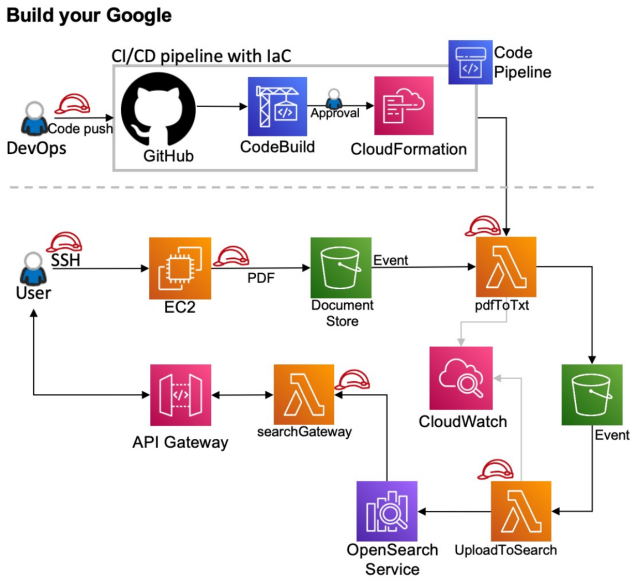
Patrick Jones, 8/4/2025..
PGPCCAUG4_PATRICK_JONES_PROJECT2

Deploying a search engine using AWS Manage Services

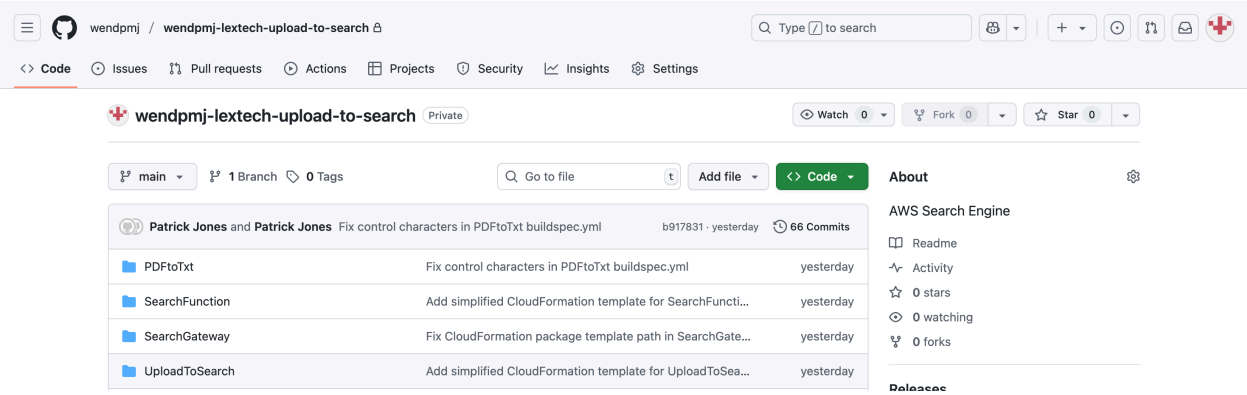
greatlearning

Architecture



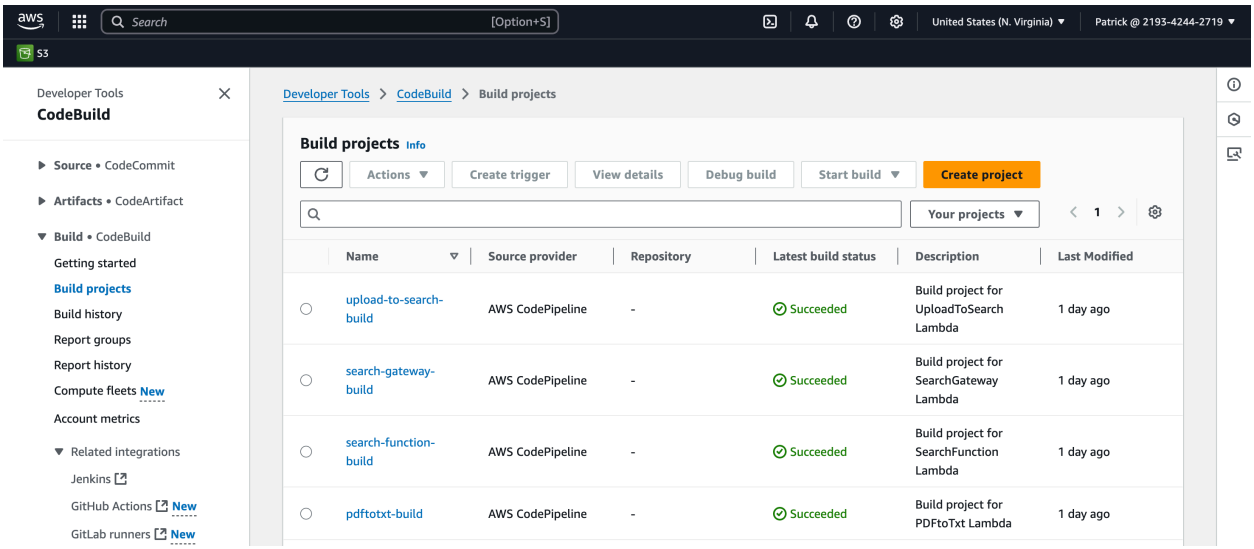
Github

- setup private repository
- created classic token for aws pipeline authentication
- upload directories from local host



Codebuild – for each function

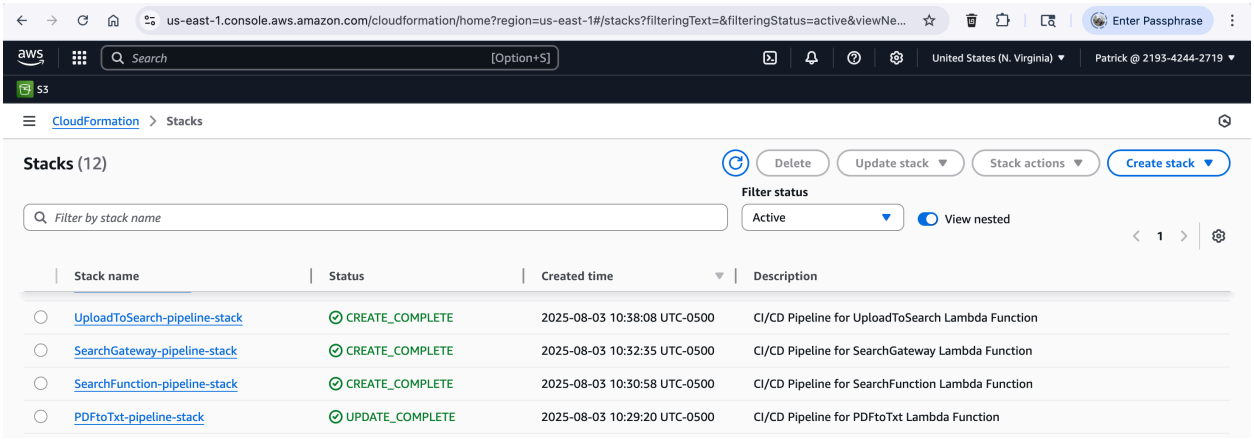
- Automates the building and deployment of the four Lambda functions.
- CodeBuild executes the buildspec.yml for each specific Lambda
- Buildspec.yml runs aws cloudformation stack.



