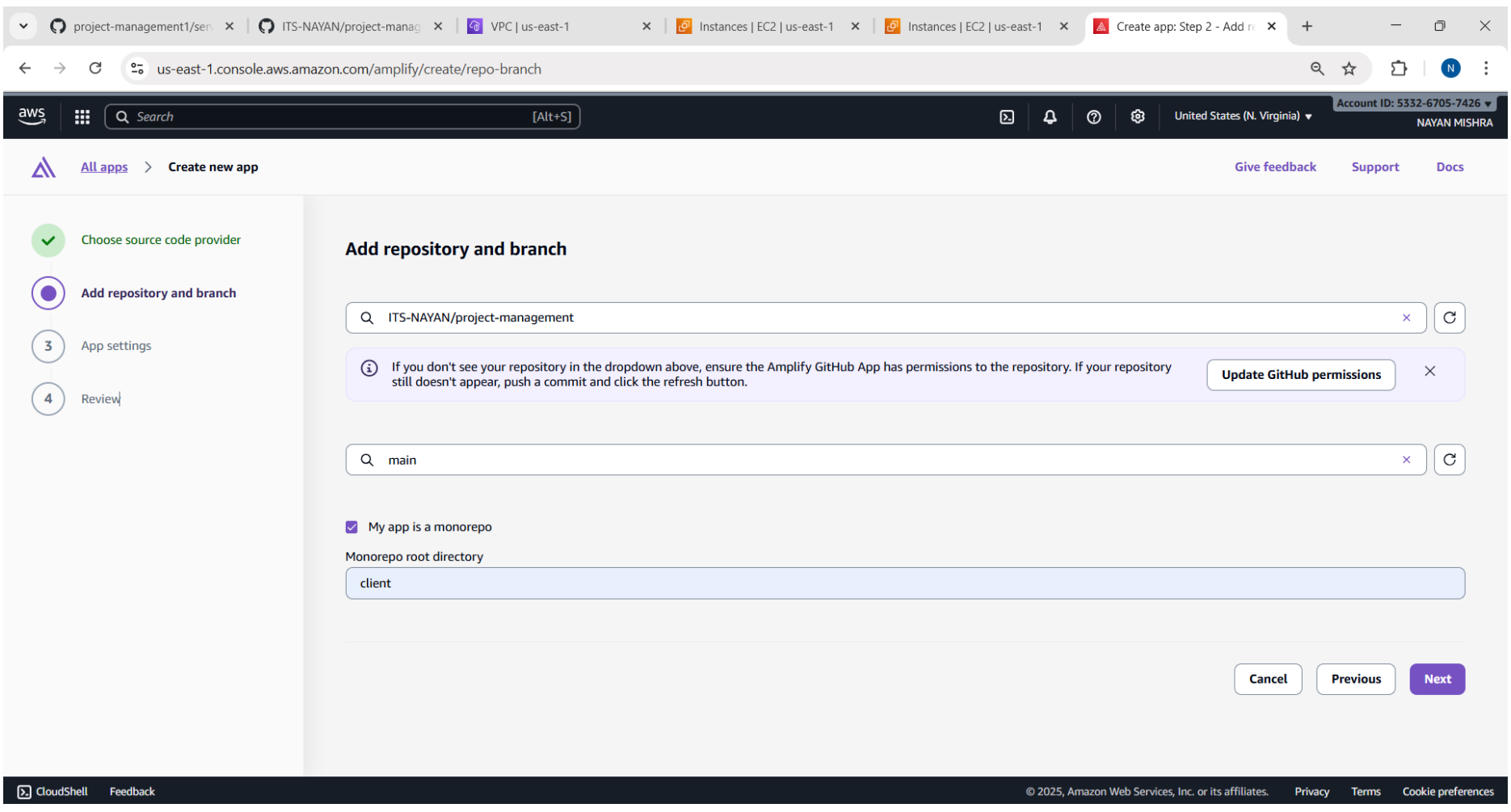
✅ Phase 3: Frontend Deployment & Secure API Access (Amplify + API Gateway)

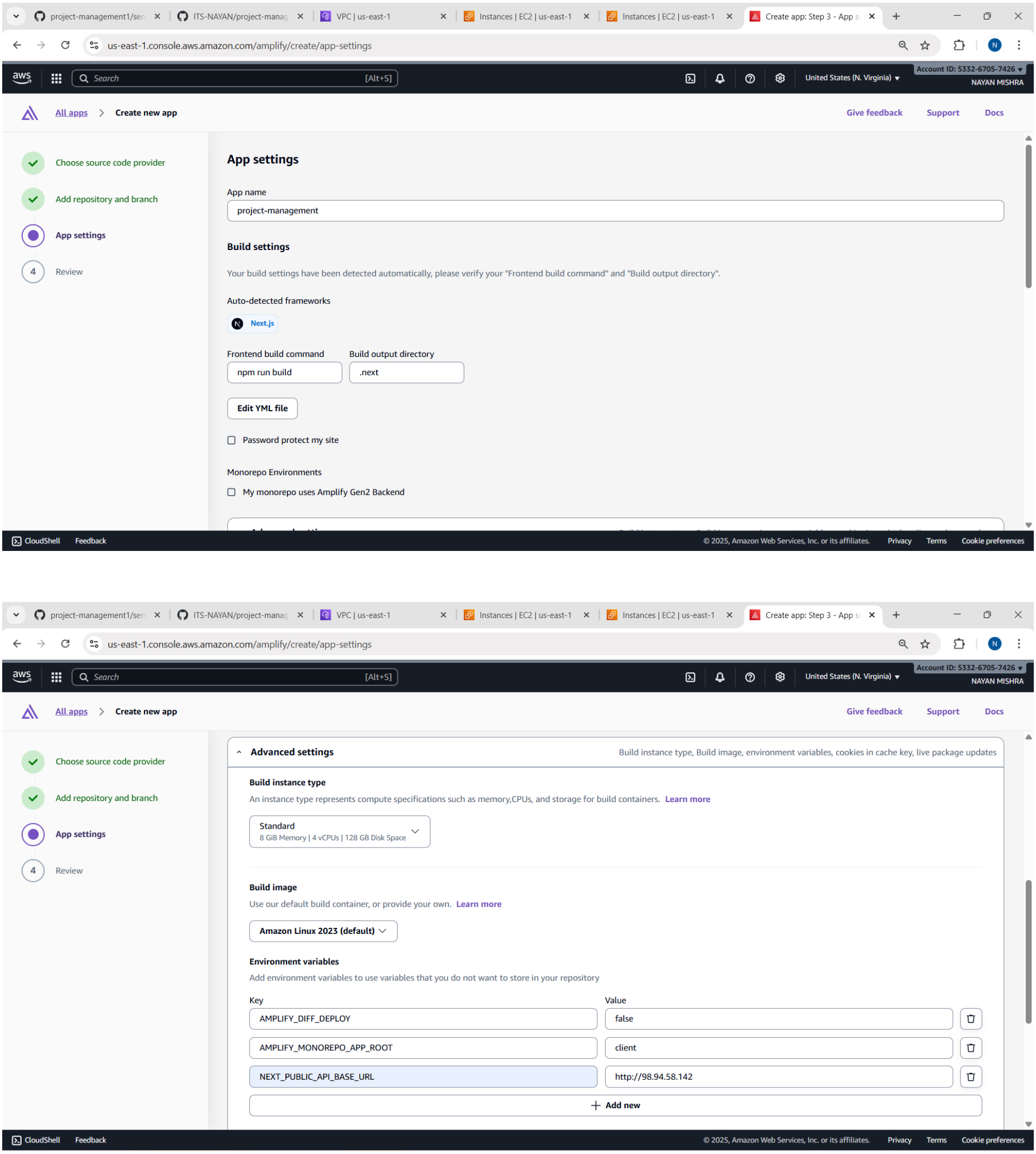
Objective: Deploy the Next.js frontend application through AWS Amplify and ensure secure, HTTPS-encrypted communication with the backend API, resolving browser mixed-content issues.

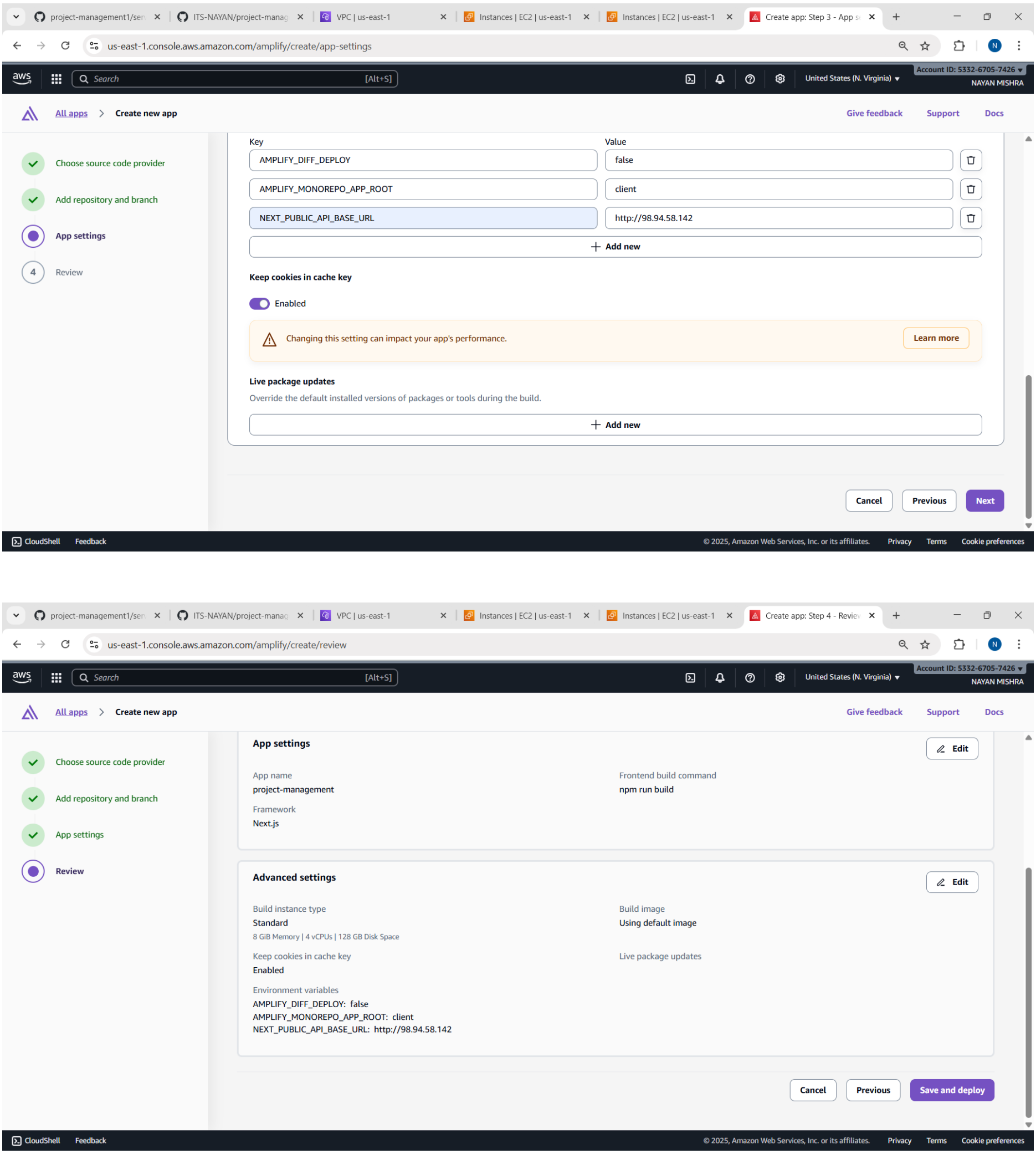
1️⃣ Frontend Deployment using AWS Amplify Hosting

- **Application Setup:**
  A new Amplify Hosting application was provisioned to manage frontend deployments.
- **GitHub Integration:**
  Amplify was configured to pull source code from the GitHub monorepo containing both frontend and backend components.
- **Monorepo Build Configuration:**
  The build root was explicitly set to the client/ directory to ensure only the Next.js frontend is built and deployed.
- **CI/CD Automation:**
  Amplify established an automated deployment pipeline where every Git commit triggers a new build and release process.
- **Initial Deployment Success:**
  The frontend was successfully launched and made publicly accessible over **HTTPS** with global CDN support.
- **Identified Problem — Mixed Content Error:**
  Since the frontend was served over **HTTPS**, but the backend API on EC2 was only accessible via **HTTP**, browsers blocked API requests due to **mixed-content** security enforcement.
  This resulted in failed data fetching and UI rendering disruptions.

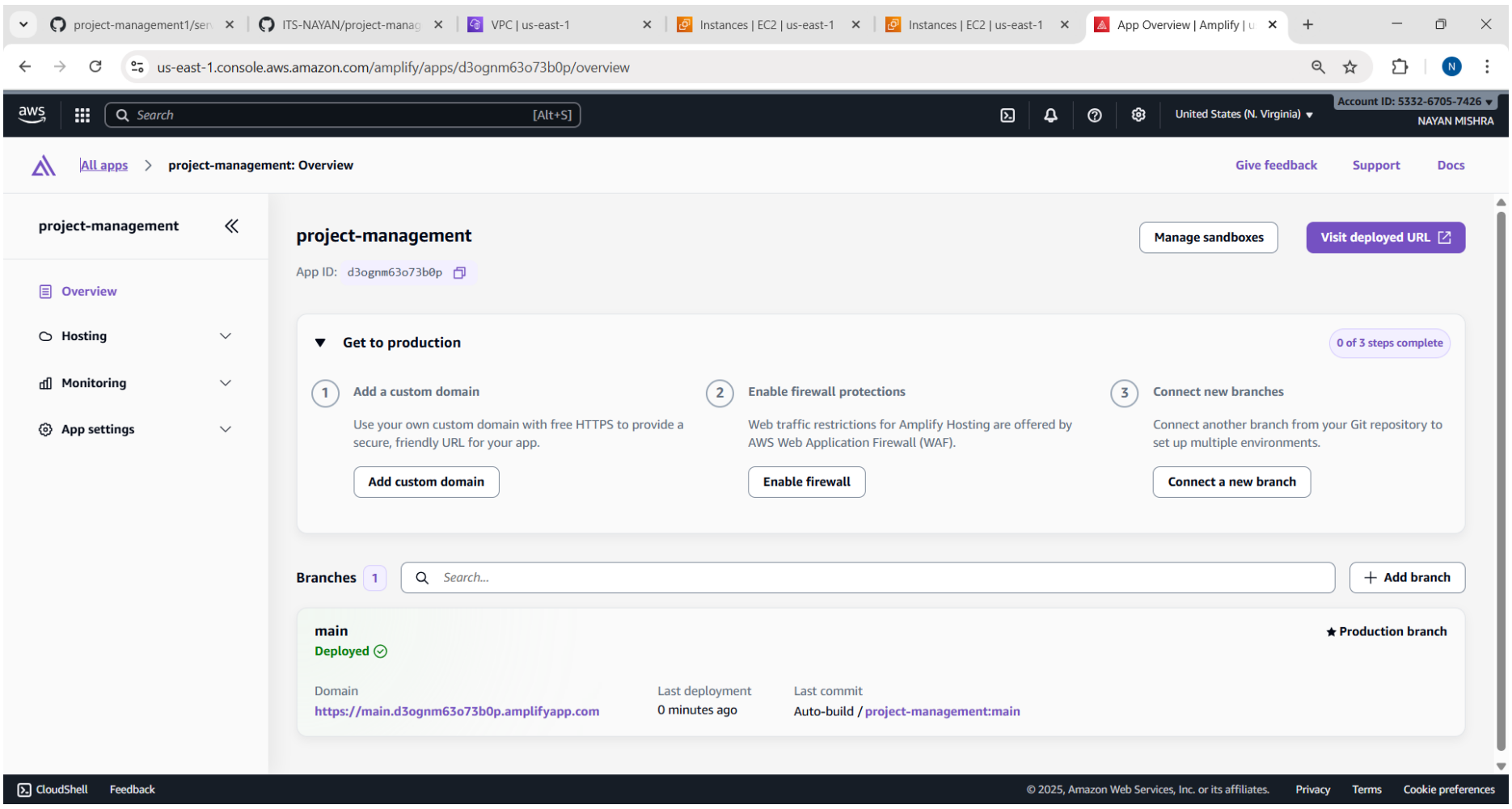

project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Instances | EC2 | us-east-1
Instances | EC2 | us-east-1
Create app: Step 1 - Choos
us-east-1.console.aws.amazon.com/amplify/create/add-repo
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
AWS Amplify
All apps
Create new app
Give feedback
Support
Docs
Choose source code provider
2 Add repository and branch
3 App settings
4 Review
Start building with Amplify
Amplify provides a fully-managed web hosting experience and a backend building service to build fullstack apps. If you need a starter project, please visit the docs.
Deploy your app
To deploy an app from a Git provider, select one of the options below:
GitHub
BitBucket
CodeCommit
GitLab
Amplify requires read-only access to your repository.
To manually deploy an app from Amazon S3 or a Zip file, select "Deploy without Git"
Deploy without Git
Start with a template
Cancel
Previous
Next
Looking to build an app with our Gen 1 tools (Amplify Studio/Amplify CLI)?
Create an app with Gen 1
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

