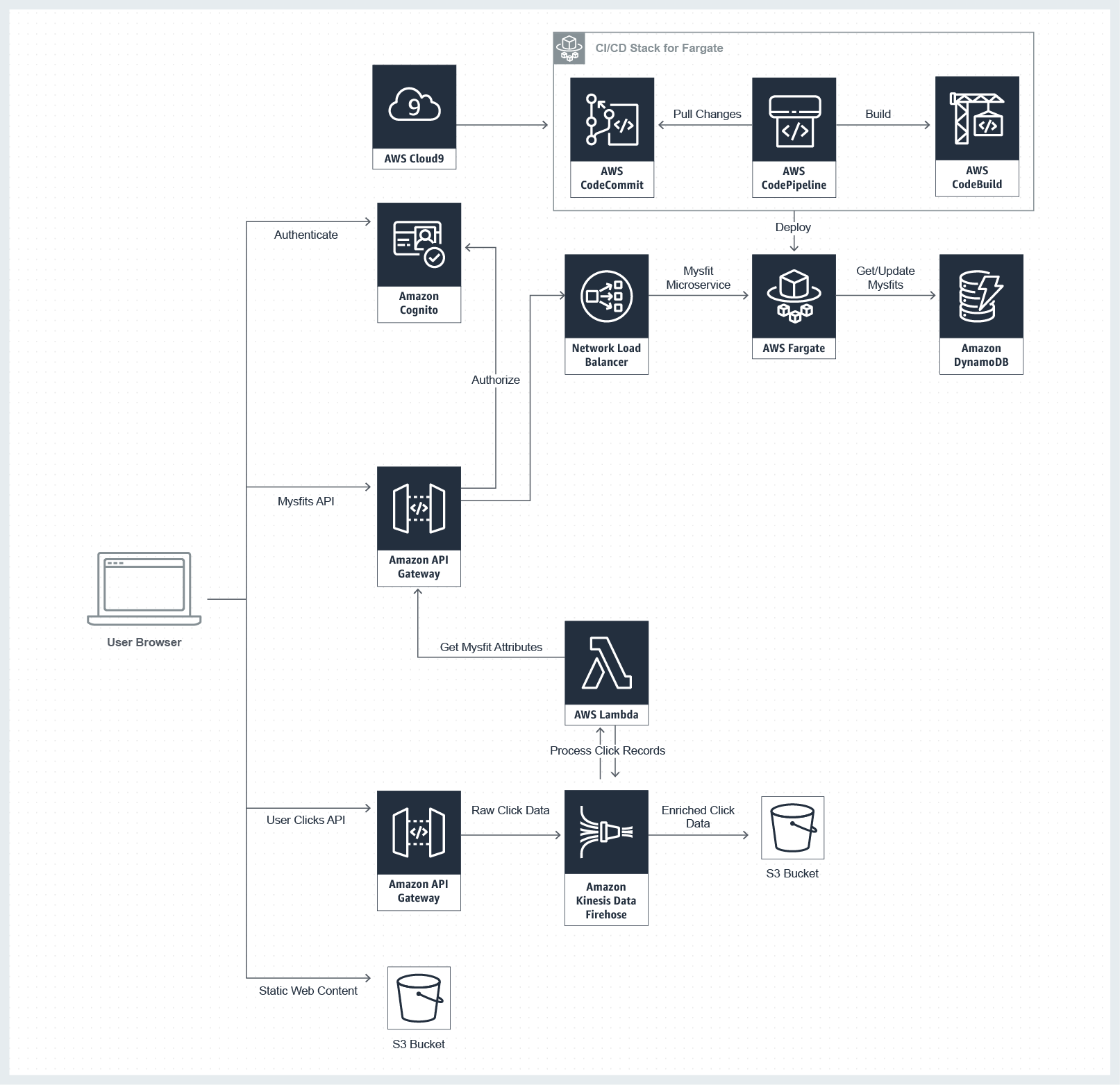
**Abstract**

Modern applications isolate business logic, optimize reuse and automates infrastructure maintenance tasks. In our project we build a website called Mythical Mysfits that lets visitors to adopt fantasy creature mysfits as pets. This adoption centre finds home for the abandoned and often mythical creatures in our community. The website lets you register with your email id and enables log in at the time of adoption. All mysfits come from different species and charcateristics. The website enables the functionality to “like” your favourite mysfit and reserve your chosen mysfit for adoption. It also allows you to gather insights about user behaviour for future analysis. Here we learn to host the web application on a front-end web server, connect it to a backend database, set up user authentication and will also be able to collect and analyse user behaviour.