Cloud Formation stacks

- Manages the API Gateway infrastructure for the project.
- Handles deployments related to the UploadToSearch function pipeline.
- Manages the deployment pipeline for the Search Gateway
- Manages the pipeline for deployments of the Search Function.

- Manages resources around the PDF to Text conversion function.

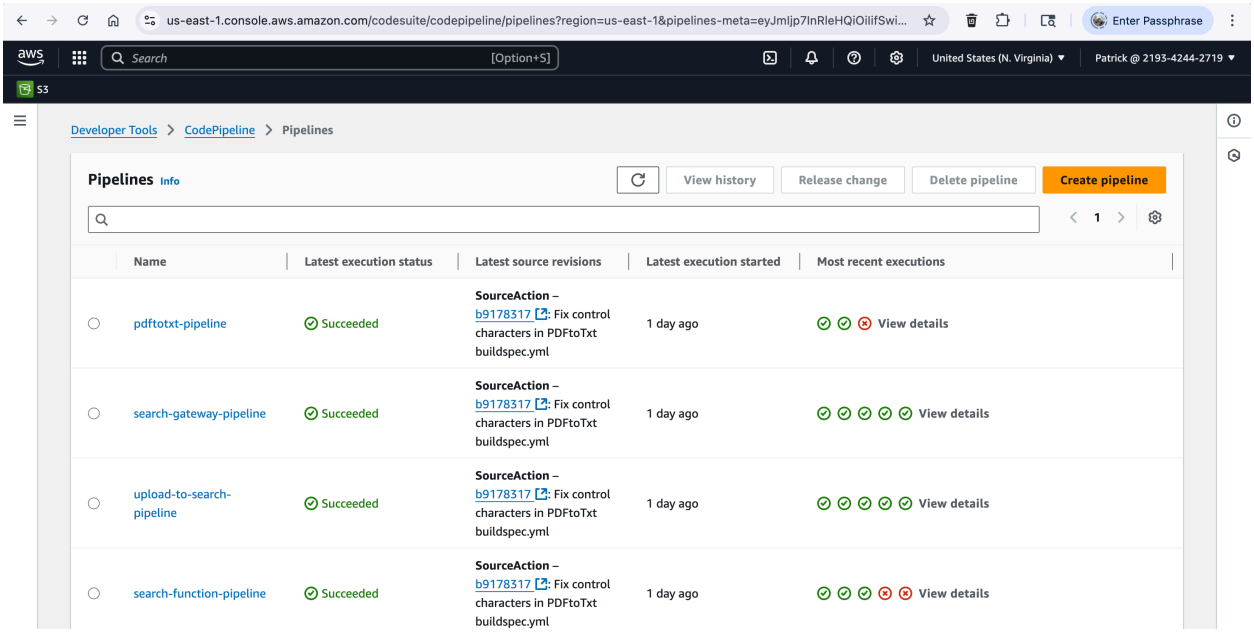


Pipelines

- CodePipeline directs the process by triggering the CodeBuild project
- CodePipeline triggers when source code changes
- S3 bucket (lextech-artifacts-bucket-219342442719)

  //Store source code from GitHub

  //Pass artifacts between Source → Build stages

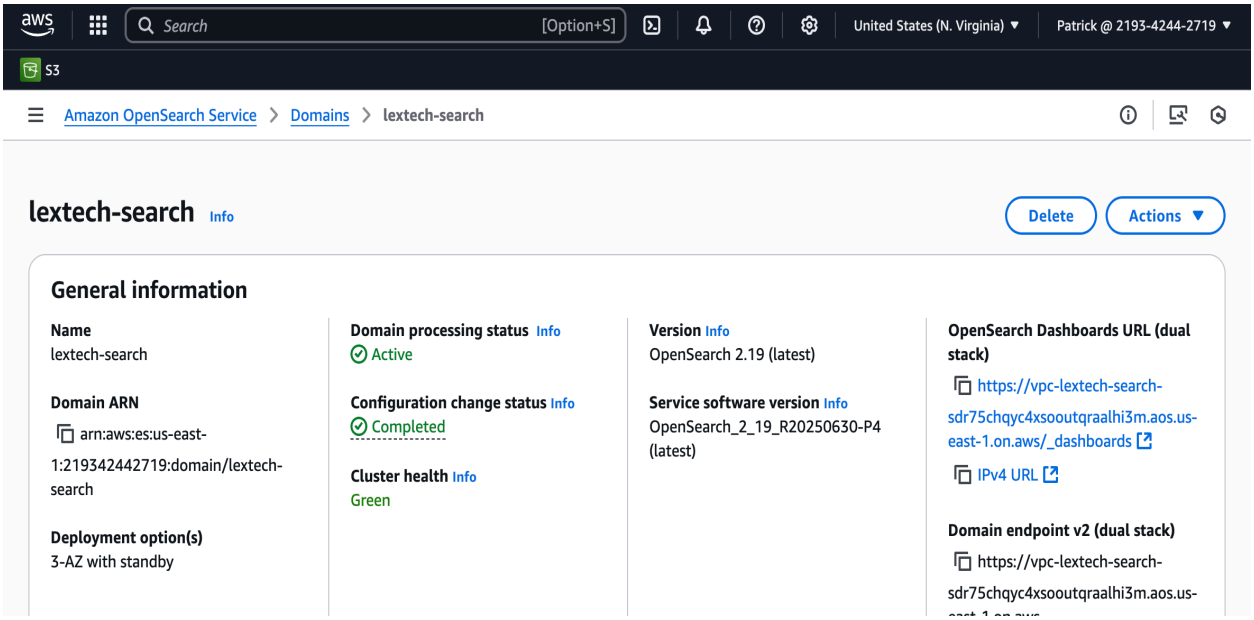


OpenSearch Domain: lextech-search

https://vpc-lextech-search-sdr75chqyc4xsooutqraalhi3m.aos.us-east-1.on.aws/_dashboards

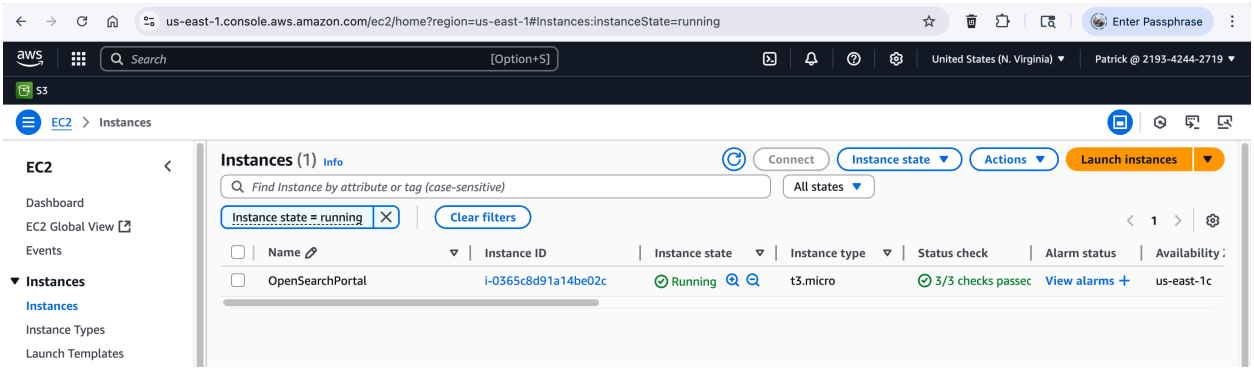
- Stores processed PDF content that was extracted and converted to text by the PDFtoTxt