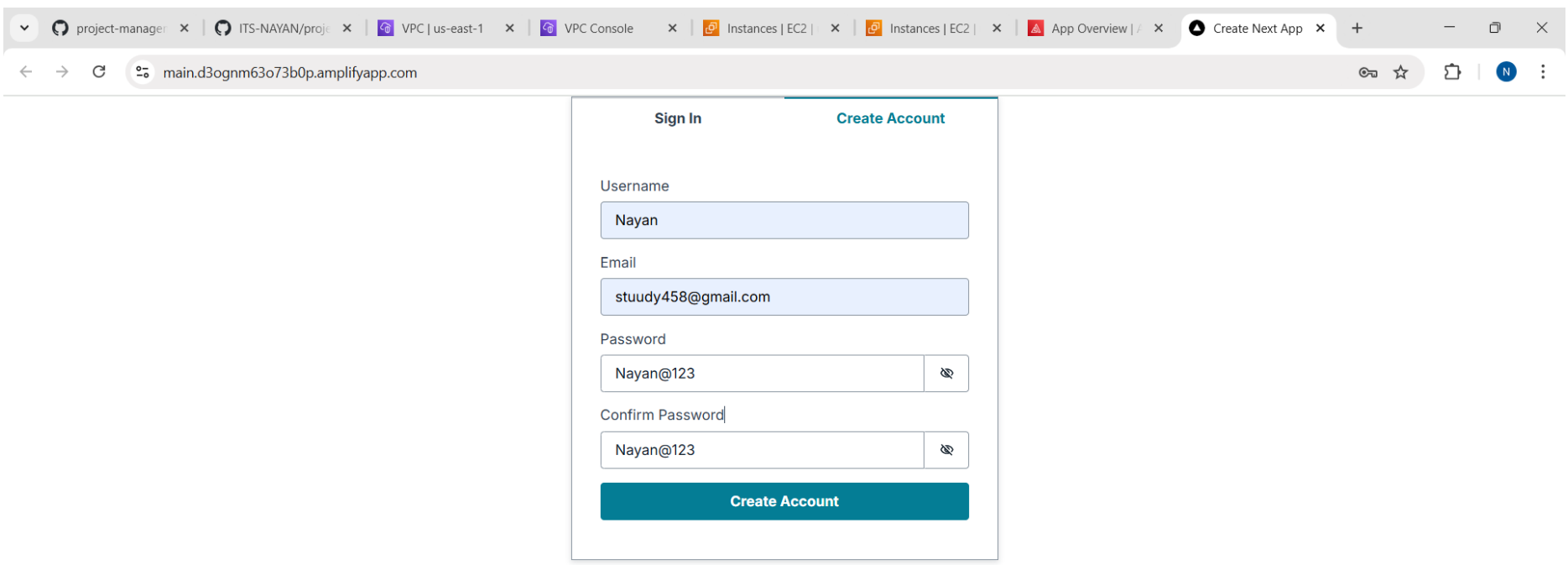

project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Instances | EC2 | us-east-1
Instances | EC2 | us-east-1
Create app: Step 2 - Add r
us-east-1.console.aws.amazon.com/amplify/create/repo-branch
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
All apps
Create new app
Give feedback
Support
Docs
Choose source code provider
Add repository and branch
3 App settings
4 Review
Add repository and branch
ITS-NAYAN/project-management
If you don't see your repository in the dropdown above, ensure the Amplify GitHub App has permissions to the repository. If your repository still doesn't appear, push a commit and click the refresh button.
Update GitHub permissions
main
My app is a monorepo
Monorepo root directory
client
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

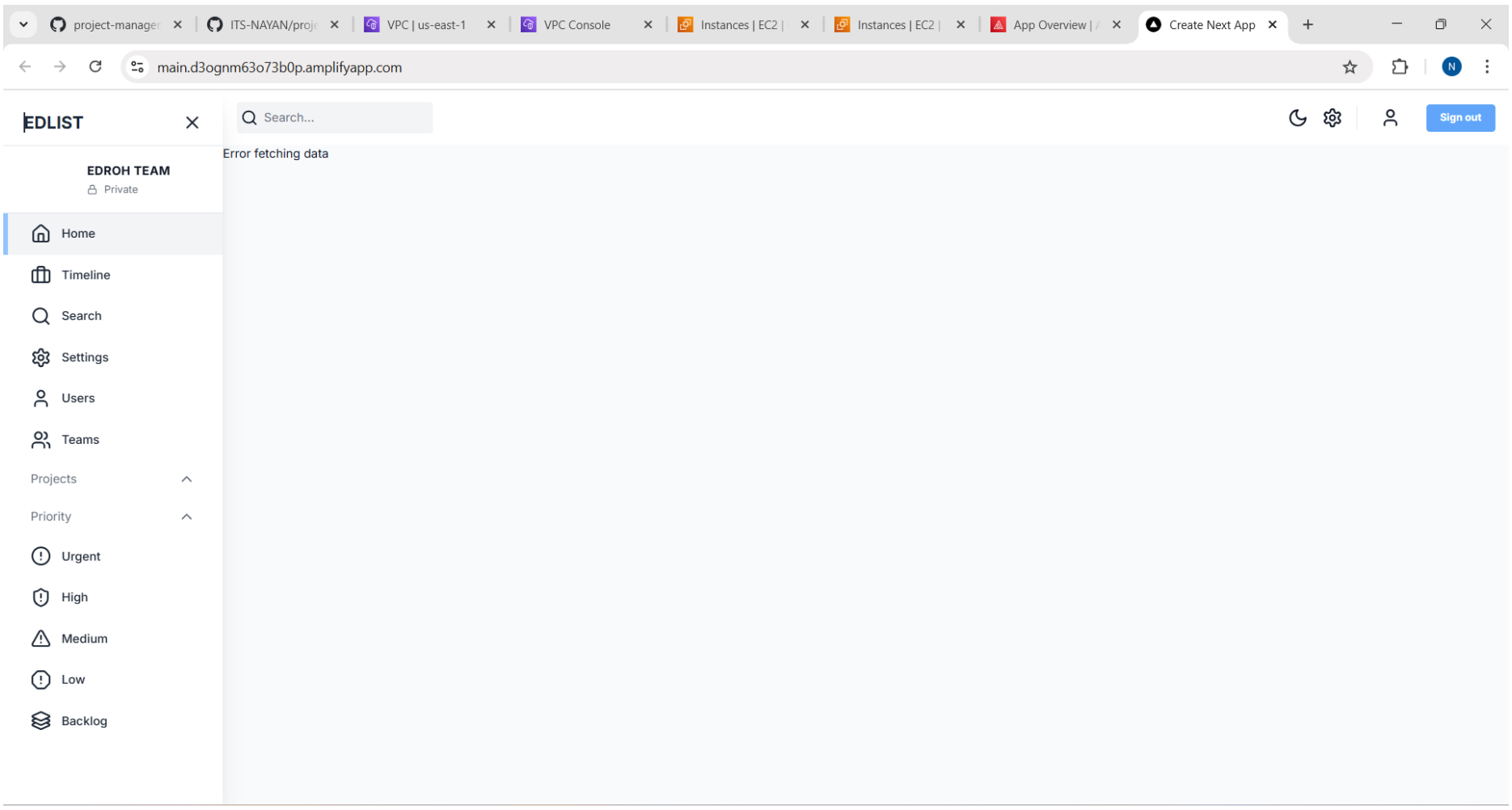

project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Instances | EC2 | us-east-1
Instances | EC2 | us-east-1
Create app: Step 3 - App s
us-east-1.console.aws.amazon.com/amplify/create/app-settings
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
All apps
Create new app
Give feedback
Support
Docs
Choose source code provider
Add repository and branch
App settings
4
Review
App settings
App name
project-management
Build settings
Your build settings have been detected automatically, please verify your "Frontend build command" and "Build output directory".
Auto-detected frameworks
Next.js
Frontend build command
npm run build
Build output directory
.next
Edit YML file
Password protect my site
Monorepo Environments
My monorepo uses Amplify Gen2 Backend
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Advanced settings
Build instance type, Build image, environment variables, cookies in cache key, live package updates
Build instance type
An instance type represents compute specifications such as memory,CPUs, and storage for build containers.
Learn more
Standard
8 GiB Memory | 4 vCPUs | 128 GB Disk Space
Build image
Use our default build container, or provide your own.
Learn more
Amazon Linux 2023 (default)
Environment variables
Add environment variables to use variables that you do not want to store in your repository
Key
Value
AMPLIFY_DIFF_DEPLOY
false
AMPLIFY_MONOREPO_APP_ROOT
client
NEXT_PUBLIC_API_BASE_URL
http://98.94.58.142
Add new


us-east-1.console.aws.amazon.com/amplify/create/app-settings
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
All apps
Create new app
Give feedback
Support
Docs
Choose source code provider
Add repository and branch
App settings
4 Review
Key
Value
AMPLIFY_DIFF_DEPLOY
false
AMPLIFY_MONOREPO_APP_ROOT
client
NEXT_PUBLIC_API_BASE_URL
http://98.94.58.142
Add new
Keep cookies in cache key
Enabled
Changing this setting can impact your app's performance.
Learn more
Live package updates
Override the default installed versions of packages or tools during the build.
Add new
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

us-east-1.console.aws.amazon.com/amplify/create/review
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
All apps
Create new app
Give feedback
Support
Docs
Choose source code provider
Add repository and branch
App settings
Review
App settings
Edit
App name
project-management
Framework
Next.js
Frontend build command
npm run build
Advanced settings
Edit
Build instance type
Standard
8 GiB Memory | 4 vCPUs | 128 GB Disk Space
Build image
Using default image
Keep cookies in cache key
Enabled
Live package updates
Environment variables
AMPLIFY_DIFF_DEPLOY: false
AMPLIFY_MONOREPO_APP_ROOT: client
NEXT_PUBLIC_API_BASE_URL: http://98.94.58.142
Cancel
Previous
Save and deploy
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


project-management1/ser
ITS-NAYAN/project-manag
VPC | us-east-1
Instances | EC2 | us-east-1
Instances | EC2 | us-east-1
App Overview | Amplify | u
us-east-1.console.aws.amazon.com/amplify/apps/d3ognm63o73b0p/overview
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
All apps
project-management: Overview
Give feedback
Support
Docs
project-management
Overview
Hosting
Monitoring
App settings
project-management
Manage sandboxes
Visit deployed URL
App ID: d3ognm63o73b0p
Get to production
0 of 3 steps complete
1 Add a custom domain
Use your own custom domain with free HTTPS to provide a secure, friendly URL for your app.
Add custom domain
2 Enable firewall protections
Web traffic restrictions for Amplify Hosting are offered by AWS Web Application Firewall (WAF).
Enable firewall
3 Connect new branches
Connect another branch from your Git repository to set up multiple environments.
Connect a new branch
Branches 1
Search...
Add branch
main
Deployed
★ Production branch
Domain
https://main.d3ognm63o73b0p.amplifyapp.com
Last deployment
0 minutes ago
Last commit
Auto-build / project-management:main
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

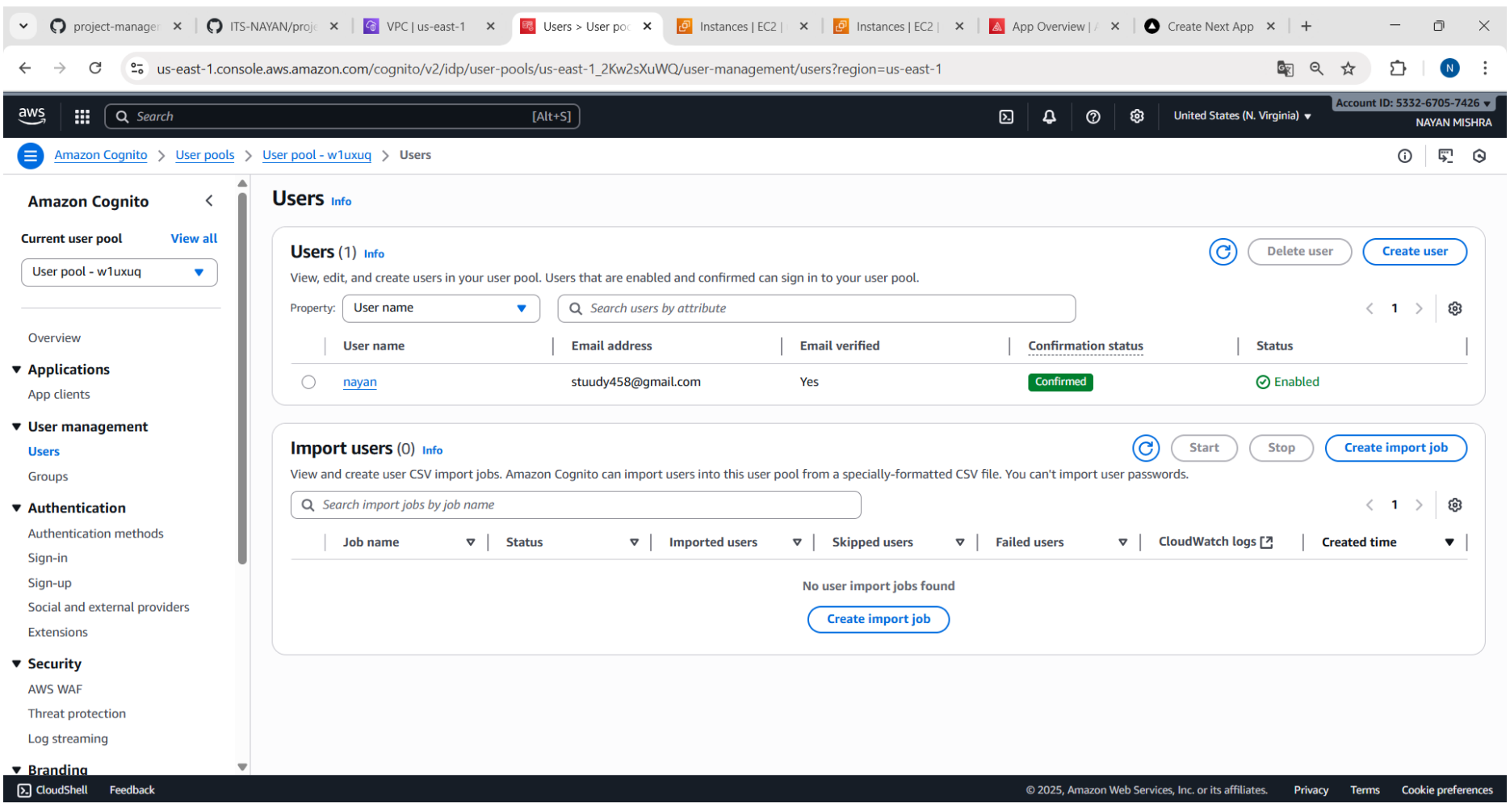

project-manager
ITS-NAYAN/proje
VPC | us-east-1
VPC Console
Instances | EC2 |
Instances | EC2 |
App Overview | /
Create Next App
main.d3ognm63o73b0p.amplifyapp.com
Sign In
Create Account
Username
Nayan
Email
stuudy458@gmail.com
Password
Nayan@123
Confirm Password
Nayan@123
Create Account



Identified Problem — Mixed Content Error:

Since the frontend was served over **HTTPS**, but the backend API on EC2 was only accessible via **HTTP**, browsers blocked API requests due to **mixed-content** security enforcement.
This resulted in failed data fetching and UI rendering disruptions.

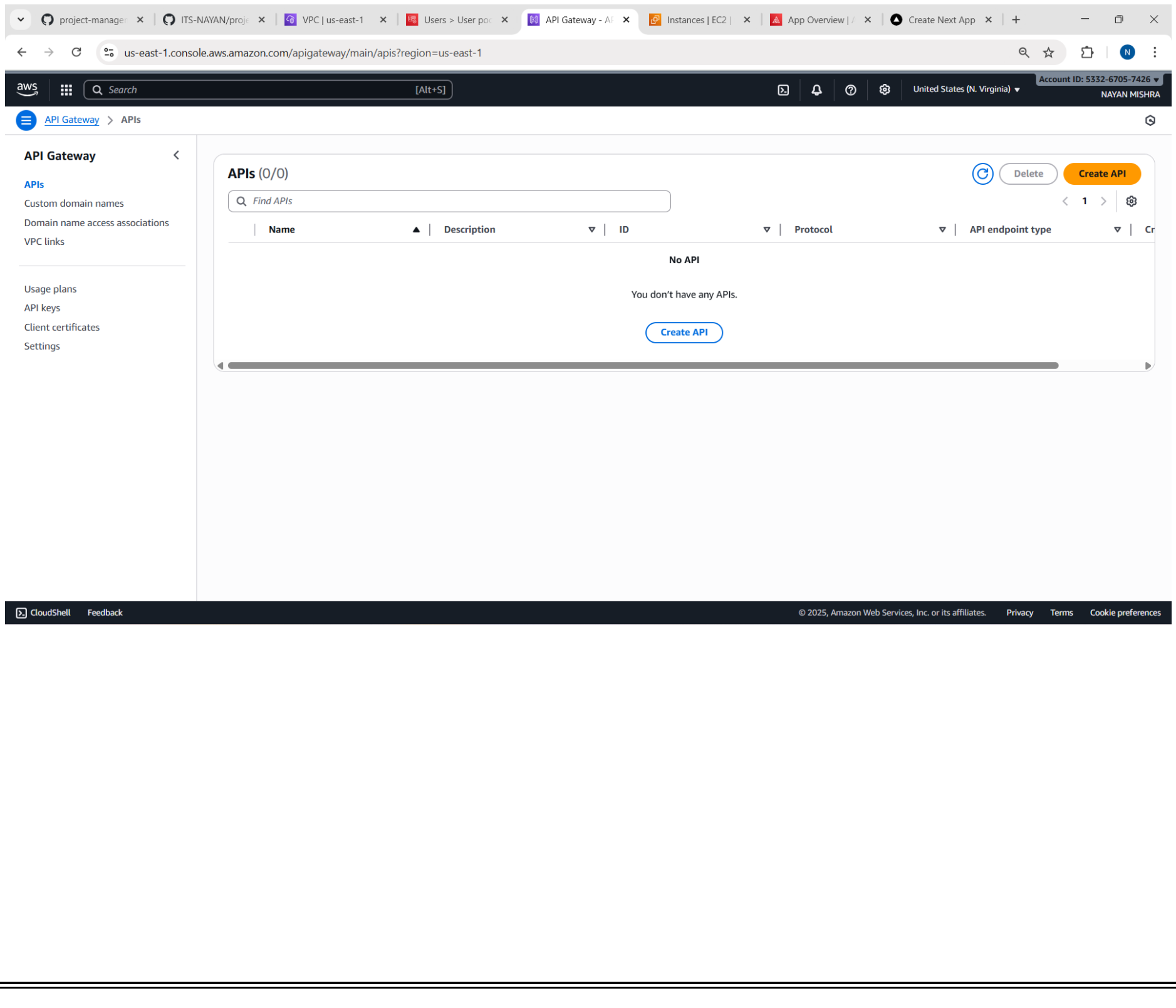
Also note user created in Cognito

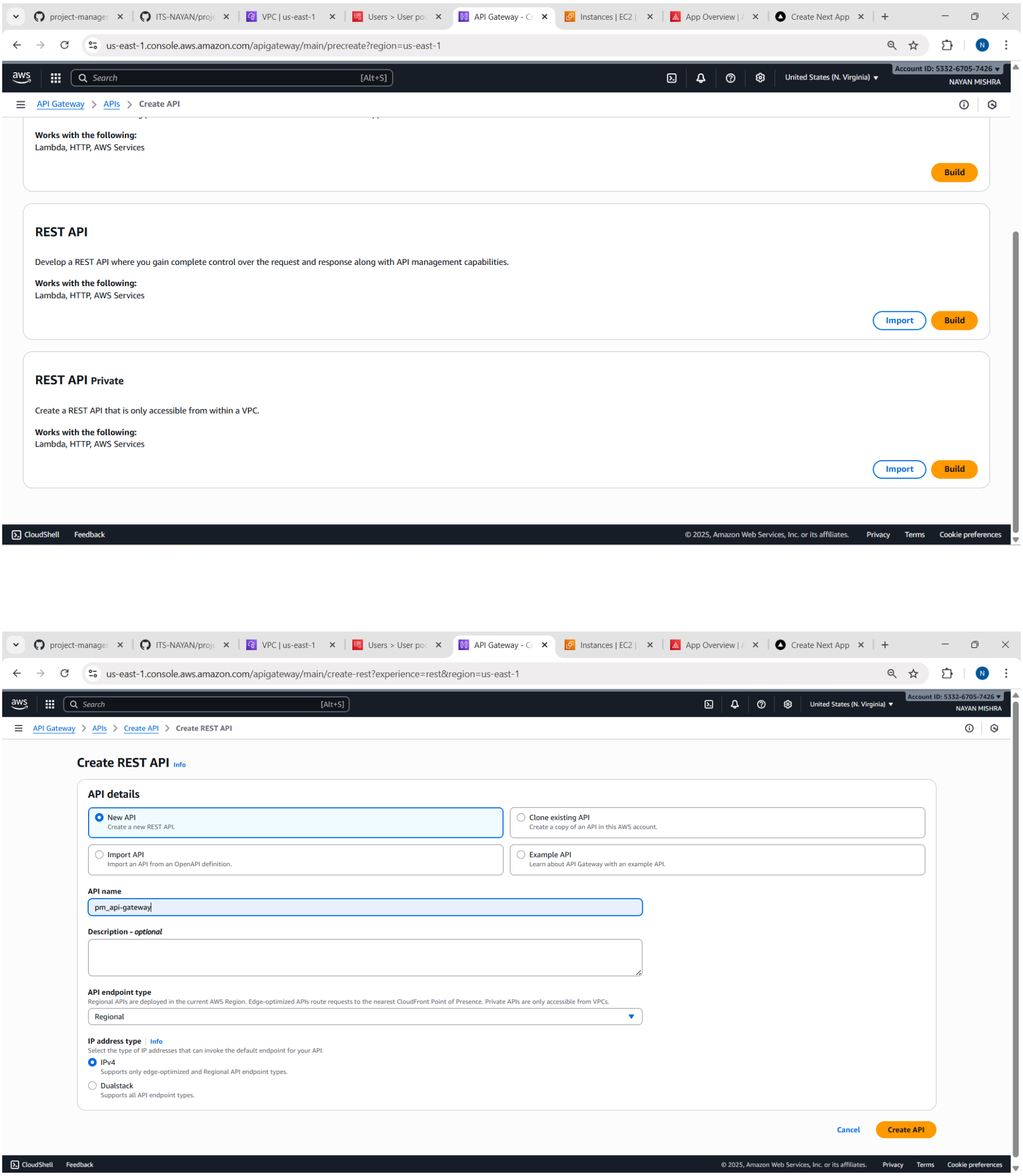


2️⃣ Solution: Securing Backend Access Using API Gateway

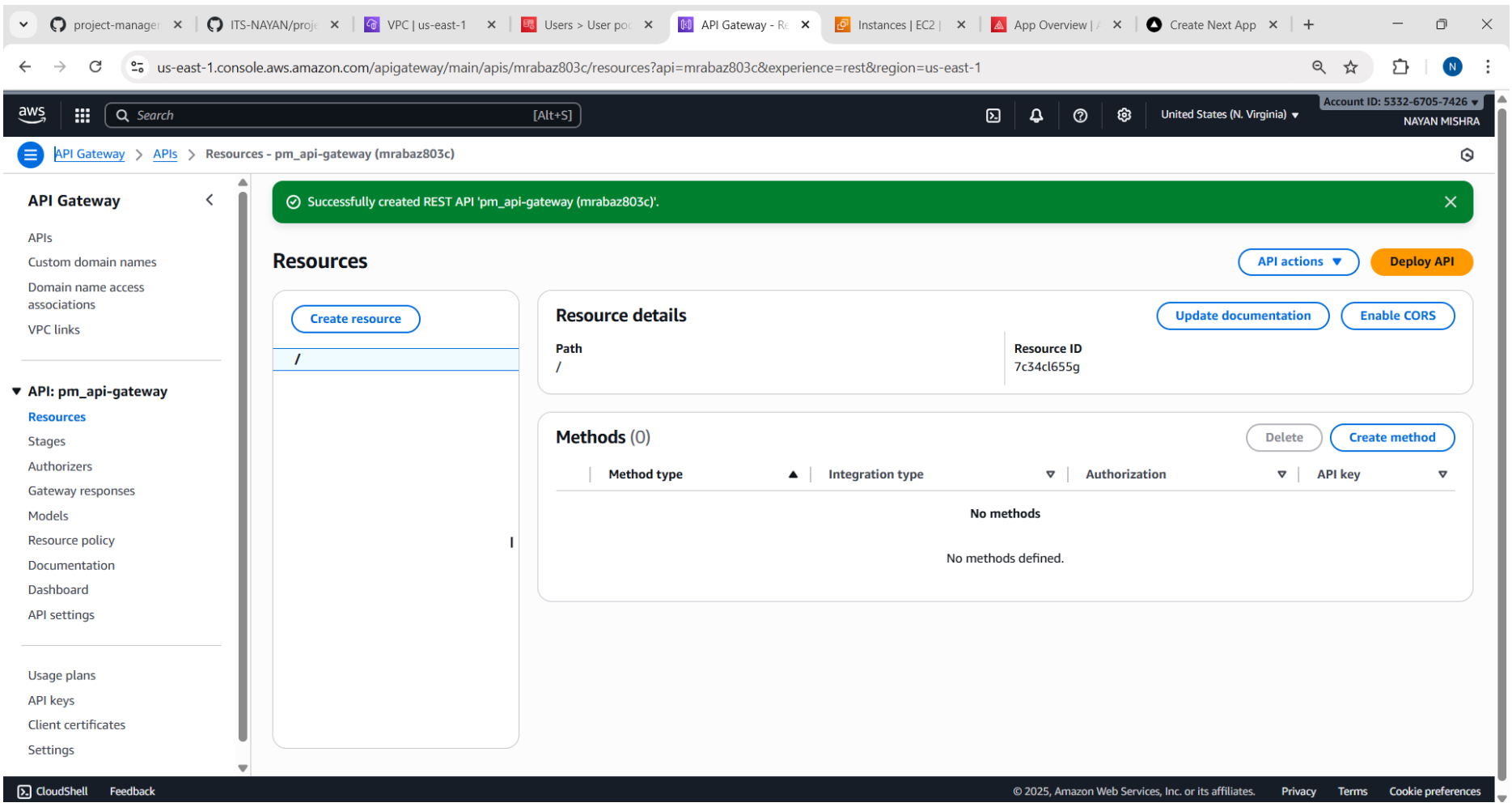
To eliminate mixed content issues and enable encrypted API communication, the EC2 backend was routed through Amazon API Gateway.

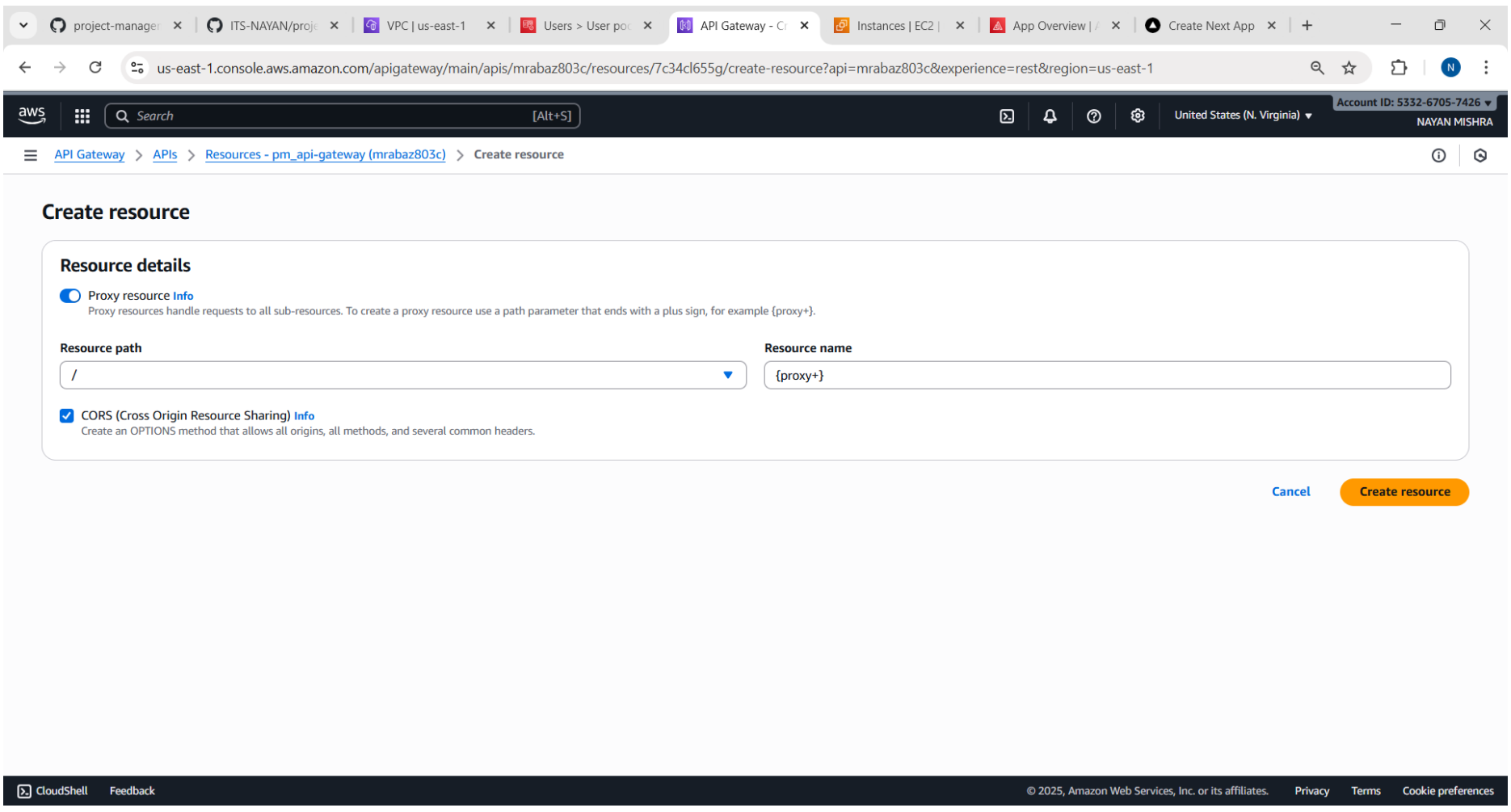
- **Purpose & Benefit:**
  API Gateway acts as a secure HTTPS entry point, providing **SSL/TLS certificate management**, **API routing**, and **network security** without modifying the EC2 instance.
- **API Gateway Type:**
  A **REST API** was created for full flexibility and ease of integration with HTTP backends.
- **Proxy Setup ({proxy+} Resource):**
  A catch-all proxy resource was created to forward all request paths dynamically to the EC2 backend — avoiding manual endpoint-by-endpoint setup.
- **CORS Enablement:**
  Cross-Origin Resource Sharing was enabled to allow browser-based clients to access the API.
- **HTTP Proxy Integration:**
  The proxy resource was configured to forward traffic directly to the backend EC2 instance's public HTTP endpoint:
  http://<EC2-Public-IP>/{proxy}
- **Deployment to prod Stage:**
  The API configuration was deployed to a production stage, automatically generating a **public HTTPS Invoke URL**.
- **Frontend Environment Update:**
  In Amplify, the NEXT_PUBLIC_API_BASE_URL variable was updated to use the new **HTTPS endpoint**.
- **Redeployment:**
  Amplify was redeployed so the updated secure API endpoint was applied to the live application.
- **Final Outcome — Mixed Content Resolved:**
  The frontend successfully communicated with the backend over **fully secure HTTPS**, eliminating security blocking and restoring full application functionality.