Lambda



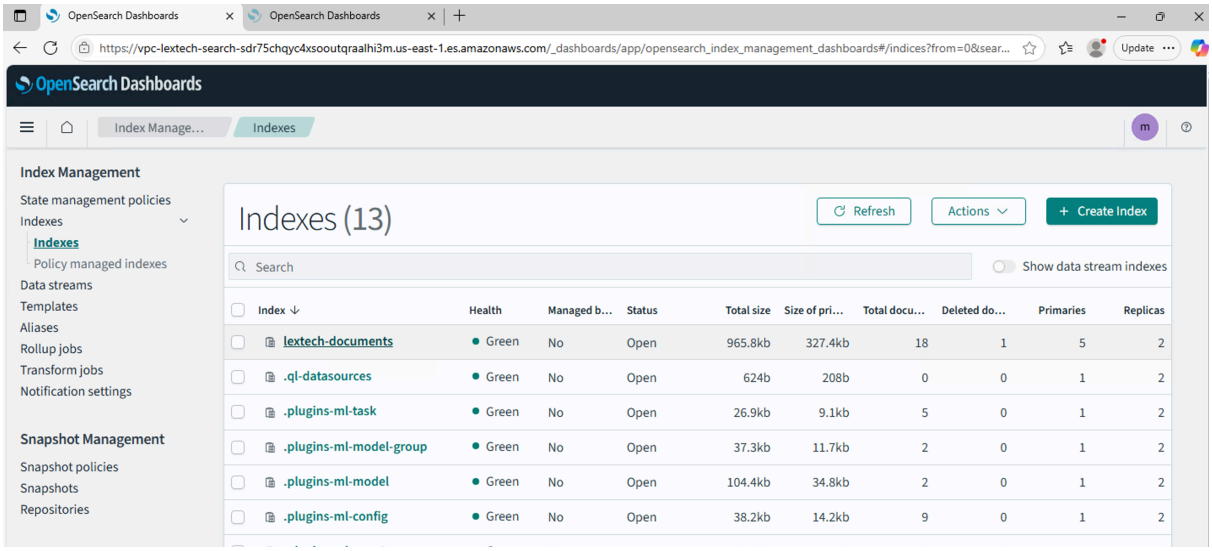
EC2 instance

- Configure instance, add to same VPC as OpenSearch dashboard.
- Open inbound rule for rdp port 3389 to remote into EC2c instance for OpenSearch dashboard access. Restricted 3389 access to my public ip 108.91.0.80/32



OpenSearch dashboard

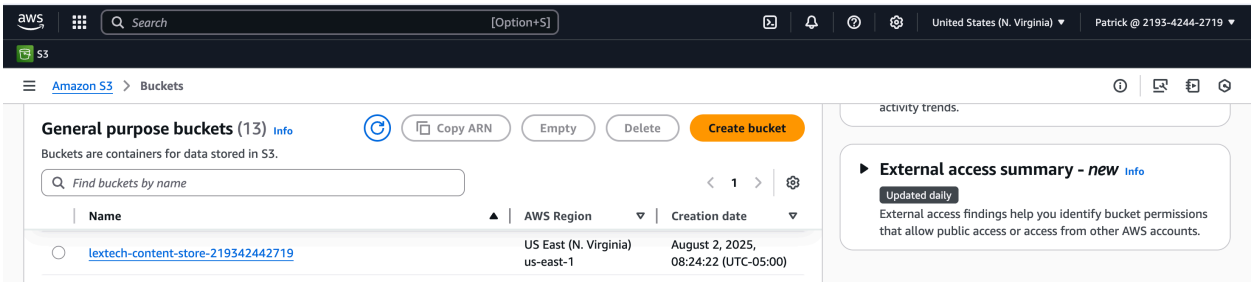
- Once accessible through EC2 jump
- Create lextech-documents index to receive PdfToTxt documents for search



Create S3 bucket

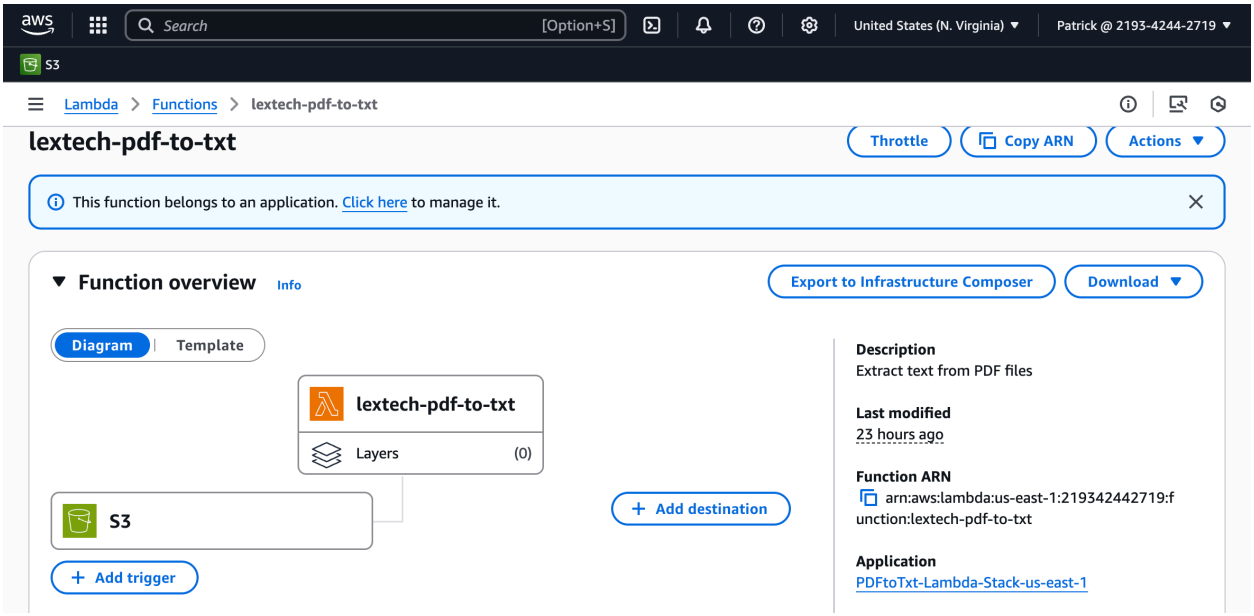
Purpose: Process pdf's for Lamba functions.

- S3 bucket (lextech-content-store-219342442719) //pdf upload bucket



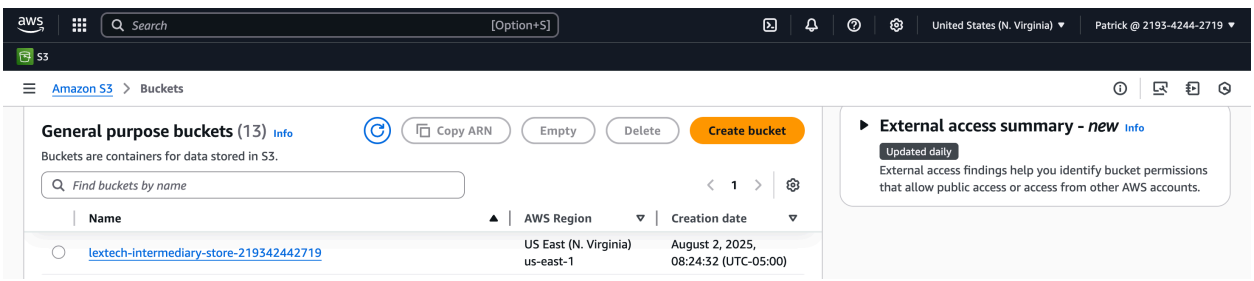
Create PdfToTxt Lambda function (lextech-pdf-txt)

- Automatically triggers when a PDF file is uploaded to the S3 content bucket (lextech-content-store-219342442719)
- Uses S3 event notifications to detect new .pdf files
- Extracts ALL text content from every page in the PDF
- Saves the extracted text to the intermediary S3 bucket (lextech-intermediary-store-219342442719)
- CloudWatch logs created automatically



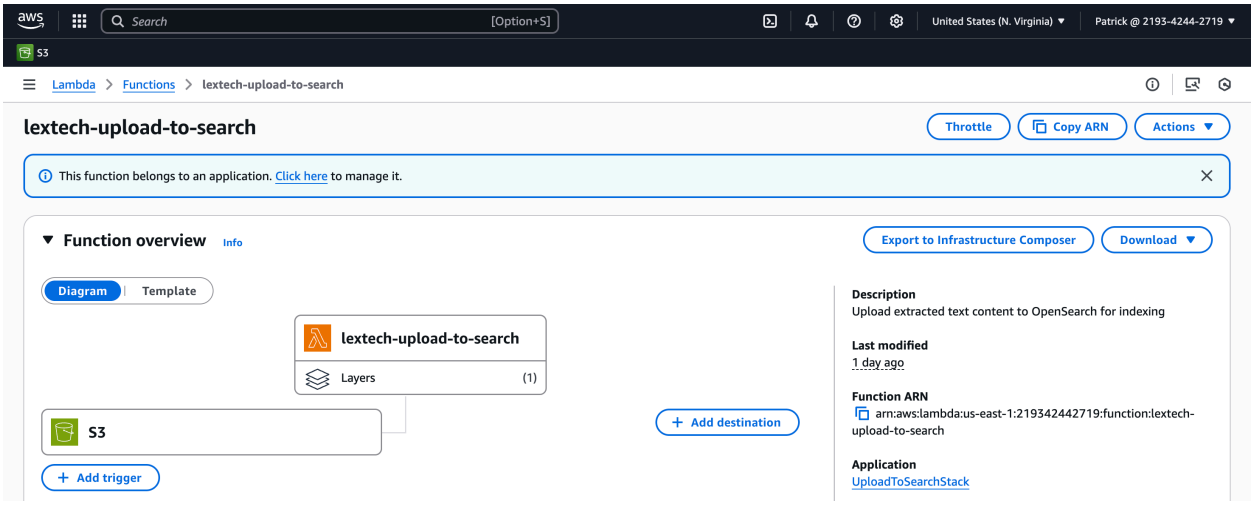
Create S3 bucket (lextech-intermediary-store-219342442719) //pdf upload bucket

- Automatically triggers when a .txt file is uploaded to the intermediary S3 bucket (lextech-intermediary-store-219342442719)
- Uses S3 event notifications to detect new .txt files created by the PDFtoTxt function
- Downloads the text file from the intermediary S3 bucket
- Uploads the document to OpenSearch using REST API



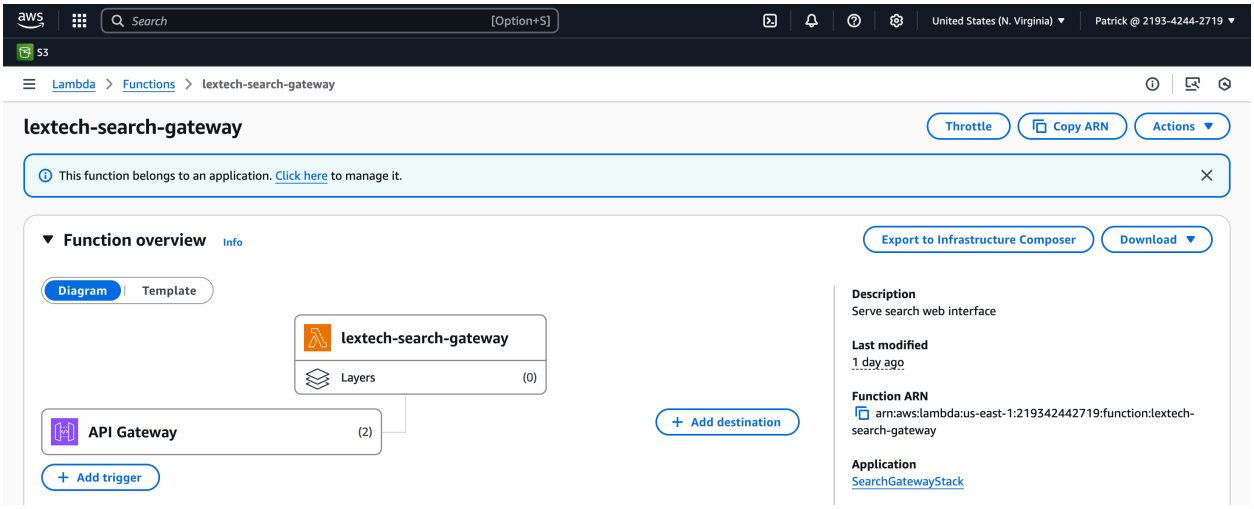
UploadToSearch Function (lextech-upload-to-search)

- Second step in the document processing pipeline, takes the text files created by PDFtoTxt and makes them searchable by indexing them in OpenSearch.
- Converts text files into searchable documents in OpenSearch.
- Automatically triggers when a .txt file is uploaded to the intermediary S3 bucket (lextech-intermediary-store-219342442719).
- Uses S3 event notifications to detect new .txt files created by the PDFtoTxt function.



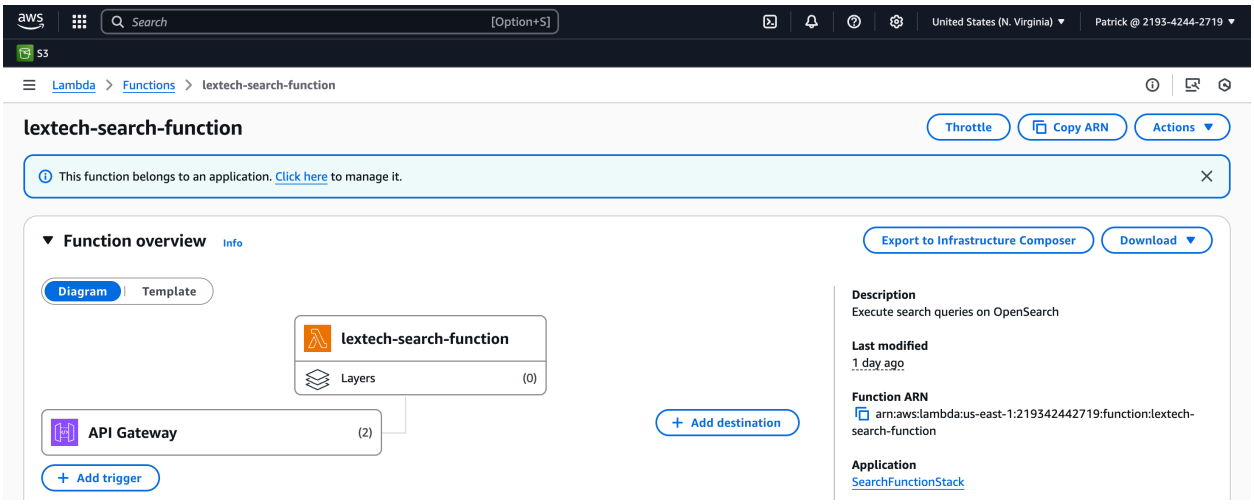
SearchGateway Function (lextech-search-gateway)