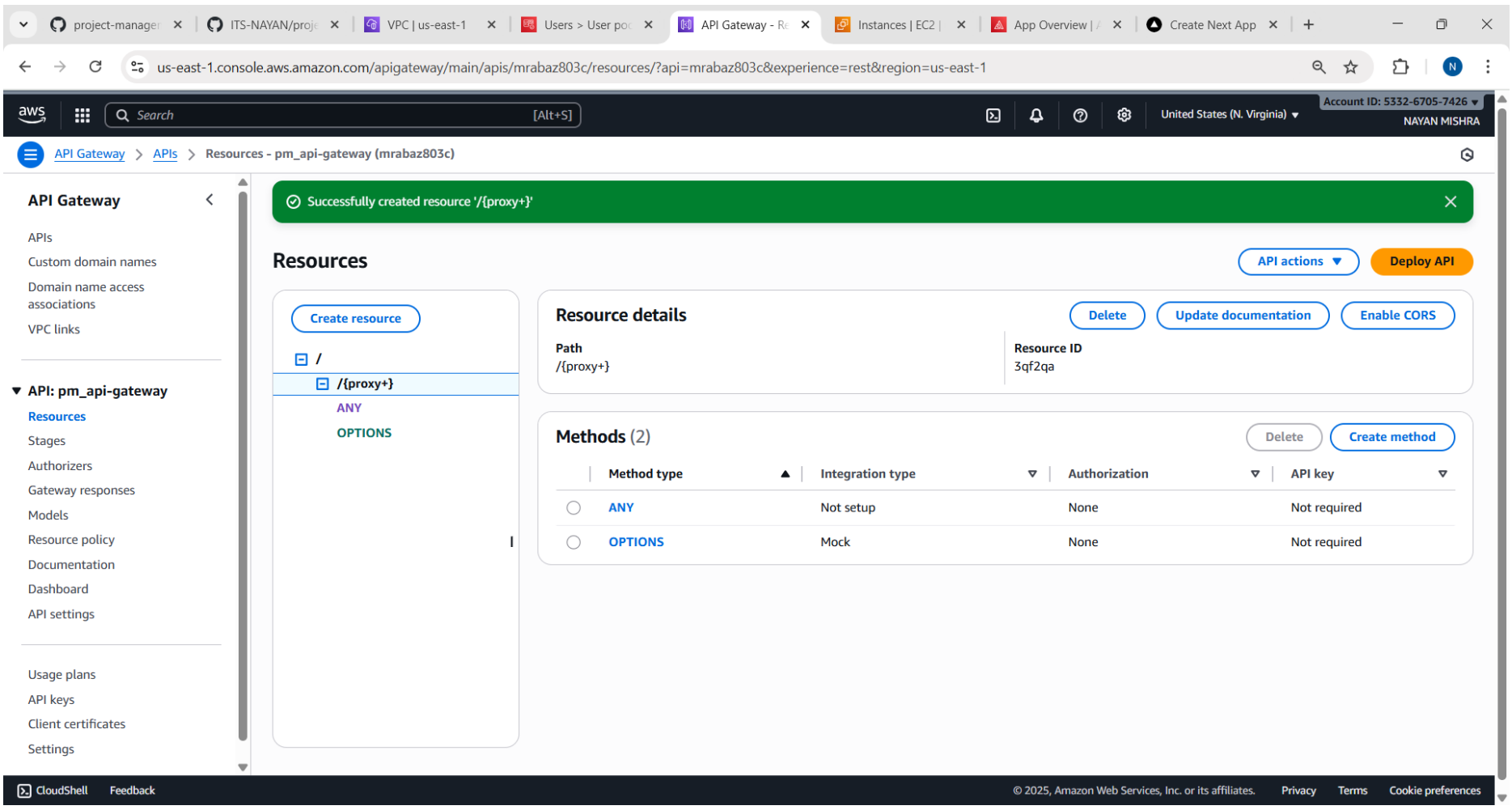


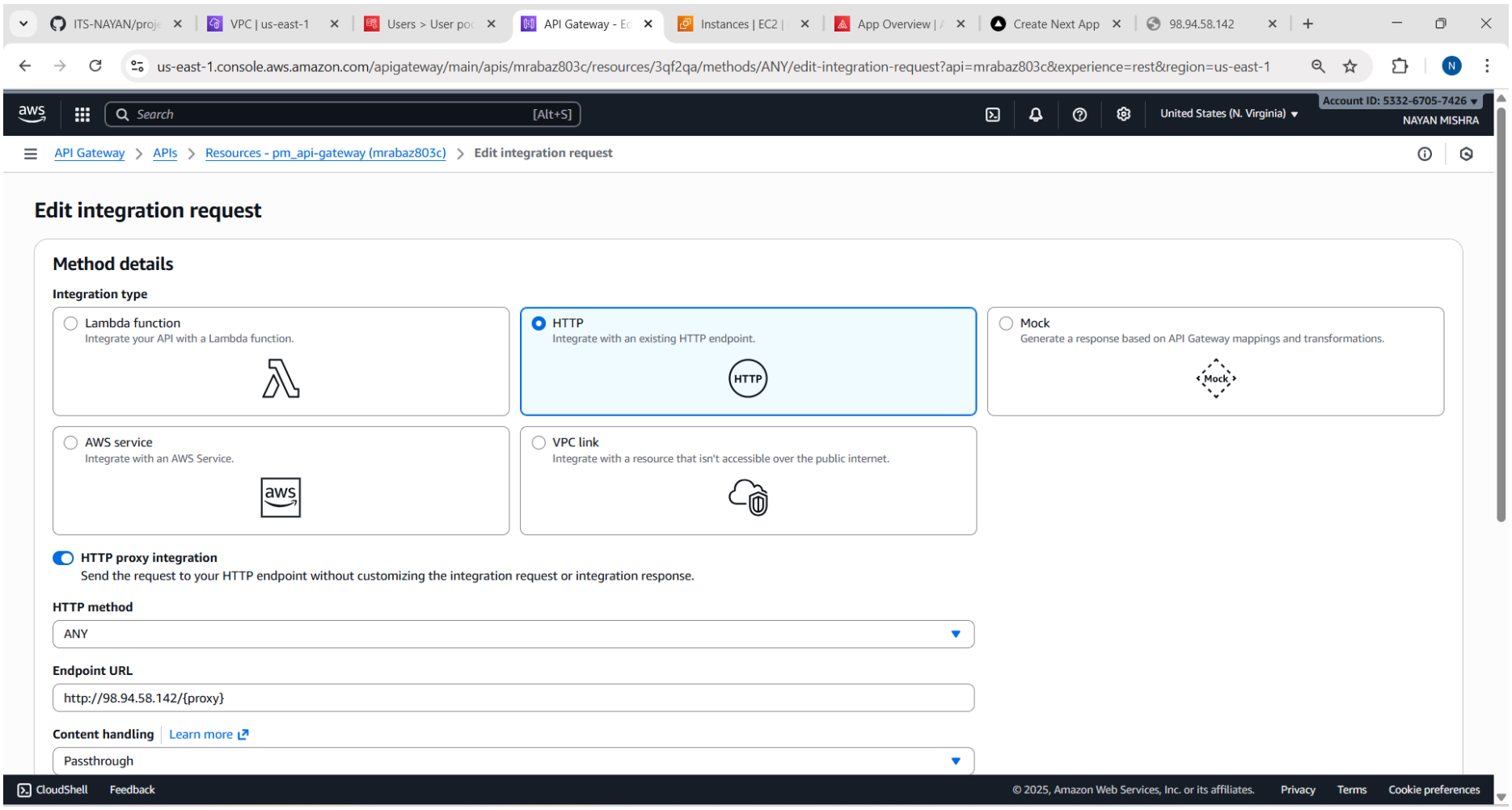

us-east-1.console.aws.amazon.com/apigateway/main/precreate?region=us-east-1
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
API Gateway > APIs > Create API
Works with the following:
Lambda, HTTP, AWS Services
Build
REST API
Develop a REST API where you gain complete control over the request and response along with API management capabilities.
Works with the following:
Lambda, HTTP, AWS Services
Import
Build
REST API Private
Create a REST API that is only accessible from within a VPC.
Works with the following:
Lambda, HTTP, AWS Services
Import
Build
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

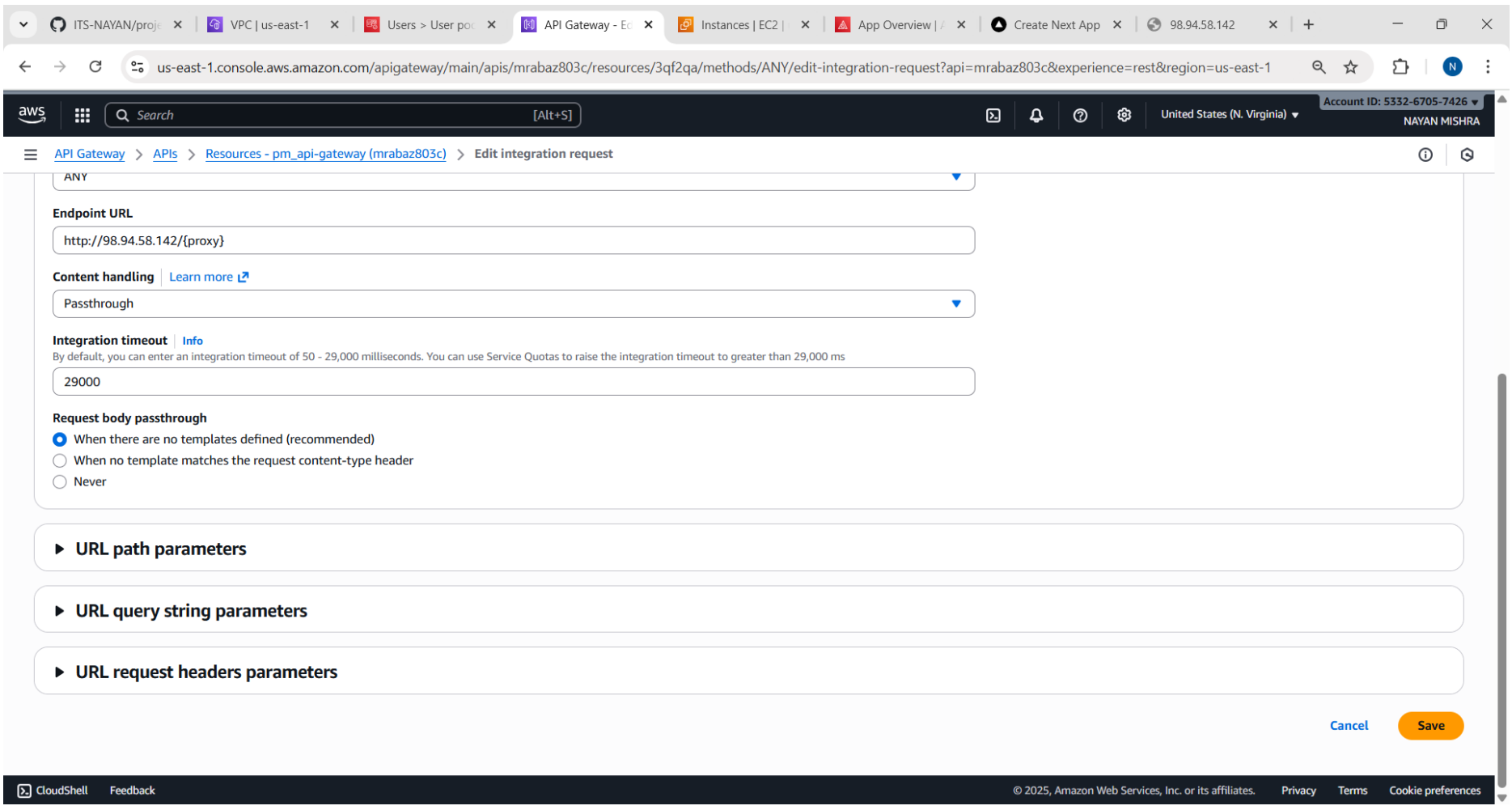
us-east-1.console.aws.amazon.com/apigateway/main/create-rest?experience=rest&region=us-east-1
API Gateway > APIs > Create API > Create REST API
Create REST API Info
API details
New API
Create a new REST API.
Clone existing API
Create a copy of an API in this AWS account.
Import API
Import an API from an OpenAPI definition.
Example API
Learn about API Gateway with an example API.
API name
pm_api-gateway
Description - optional
API endpoint type
Regional APIs are deployed in the current AWS Region. Edge-optimized APIs route requests to the nearest CloudFront Point of Presence. Private APIs are only accessible from VPCs.
Regional
IP address type Info
Select the type of IP addresses that can invoke the default endpoint for your API.
IPv4
Supports only edge-optimized and Regional API endpoint types.
Dualstack
Supports all API endpoint types.
Cancel
Create API
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

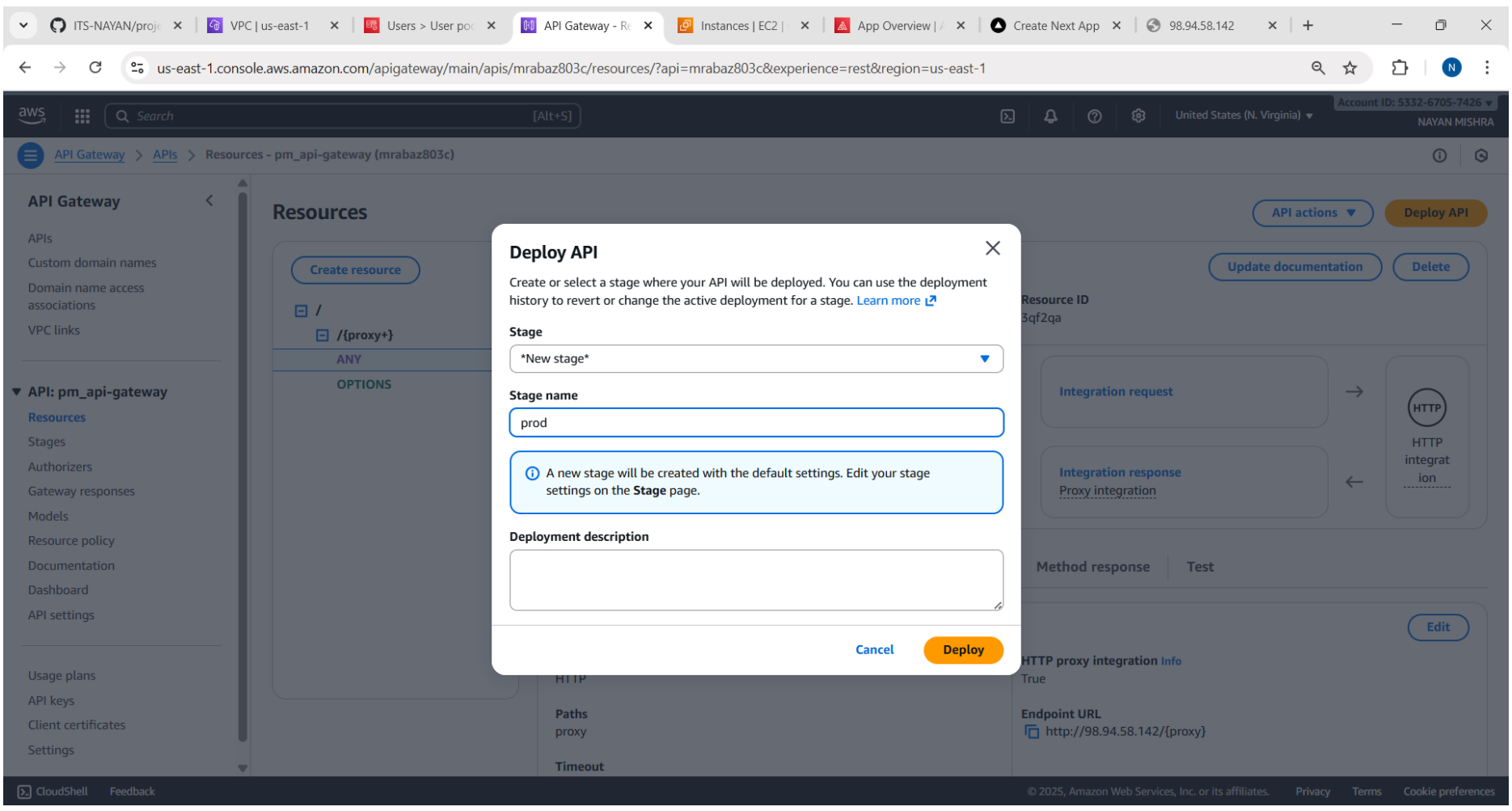

us-east-1.console.aws.amazon.com/apigateway/main/apis/mrabaz803c/resources?api=mrabaz803c&experience=rest&region=us-east-1
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
API Gateway > APIs > Resources - pm_api-gateway (mrabaz803c)
API Gateway
APIs
Custom domain names
Domain name access associations
VPC links
API: pm_api-gateway
Resources
Stages
Authorizers
Gateway responses
Models
Resource policy
Documentation
Dashboard
API settings
Usage plans
API keys
Client certificates
Settings
Successfully created REST API 'pm_api-gateway (mrabaz803c)'.
Resources
API actions
Deploy API
Create resource
/
Resource details
Update documentation
Enable CORS
Path
/
Resource ID
7c34cl655g
Methods (0)
Delete
Create method
Method type
Integration type
Authorization
API key
No methods
No methods defined.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

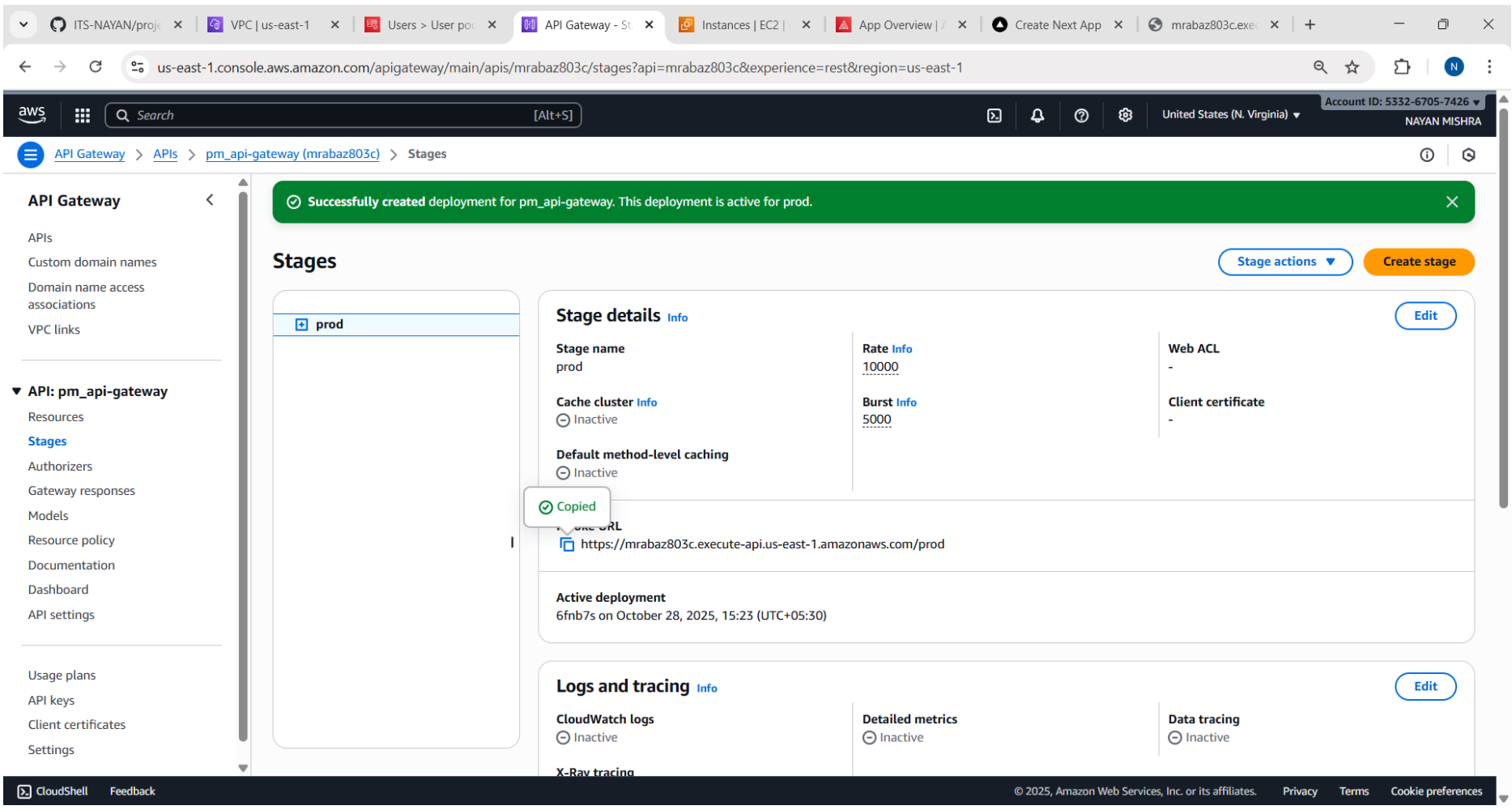

us-east-1.console.aws.amazon.com/apigateway/main/apis/mrabaz803c/resources/7c34cl655g/create-resource?api=mrabaz803c&experience=rest&region=us-east-1
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
API Gateway > APIs > Resources - pm_api-gateway (mrabaz803c) > Create resource
Create resource
Resource details
Proxy resource Info
Proxy resources handle requests to all sub-resources. To create a proxy resource use a path parameter that ends with a plus sign, for example {proxy+}.
Resource path
/
Resource name
{proxy+}
CORS (Cross Origin Resource Sharing) Info
Create an OPTIONS method that allows all origins, all methods, and several common headers.
Cancel
Create resource
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

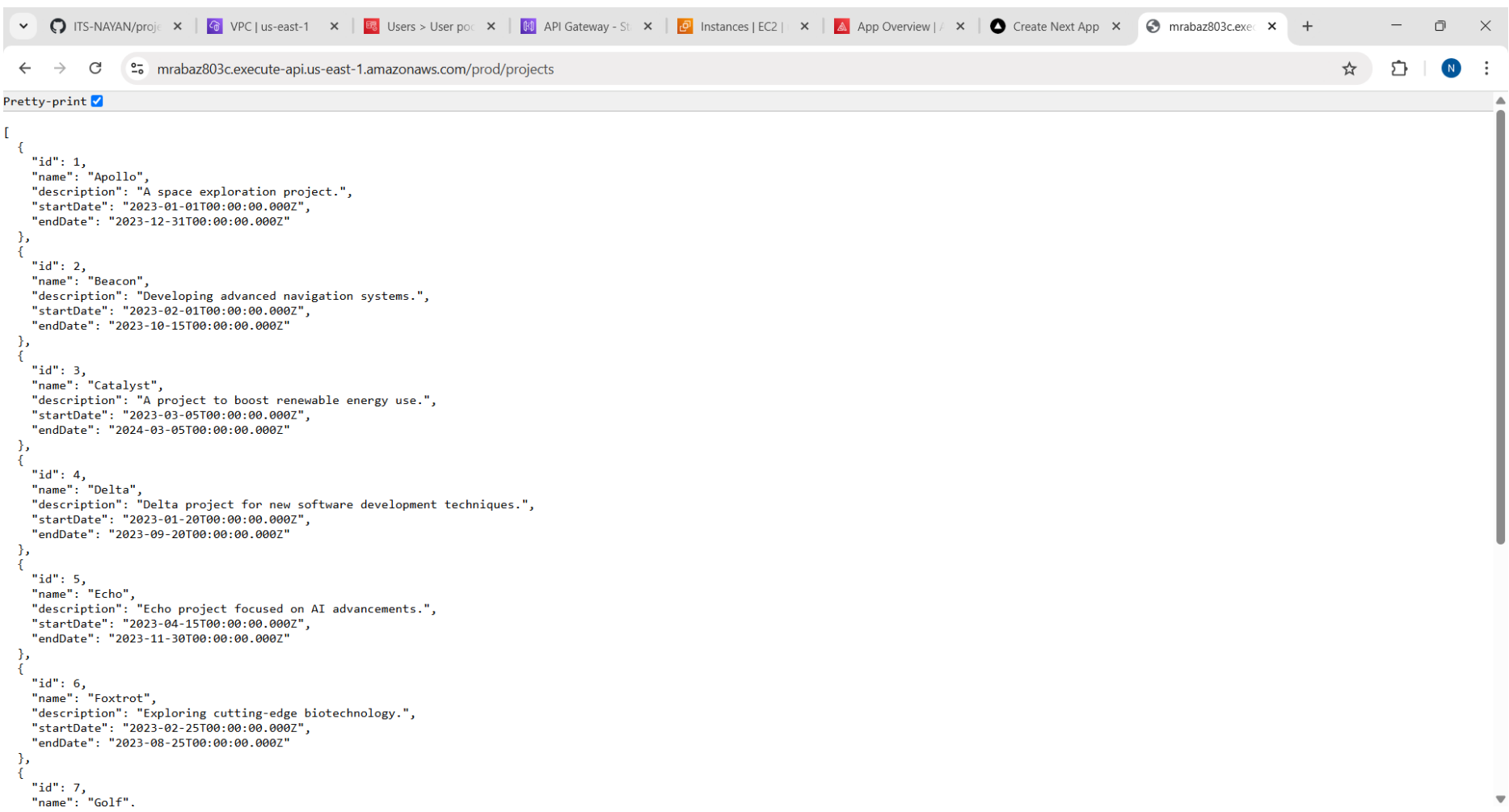

us-east-1.console.aws.amazon.com/apigateway/main/apis/mrabaz803c/resources/?api=mrabaz803c&experience=rest&region=us-east-1
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
API Gateway > APIs > Resources - pm_api-gateway (mrabaz803c)
API Gateway
APIs
Custom domain names
Domain name access associations
VPC links
API: pm_api-gateway
Resources
Stages
Authorizers
Gateway responses
Models
Resource policy
Documentation
Dashboard
API settings
Usage plans
API keys
Client certificates
Settings
Successfully created resource '/{proxy+}'
Resources
API actions
Deploy API
Create resource
/
/{proxy+}
ANY
OPTIONS
Resource details
Delete
Update documentation
Enable CORS
Path
/{proxy+}
Resource ID
3qf2qa
Methods (2)
Delete
Create method
Method type
Integration type
Authorization
API key
ANY
Not setup
None
Not required
OPTIONS
Mock
None
Not required
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/apigateway/main/apis/mrabaz803c/resources/3qf2qa/methods/ANY/edit-integration-request?api=mrabaz803c&experience=rest&region=us-east-1
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
API Gateway > APIs > Resources - pm_api-gateway (mrabaz803c) > Edit integration request
Edit integration request
Method details
Integration type
Lambda function
Integrate your API with a Lambda function.
HTTP
Integrate with an existing HTTP endpoint.
Mock
Generate a response based on API Gateway mappings and transformations.
AWS service
Integrate with an AWS Service.
VPC link
Integrate with a resource that isn't accessible over the public internet.
HTTP proxy integration
Send the request to your HTTP endpoint without customizing the integration request or integration response.
HTTP method
ANY
Endpoint URL
http://98.94.58.142/{proxy}
Content handling
Learn more
Passthrough
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/apigateway/main/apis/mrabaz803c/resources/3qf2qa/methods/ANY/edit-integration-request?api=mrabaz803c&experience=rest&region=us-east-1
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
API Gateway > APIs > Resources - pm_api-gateway (mrabaz803c) > Edit integration request
ANY
Endpoint URL
http://98.94.58.142/{proxy}
Content handling Learn more
Passthrough
Integration timeout Info
By default, you can enter an integration timeout of 50 - 29,000 milliseconds. You can use Service Quotas to raise the integration timeout to greater than 29,000 ms
29000
Request body passthrough
When there are no templates defined (recommended)
When no template matches the request content-type header
Never
URL path parameters
URL query string parameters
URL request headers parameters
Cancel
Save
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

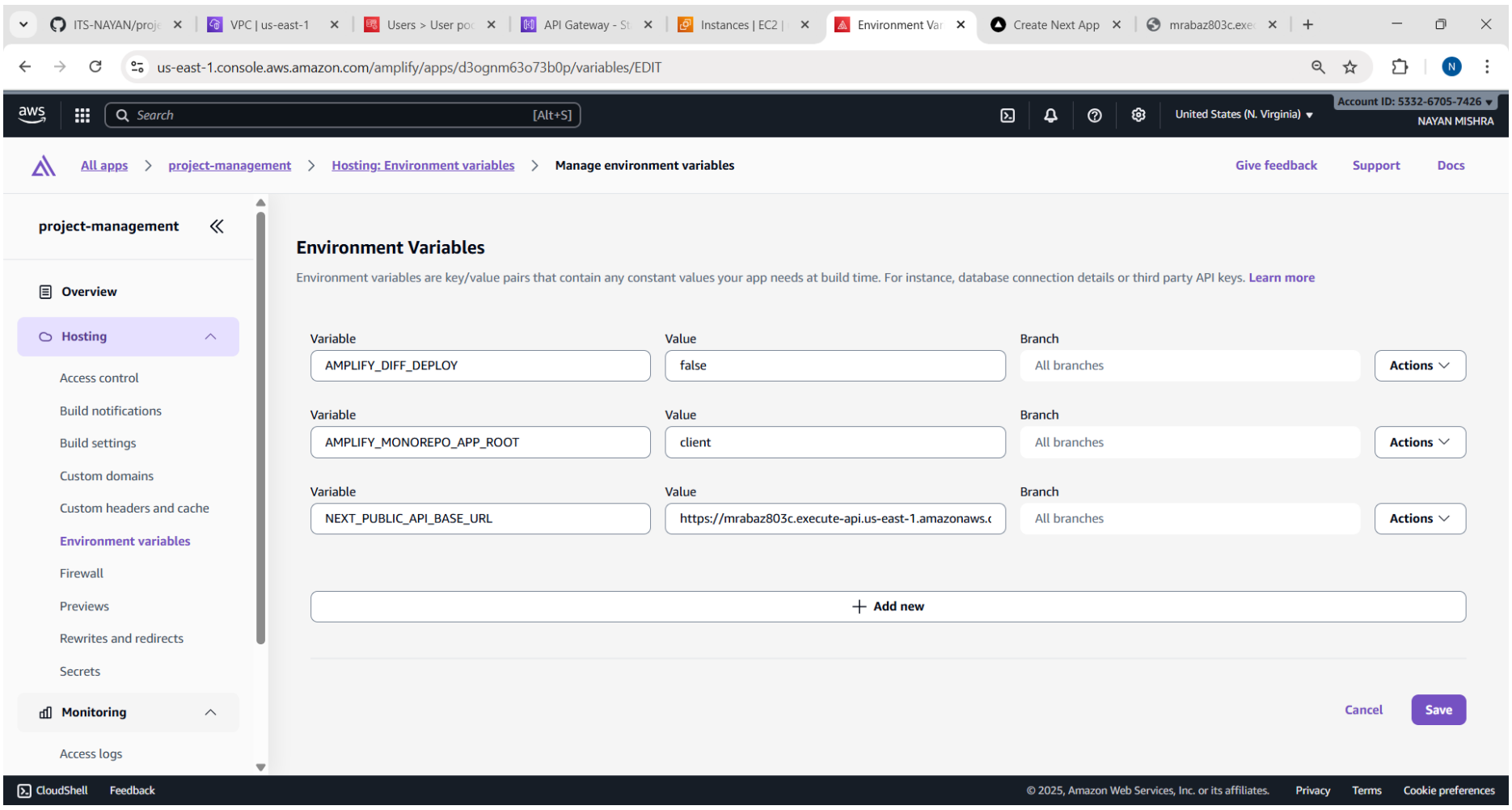

us-east-1.console.aws.amazon.com/apigateway/main/apis/mrabaz803c/resources/?api=mrabaz803c&experience=rest&region=us-east-1
API Gateway > APIs > Resources - pm_api-gateway (mrabaz803c)
API Gateway
APIs
Custom domain names
Domain name access associations
VPC links
API: pm_api-gateway
Resources
Stages
Authorizers
Gateway responses
Models
Resource policy
Documentation
Dashboard
API settings
Usage plans
API keys
Client certificates
Settings
Resources
Create resource
/
/{proxy+}
ANY
OPTIONS
API actions
Deploy API
Update documentation
Delete
Resource ID
3qf2qa
Integration request
Integration response
Proxy integration
HTTP integration
Method response
Test
Edit
HTTP proxy integration Info
True
Endpoint URL
http://98.94.58.142/{proxy}
Paths
proxy
Timeout
HTTP
Deploy API
Create or select a stage where your API will be deployed. You can use the deployment history to revert or change the active deployment for a stage. Learn more
Stage
New stage
Stage name
prod
A new stage will be created with the default settings. Edit your stage settings on the Stage page.
Deployment description
Cancel
Deploy
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