- Serves HTML content when users visit the main API Gateway URL
- Acts as a web server delivering a complete HTML page with CSS and JavaScript
- No database or file storage, generates the interface dynamically



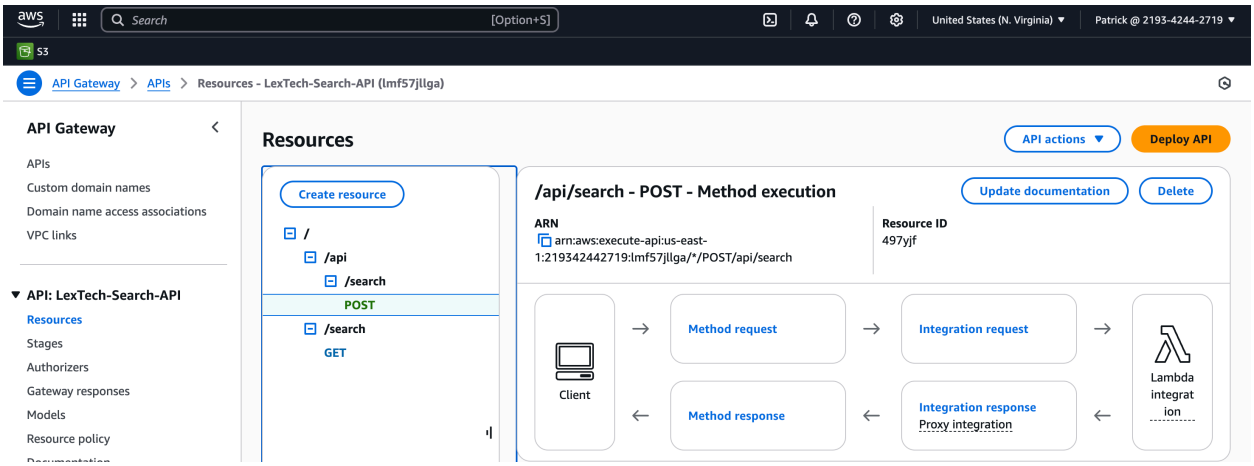
LextechSearchFunction (lextech-search-function)

- Receives search requests from the SearchGateway web interface via API Gateway.



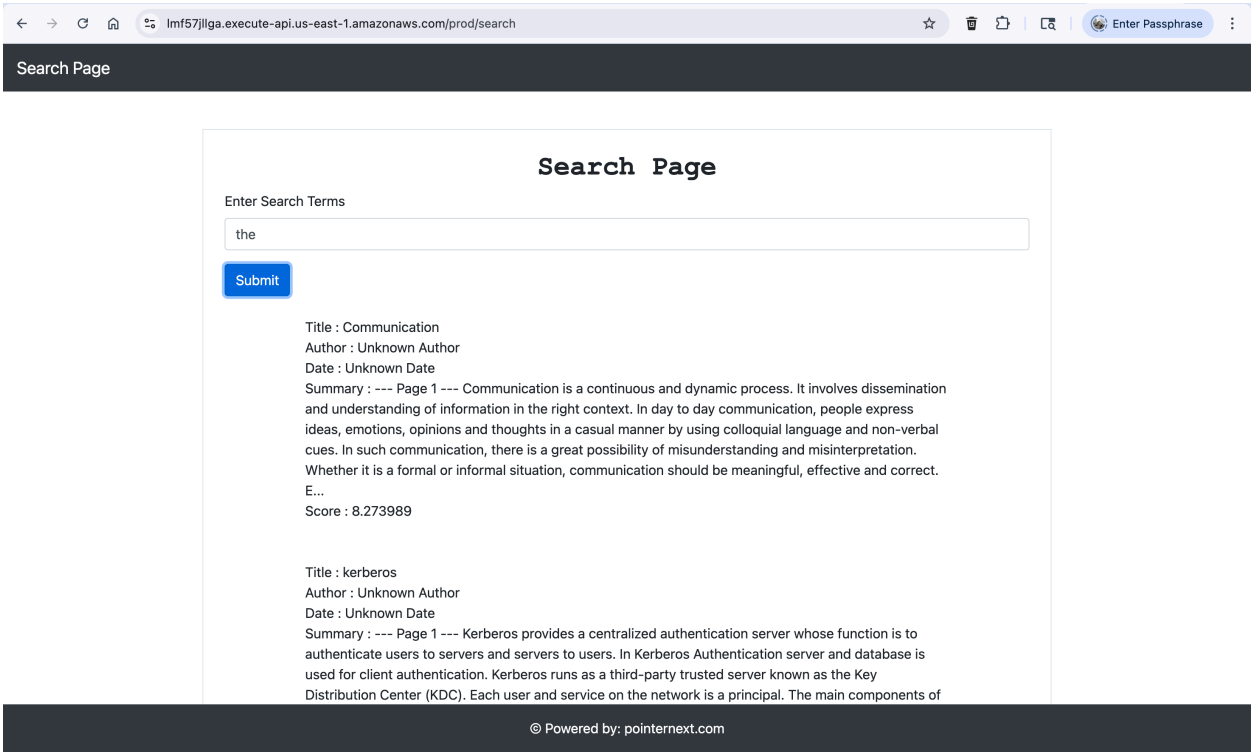
API Gateway

- Search that receives requests with search queries and returns results from the OpenSearch backend.
- The API Gateway acts as a proxy to invoke the appropriate Lambda functions (SearchGateway for web interface, SearchFunction for search queries)