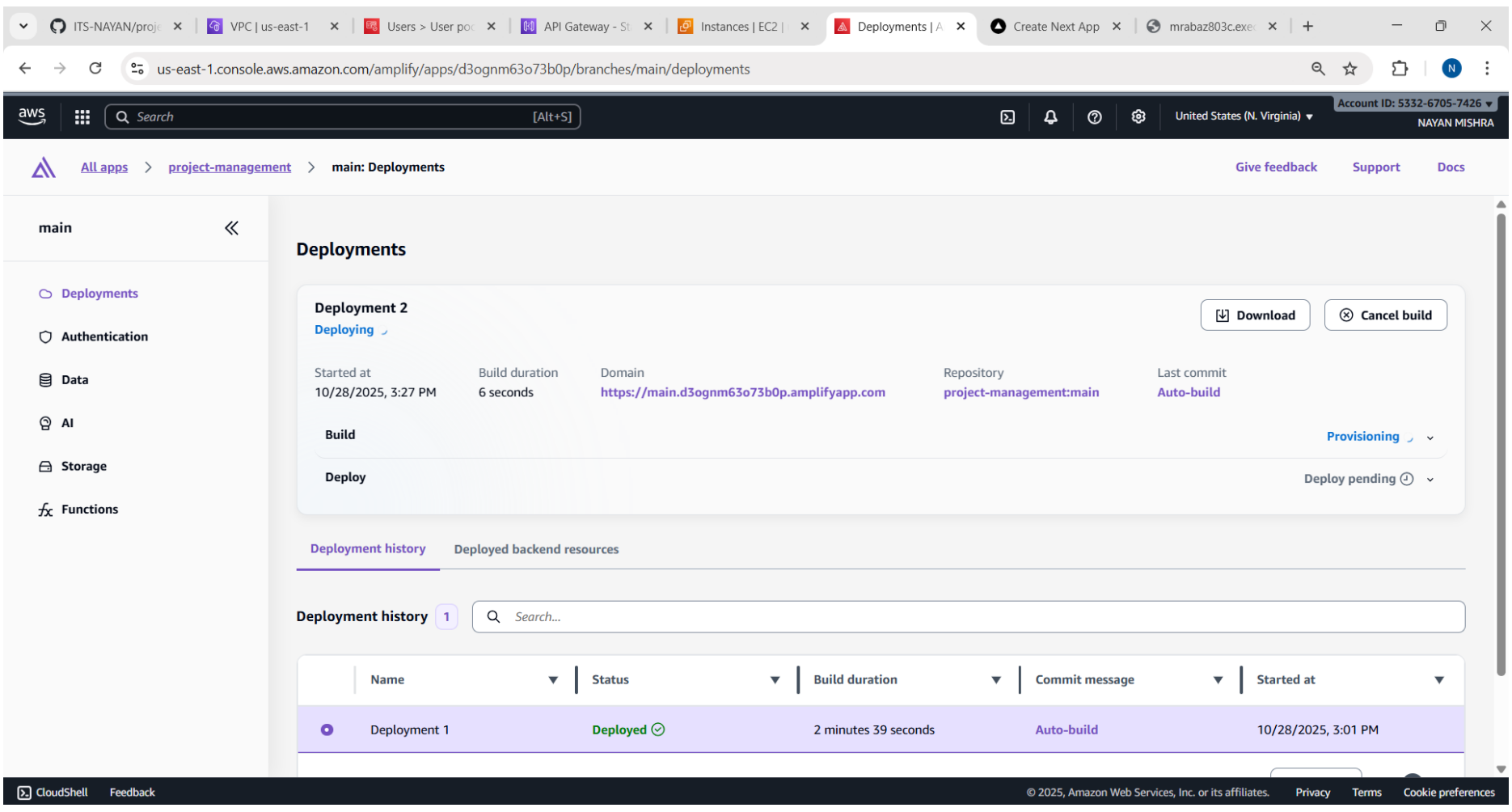




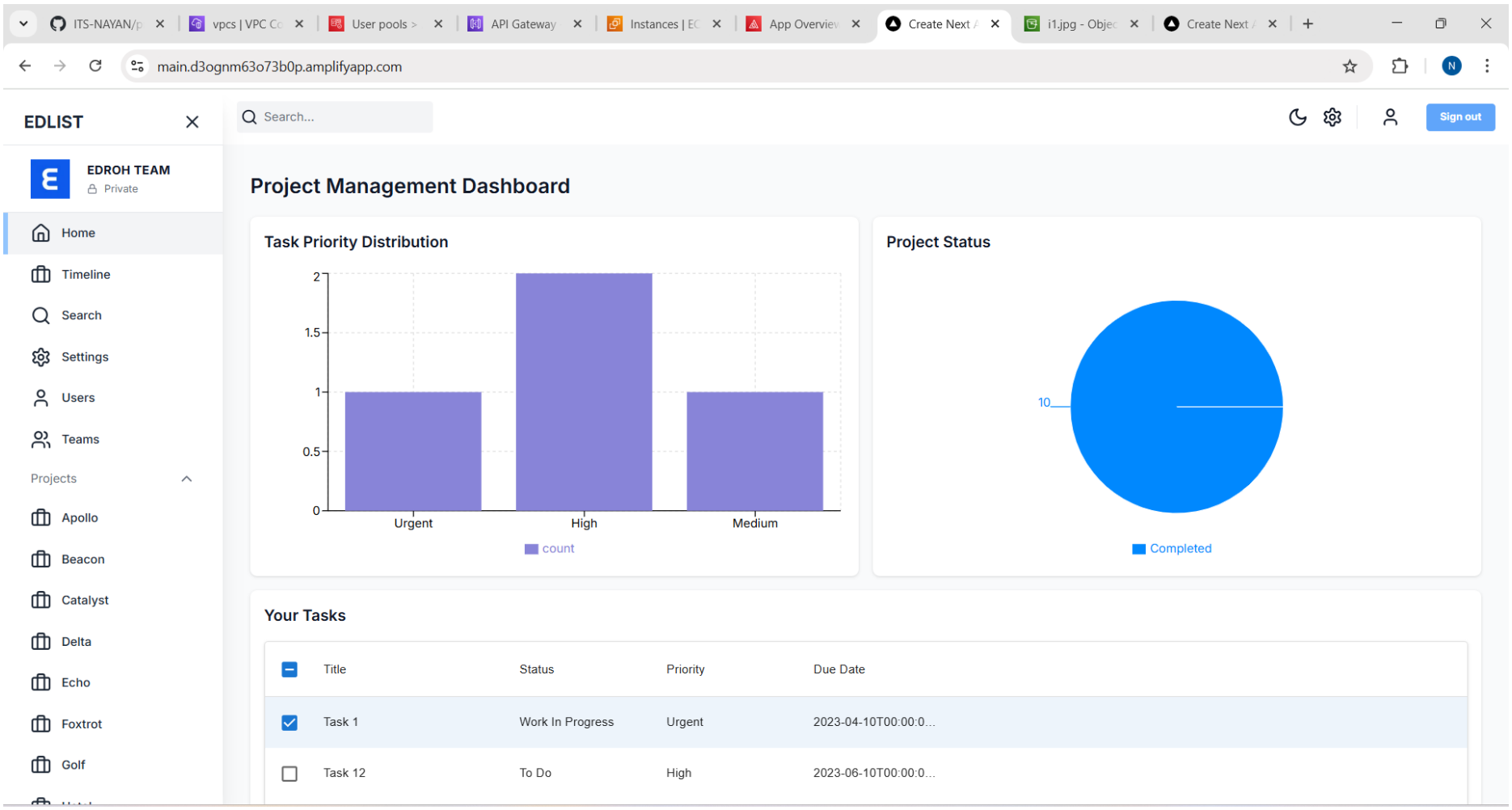

```
[
  {
    "id": 1,
    "name": "Apollo",
    "description": "A space exploration project.",
    "startDate": "2023-01-01T00:00:00.000Z",
    "endDate": "2023-12-31T00:00:00.000Z"
  },
  {
    "id": 2,
    "name": "Beacon",
    "description": "Developing advanced navigation systems.",
    "startDate": "2023-02-01T00:00:00.000Z",
    "endDate": "2023-10-15T00:00:00.000Z"
  },
  {
    "id": 3,
    "name": "Catalyst",
    "description": "A project to boost renewable energy use.",
    "startDate": "2023-03-05T00:00:00.000Z",
    "endDate": "2024-03-05T00:00:00.000Z"
  },
  {
    "id": 4,
    "name": "Delta",
    "description": "Delta project for new software development techniques.",
    "startDate": "2023-01-20T00:00:00.000Z",
    "endDate": "2023-09-20T00:00:00.000Z"
  },
  {
    "id": 5,
    "name": "Echo",
    "description": "Echo project focused on AI advancements.",
    "startDate": "2023-04-15T00:00:00.000Z",
    "endDate": "2023-11-30T00:00:00.000Z"
  },
  {
    "id": 6,
    "name": "Foxtrot",
    "description": "Exploring cutting-edge biotechnology.",
    "startDate": "2023-02-25T00:00:00.000Z",
    "endDate": "2023-08-25T00:00:00.000Z"
  },
  {
    "id": 7,
    "name": "Golf",
```

ITS Confusing first try with URL only till amazonaws.com, if not then amazonaws.com/prod.