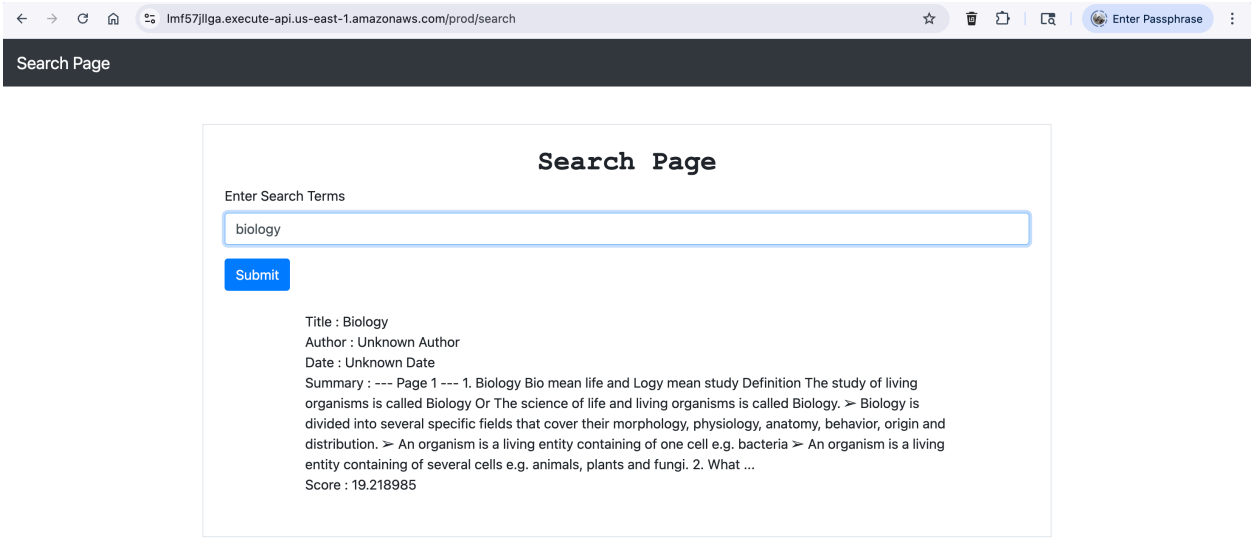
Example 1

Search ‘the’ and multiple documents returned successfully that had the word ‘the’...



Example 2

Search word ‘biology’ and document returned successfully that had the word ‘biology’...



Lessons and Observations

This search engine project was very hard and complex for me. I worked on this project daily for almost 2 weeks. I learned a lot along the way through trial and error to the point of almost giving up. I would make progress then hit a brick wall, then get past the brick wall to run into it all over again consistently. Through perseverance, I was able to finally see the vision of this project and work methodically through each step to completion. Looking back now, I have great deal of satisfaction for completing this project.

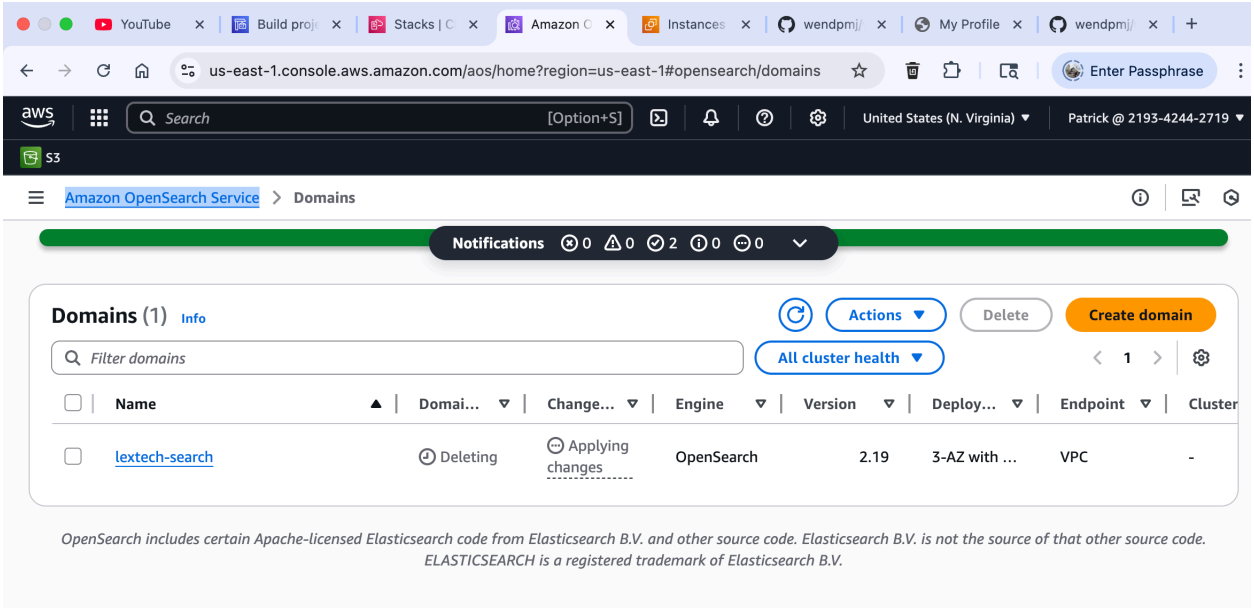
A few complex issues throughout this project included:

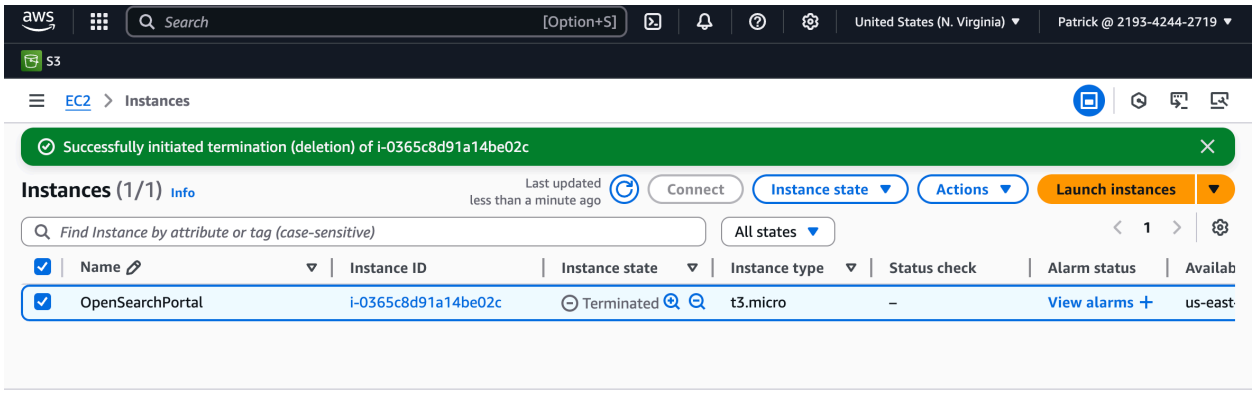
- Misconfigured Lambda environment variables and VPC settings that broke OpenSearch.
- CI/CD pipeline setup was very challenging, failure after failure, mostly permissions and VPC related
- OpenSearch initial setup was rather simple, but the search, indexing and data upload was challenging. Main issue was the local roles security permissions, and mapping (mapped users) additional AWS roles to resolve permissions/access issues.
- Experienced multiple failures with pipeline implementation, trial and error took a great deal of time.