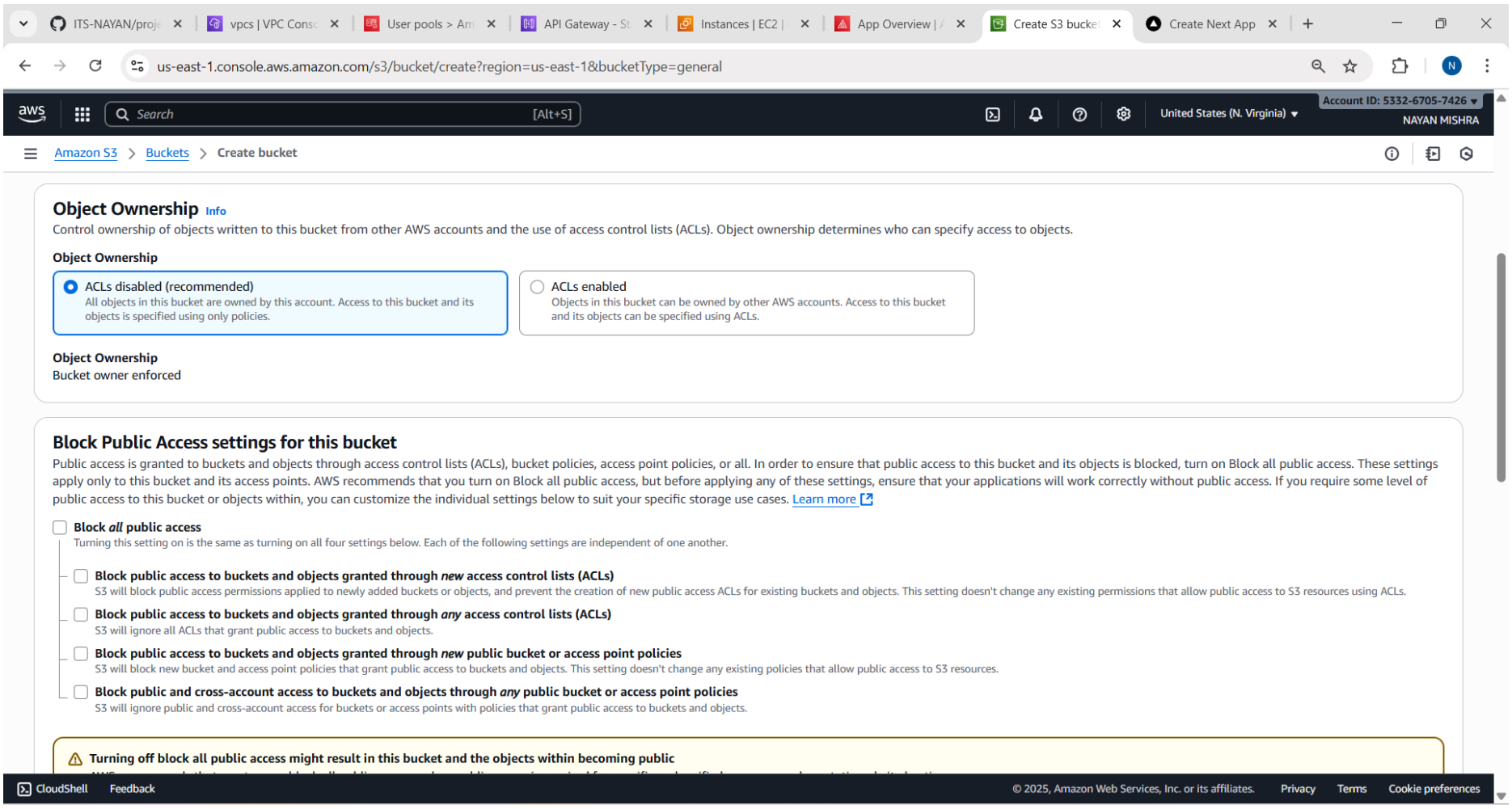


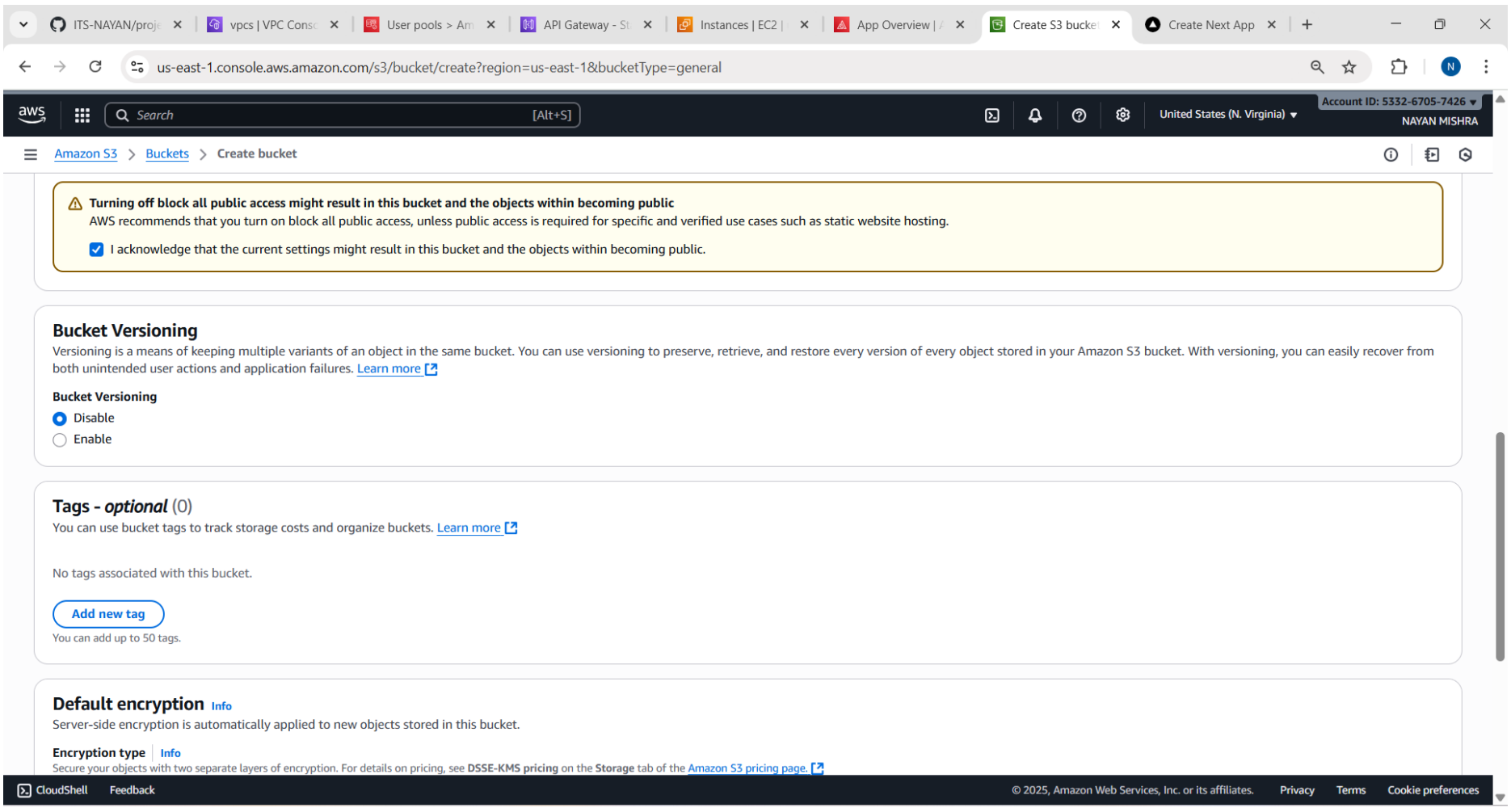


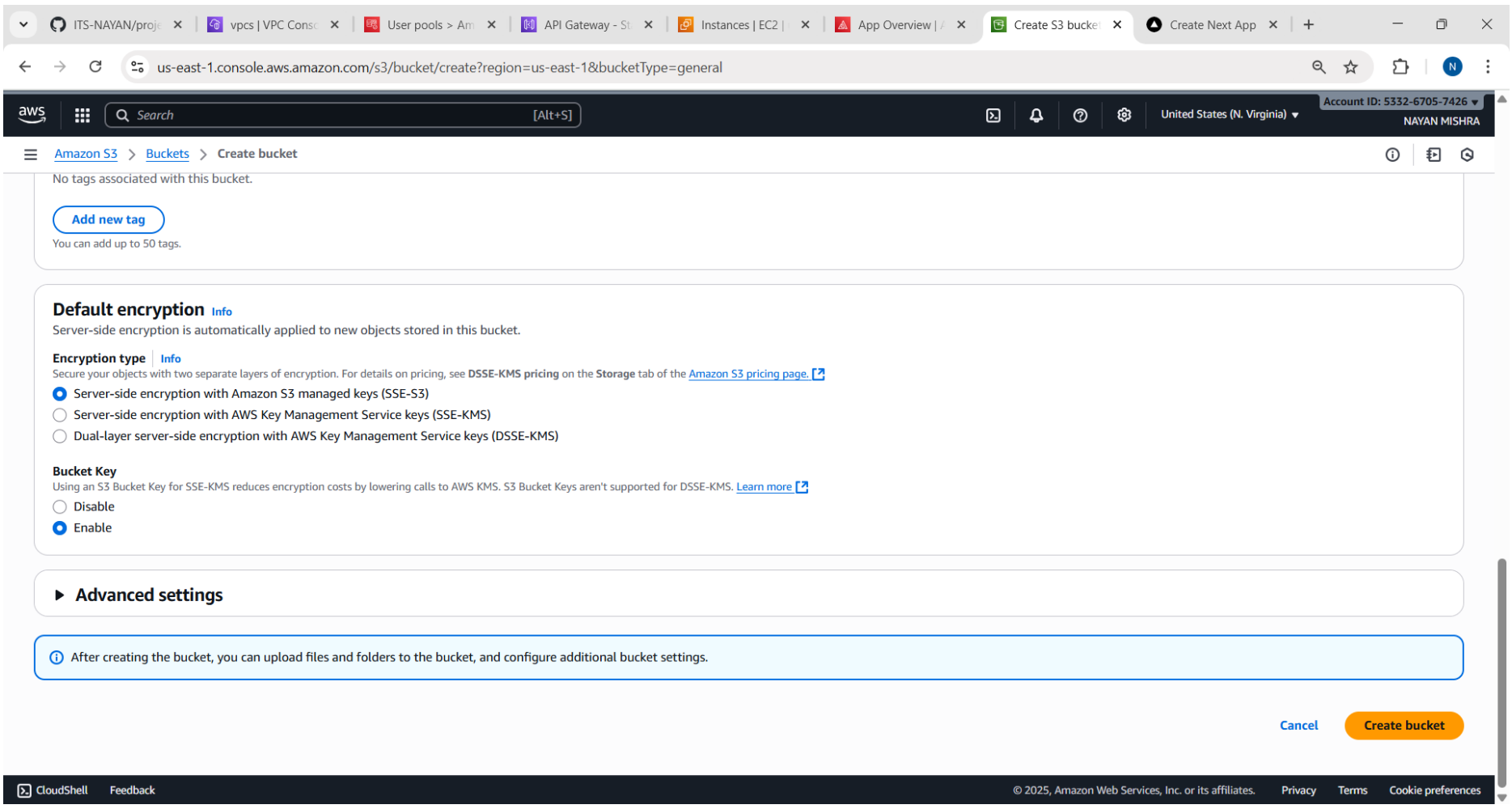
RESULT:



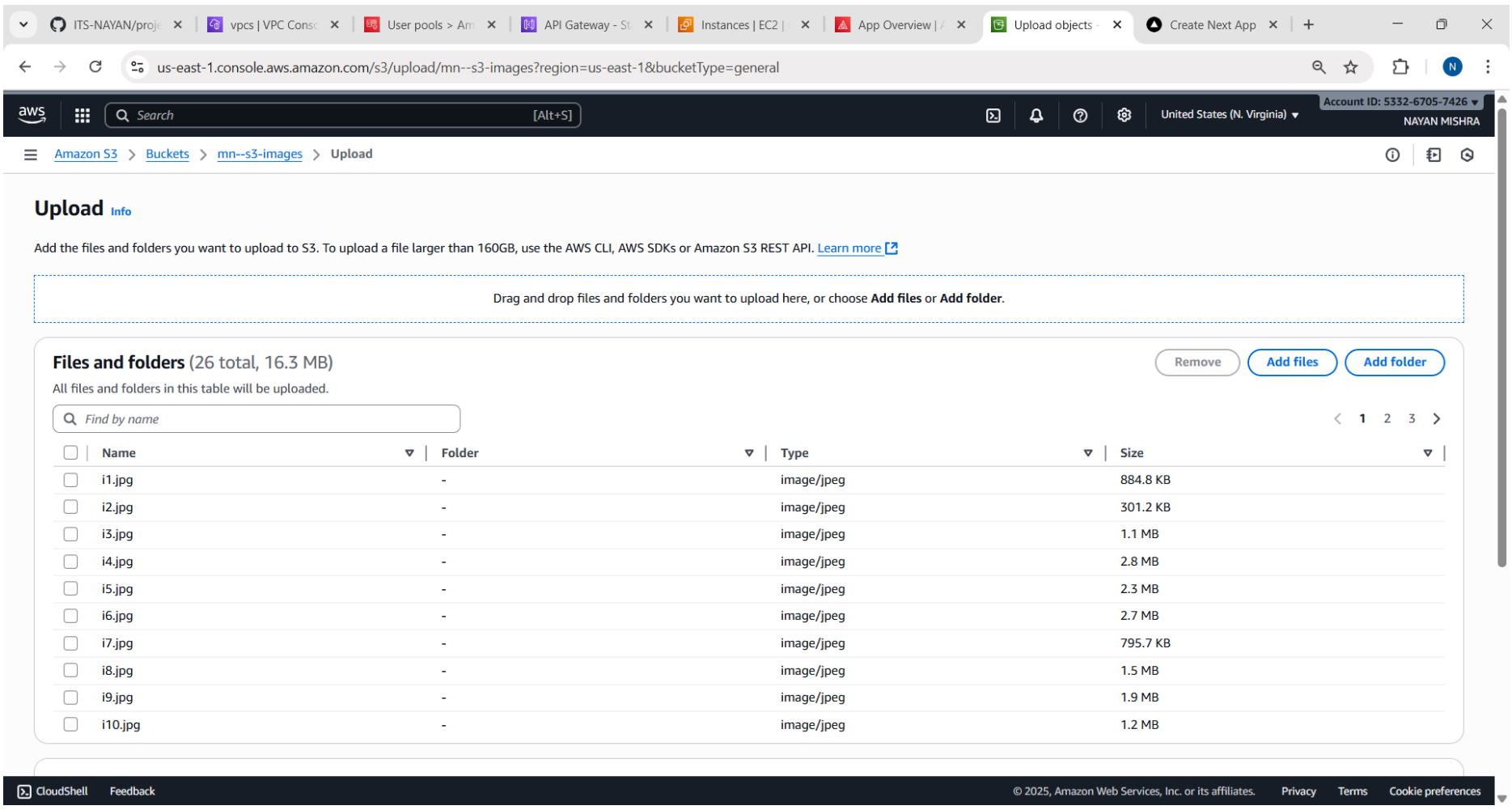
✅ SOME IMAGES MIGHT NOT LOADED SO FOR THAT FALLOW SOME STEPS:

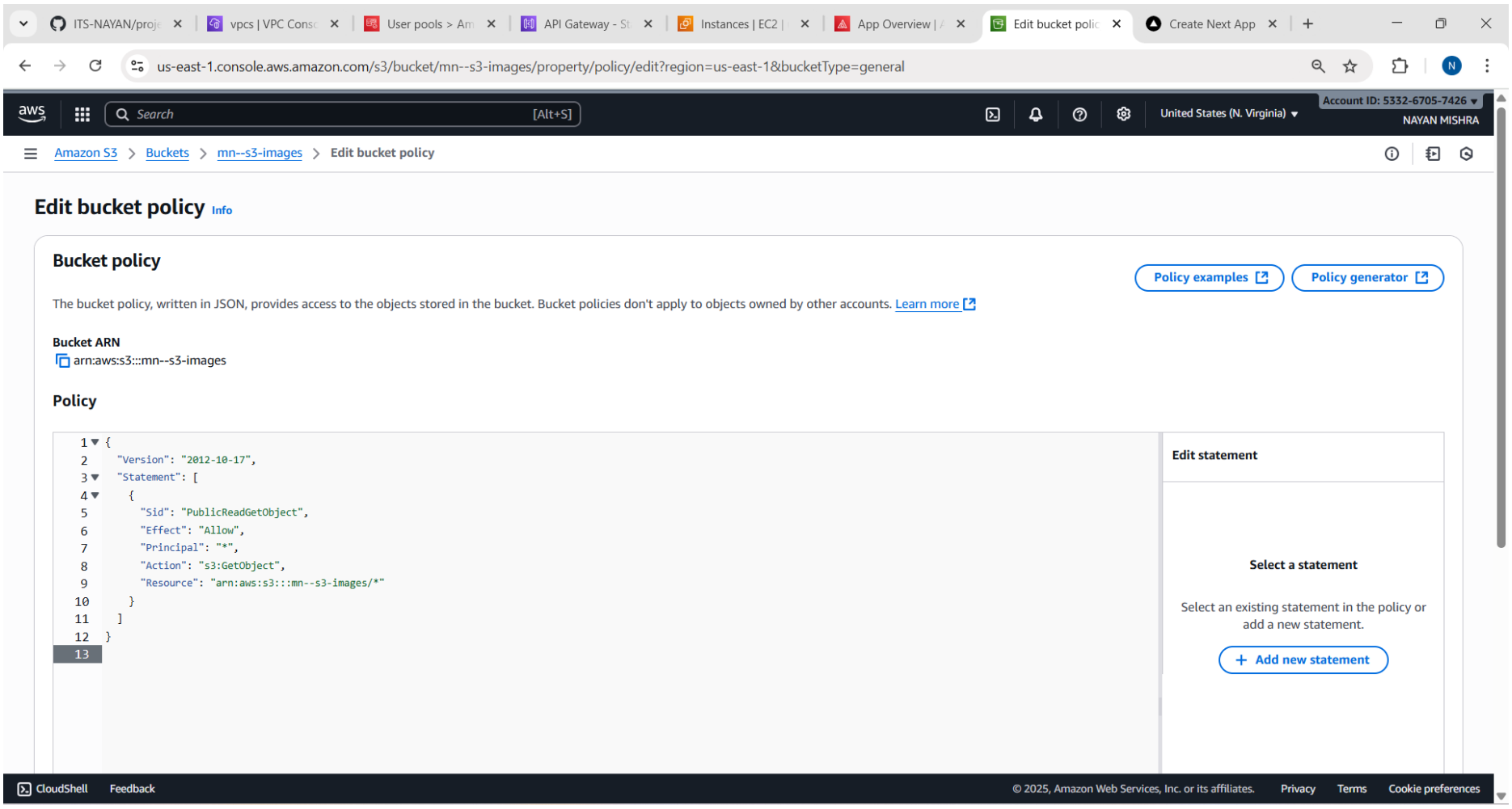

us-east-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Amazon S3 > Buckets > Create bucket
Object Ownership Info
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.
Object Ownership
ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.
Object Ownership
Bucket owner enforced
Block Public Access settings for this bucket
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
Block public access to buckets and objects granted through new access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
Block public access to buckets and objects granted through any access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.
Block public access to buckets and objects granted through new public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
Block public and cross-account access to buckets and objects through any public bucket or access point policies
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.
Turning off block all public access might result in this bucket and the objects within becoming public
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-east-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general
Search [Alt+S]
United States (N. Virginia)
Account ID: 5332-6705-7426
NAYAN MISHRA
Amazon S3 > Buckets > Create bucket
Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.
I acknowledge that the current settings might result in this bucket and the objects within becoming public.
Bucket Versioning
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more
Bucket Versioning
Disable
Enable
Tags - optional (0)
You can use bucket tags to track storage costs and organize buckets. Learn more
No tags associated with this bucket.
Add new tag
You can add up to 50 tags.
Default encryption Info
Server-side encryption is automatically applied to new objects stored in this bucket.
Encryption type Info
Secure your objects with two separate layers of encryption. For details on pricing, see DSSE-KMS pricing on the Storage tab of the Amazon S3 pricing page.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

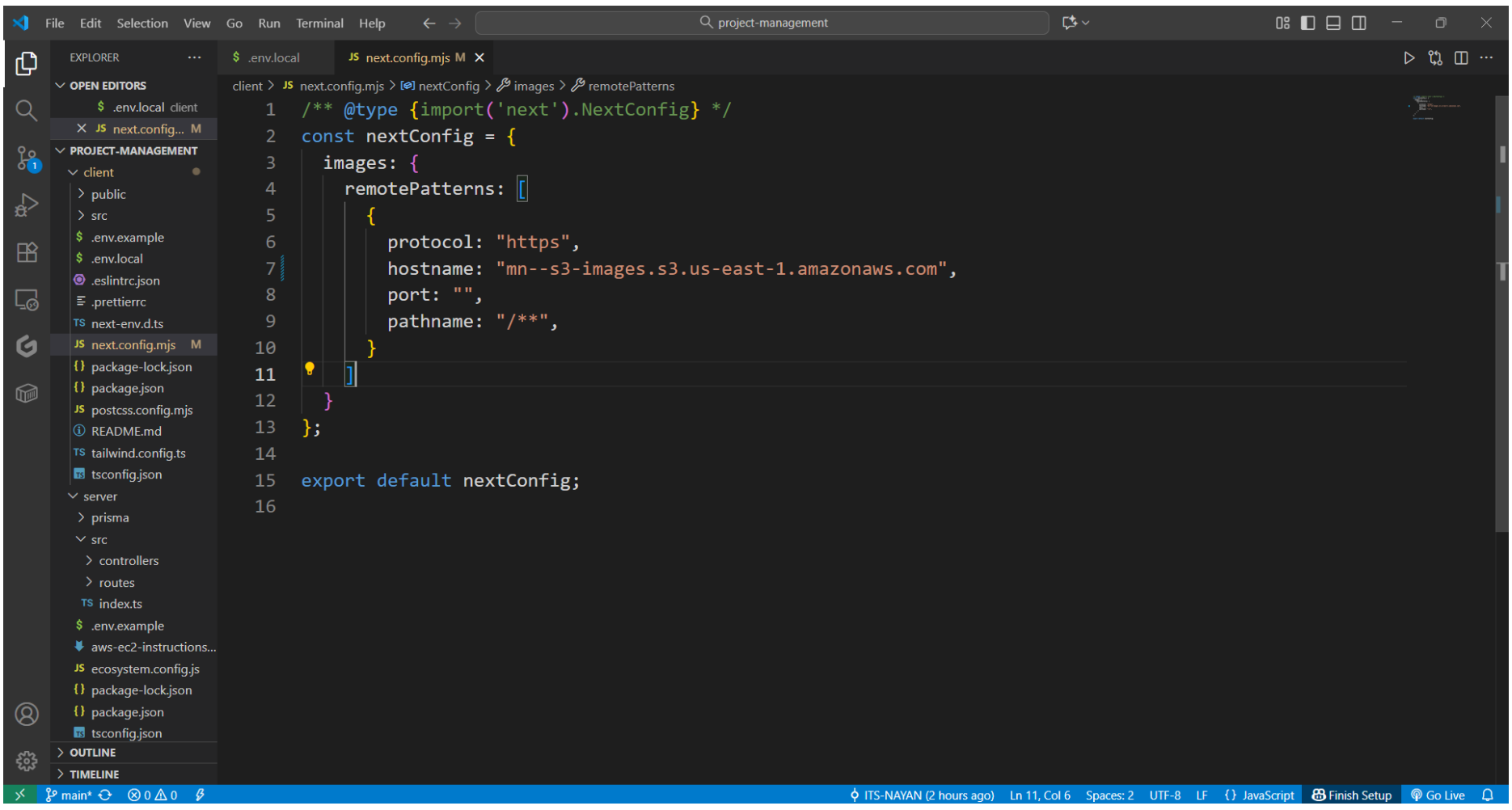


THIS IMAGES ARE IN PUBLIC IN SAME REPO

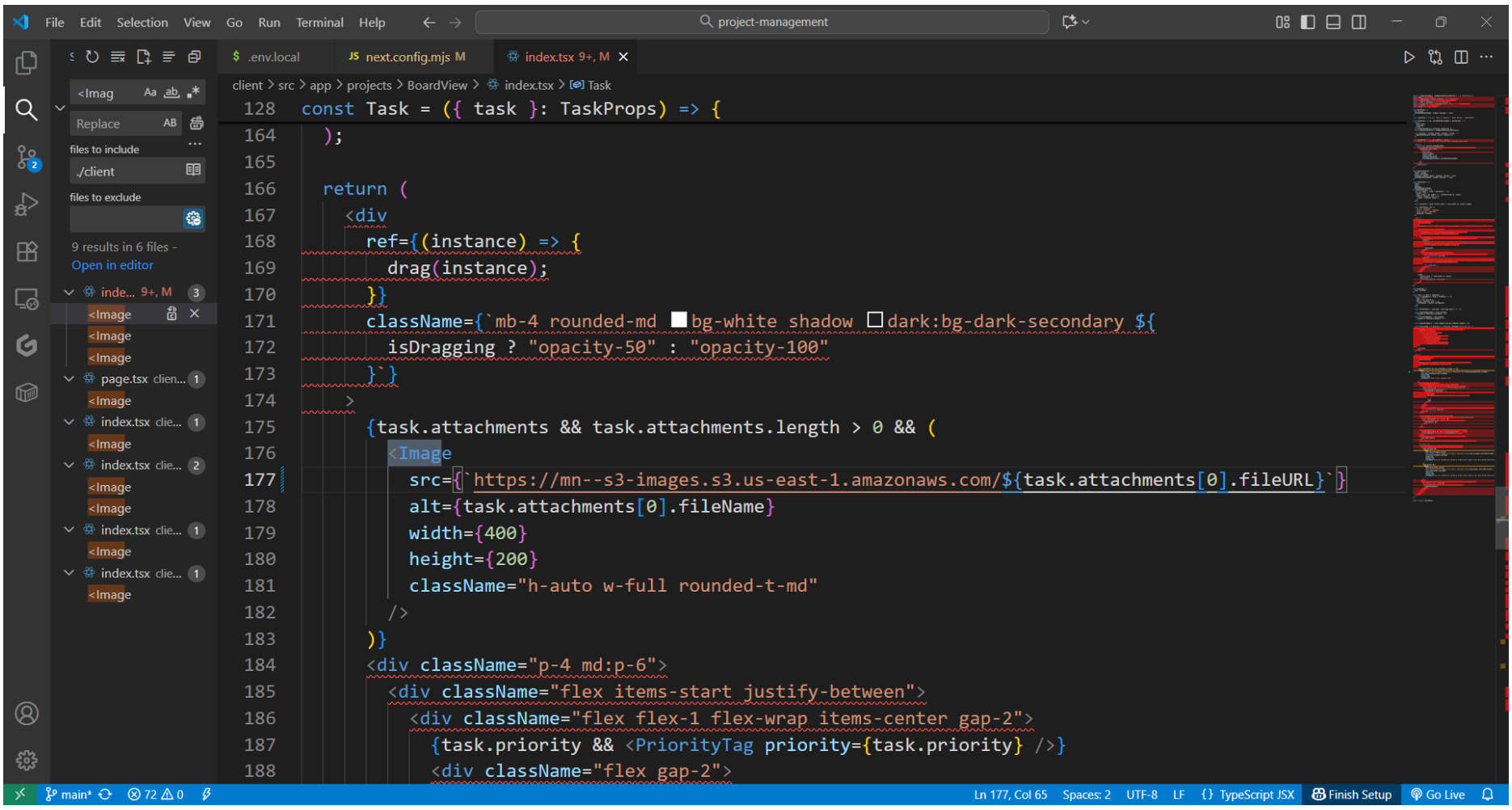


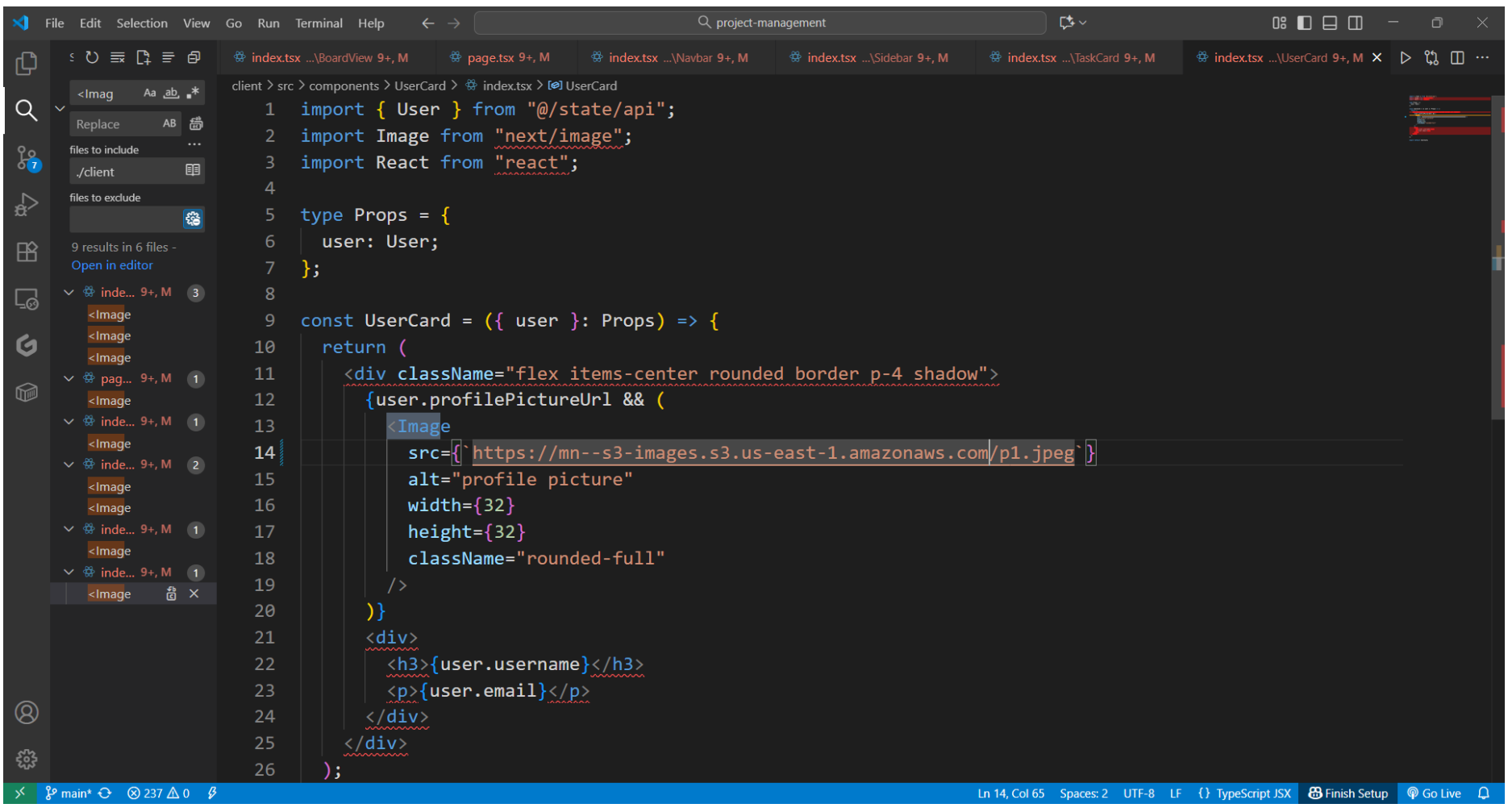


ALSO DO SOME CHANGES IN CODE REPO

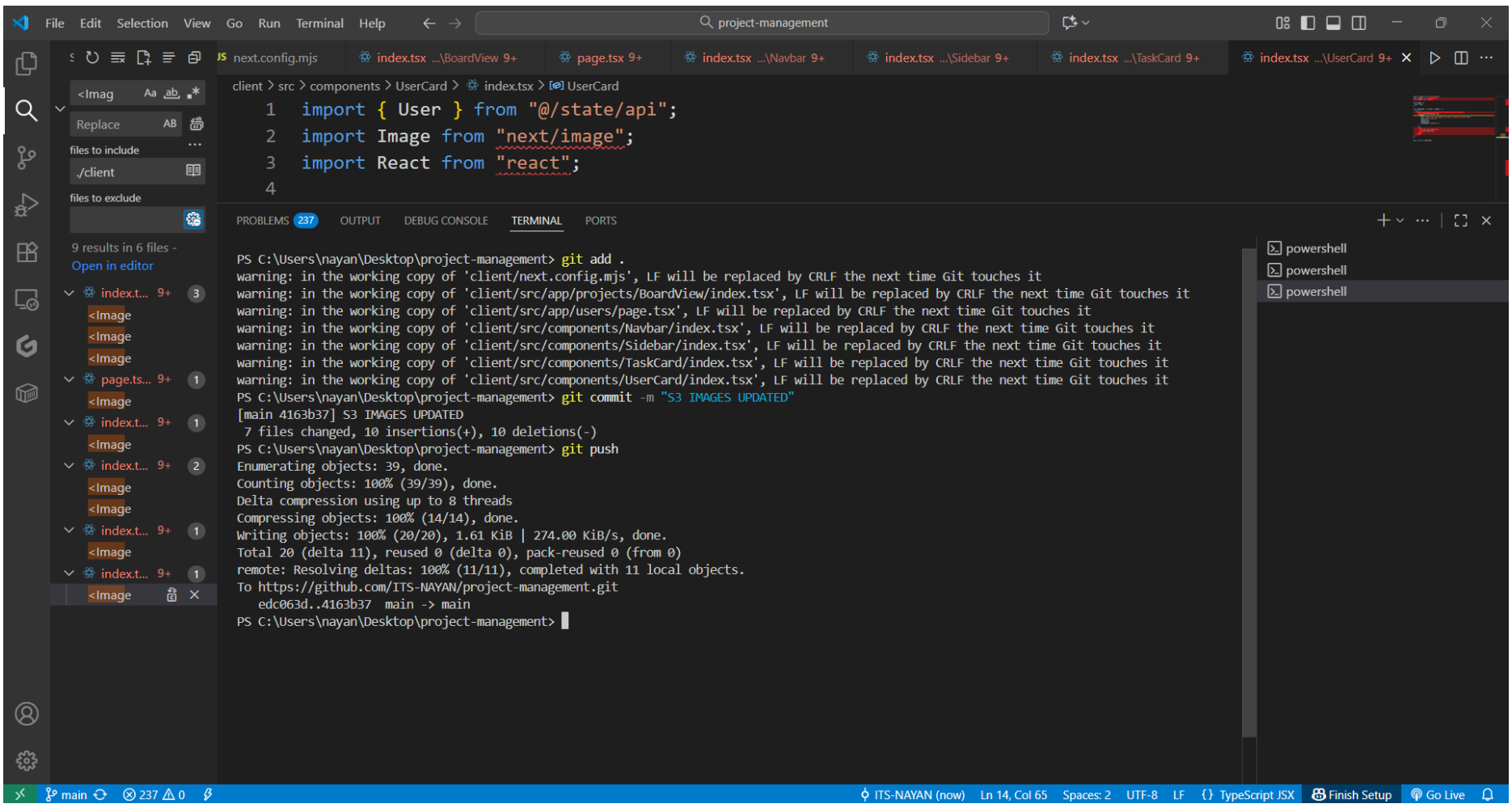


LEFT CLICK ON FOLDER IN VS CODE AND SELECT FIND IN FOLDER AND REPLACE ALL WITH YOUR S3 URL

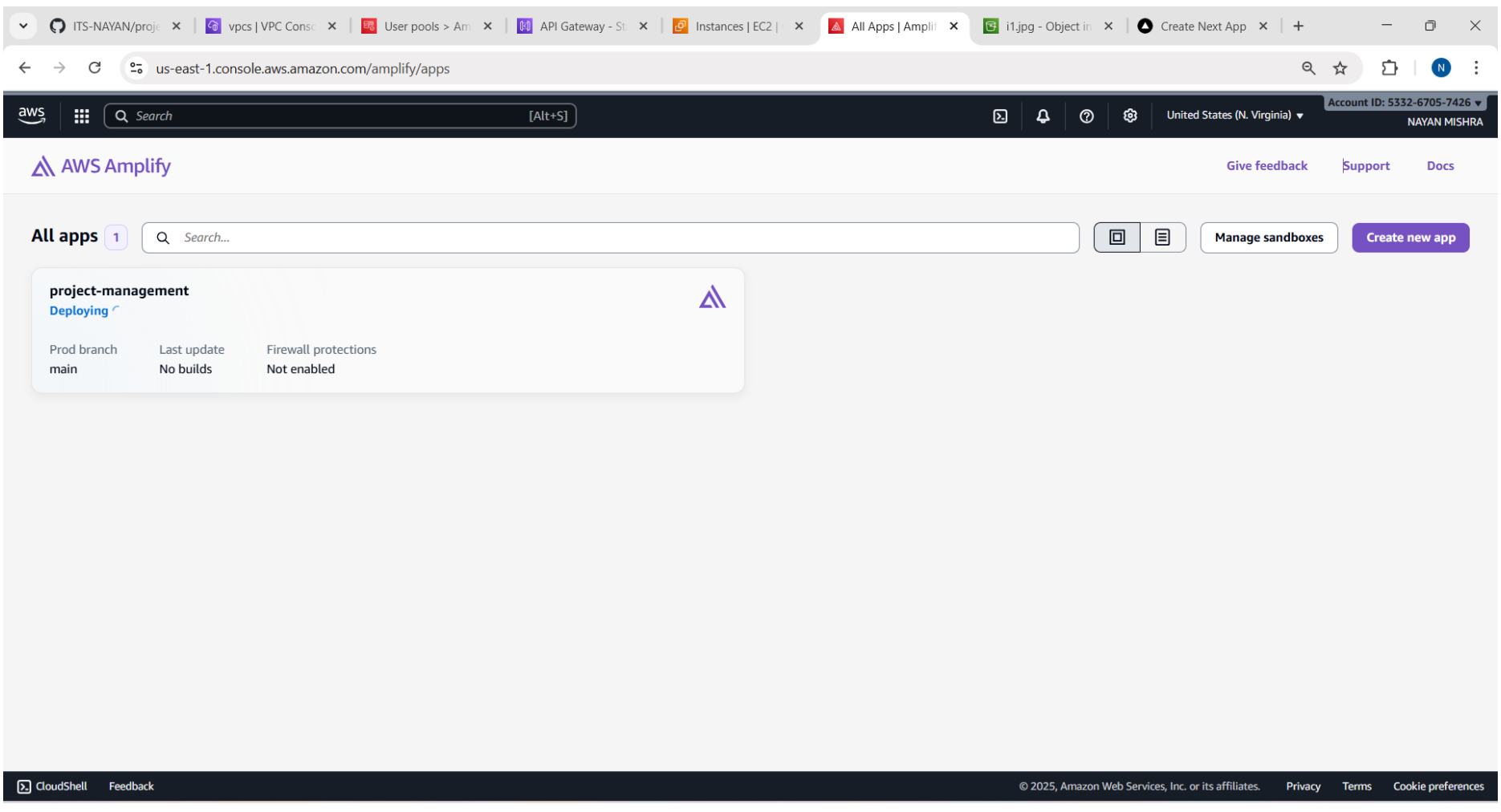




COMMIT ALL CHANGES TO GIT HUB



WAIT FOR SOME TIME



WANTS TO ADD DATABASE IN NEXT PART