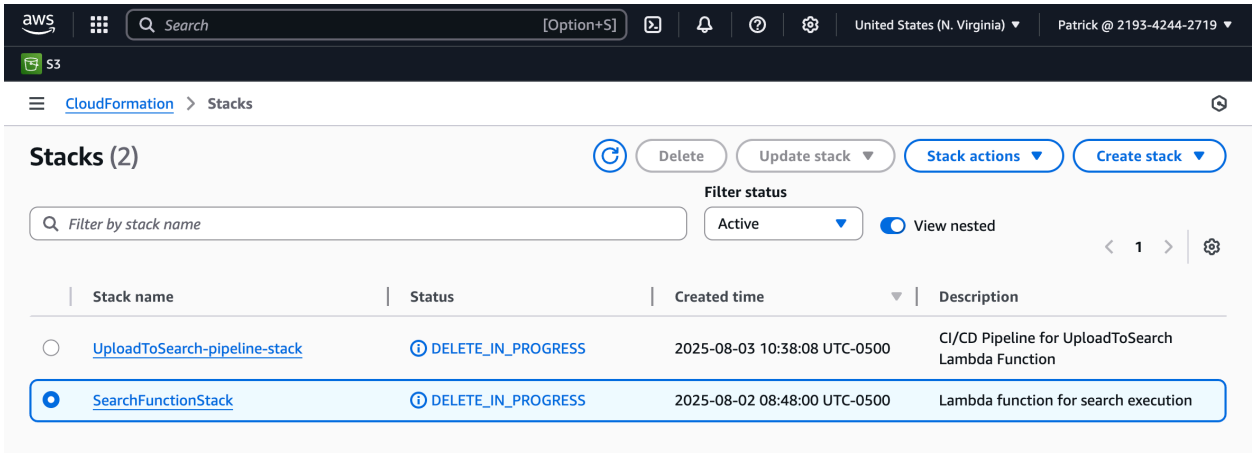
- VPC configuration preventing Lambda-to-OpenSearch communication - Configured security groups and subnets to allow Lambda functions to reach the VPC-based OpenSearch.
- Incomplete CI/CD pipeline deployment, "PLACEHOLDER_TOKEN" instead of valid GitHub personal access tokens was the issue, preventing source code retrieval.
- PDF upload not appearing in search results - S3 upload → PDF processing → text extraction → OpenSearch indexing.
- Lambda function deployment inconsistencies, resolved through persistent trial and error.

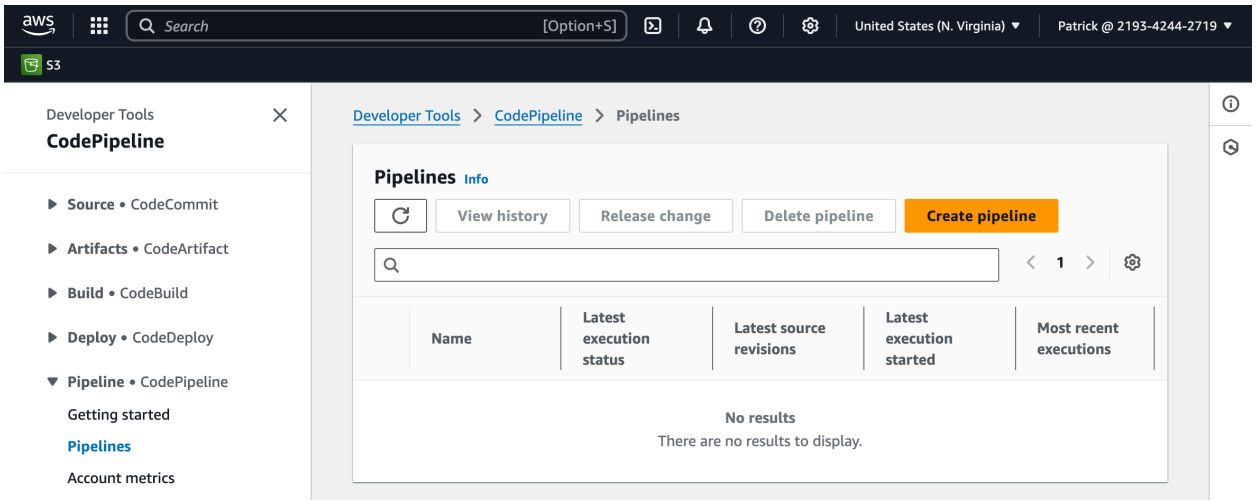
Patrick M. Jones
8/4/2025

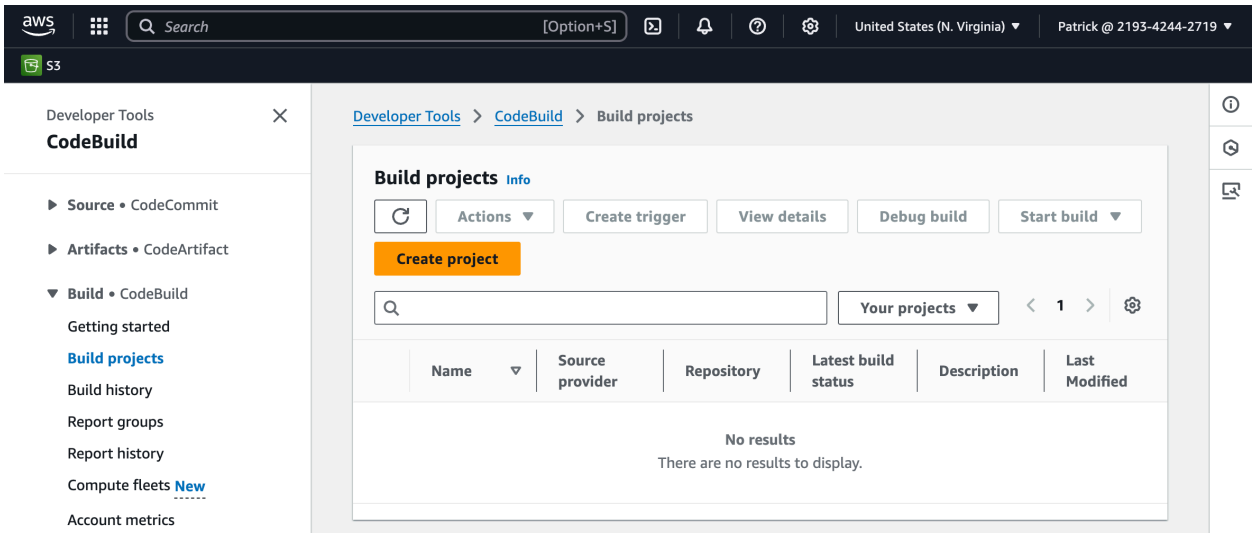
Project Cleanup

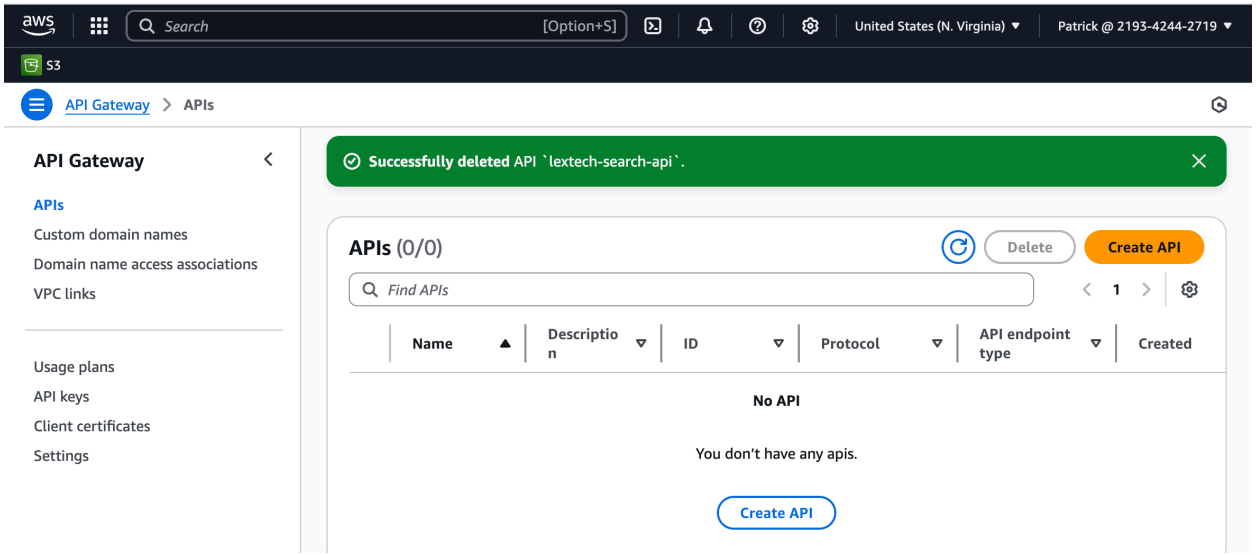


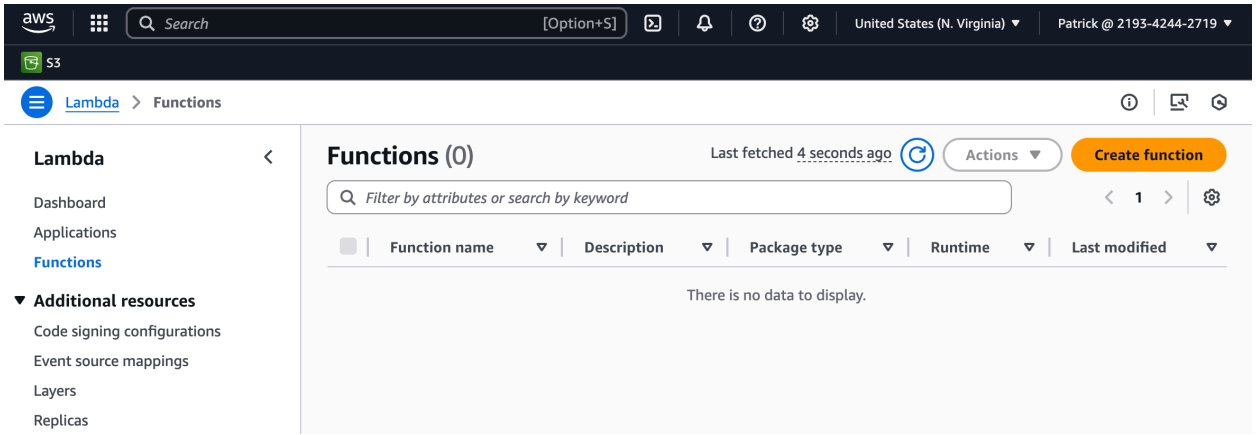

Search [Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
EC2 > Instances
Successfully initiated termination (deletion) of i-0365c8d91a14be02c
Instances (1/1) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
OpenSearchPortal
i-0365c8d91a14be02c
Terminated
t3.micro
–
View alarms
us-east

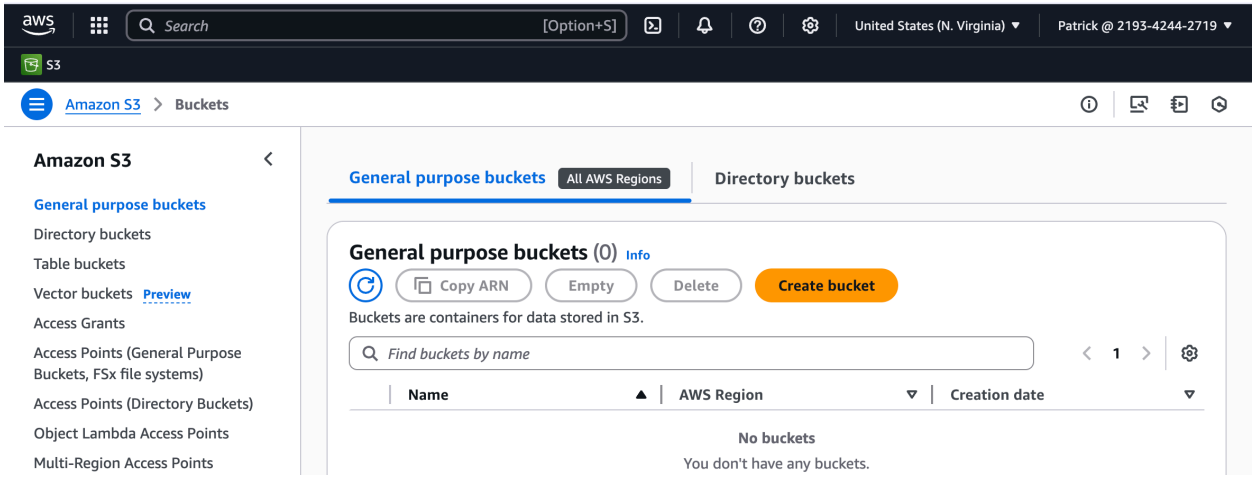

Search [Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
CloudFormation > Stacks
Stacks (2)
Delete
Update stack
Stack actions
Create stack
Filter by stack name
Filter status
Active
View nested
Stack name
Status
Created time
Description
UploadToSearch-pipeline-stack
DELETE_IN_PROGRESS
2025-08-03 10:38:08 UTC-0500
CI/CD Pipeline for UploadToSearch Lambda Function
SearchFunctionStack
DELETE_IN_PROGRESS
2025-08-02 08:48:00 UTC-0500
Lambda function for search execution


Search [Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
Developer Tools
CodePipeline
Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Account metrics
Developer Tools > CodePipeline > Pipelines
Pipelines Info
View history
Release change
Delete pipeline
Create pipeline
Name
Latest execution status
Latest source revisions
Latest execution started
Most recent executions
No results
There are no results to display.


Search
[Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
Developer Tools
CodeBuild
Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Getting started
Build projects
Build history
Report groups
Report history
Compute fleets New
Account metrics
Developer Tools > CodeBuild > Build projects
Build projects Info
Actions
Create trigger
View details
Debug build
Start build
Create project
Your projects
Name
Source provider
Repository
Latest build status
Description
Last Modified
No results
There are no results to display.


Search
[Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
API Gateway > APIs
API Gateway
APIs
Custom domain names
Domain name access associations
VPC links
Usage plans
API keys
Client certificates
Settings
Successfully deleted API `lextech-search-api`.
APIs (0/0)
Delete
Create API
Find APIs
Name
Description
ID
Protocol
API endpoint type
Created
No API
You don't have any apis.
Create API


Search
[Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
Lambda > Functions
Lambda
Dashboard
Applications
Functions
Additional resources
Code signing configurations
Event source mappings
Layers
Replicas
Functions (0)
Last fetched 4 seconds ago
Actions
Create function
Filter by attributes or search by keyword
Function name
Description
Package type
Runtime
Last modified
There is no data to display.


Search
[Option+S]
United States (N. Virginia)
Patrick @ 2193-4244-2719
S3
Amazon S3
Buckets
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Vector buckets
Preview
Access Grants
Access Points (General Purpose Buckets, FSx file systems)
Access Points (Directory Buckets)
Object Lambda Access Points
Multi-Region Access Points
General purpose buckets
All AWS Regions
Directory buckets
General purpose buckets (0)
Info
Copy ARN
Empty
Delete
Create bucket
Buckets are containers for data stored in S3.
Find buckets by name
1
Name
AWS Region
Creation date
No buckets
You don't have any buckets.