**Application Architecture**

The snapshot of the application architecture given below provides a structural representation of services that make up Mythical Mysfits and how these services interact with each other.



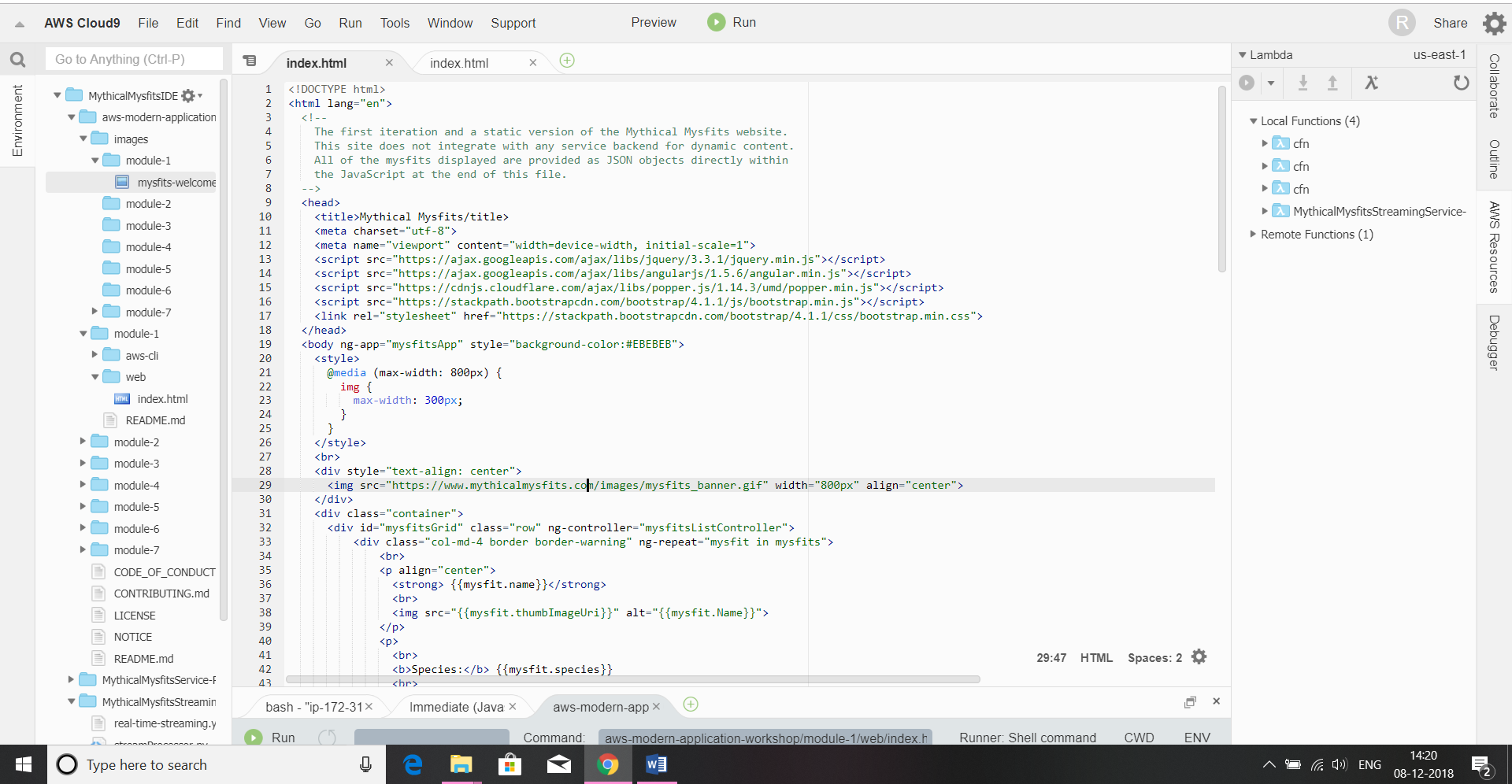
**Implementation**

The implementation of Mythical Mysfits is done using the modules:

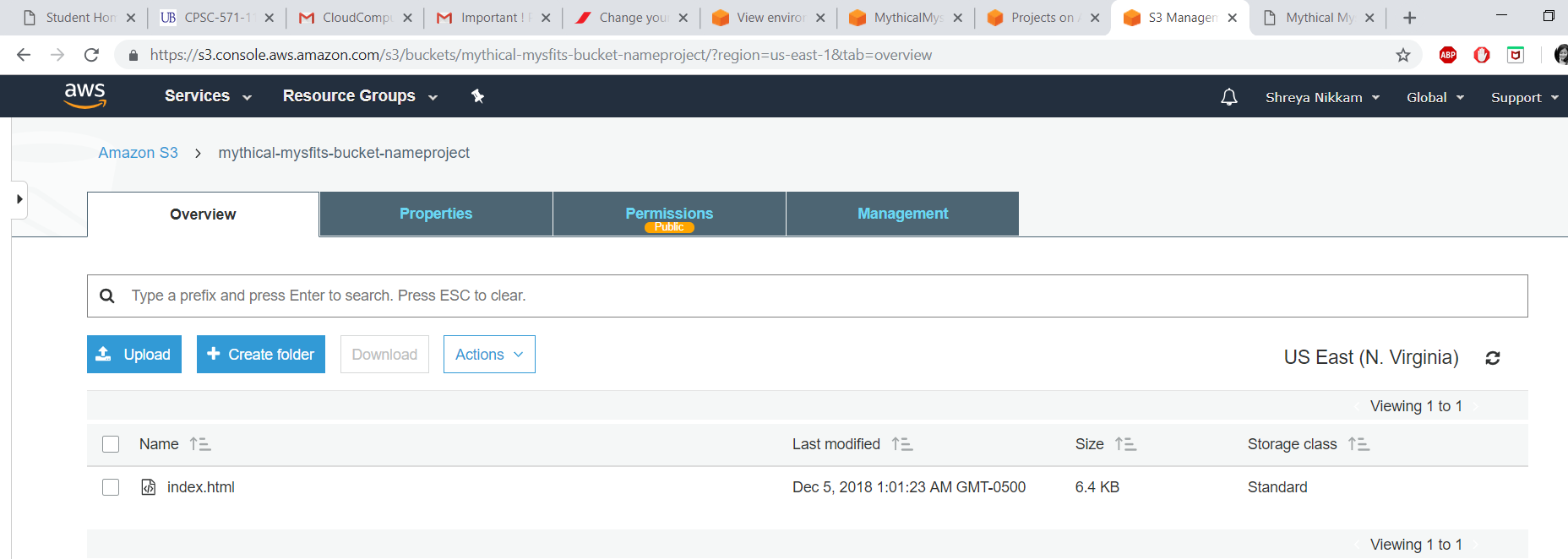
1. **Create Static Website:** Build a static website, using Amazon Simple Storage Service (S3) that serves static content.
2. **Build Dynamic Website:**Host your application logic on a web server, using an API backend microservice.
3. **Store Mysfit Data:** All of the mysfit is managed with NoSQL database provided by Amazon DynamoDB.
4. **Add User Registration:** Enable users to registration, authentication, and authorization so that Mythical Mysfits visitors can like and adopt mysfits, enabled through AWS API Gateway and its integration with Amazon Cognito.
5. **Capture User Clicks Capture:** It allows user behaviour to stream real time data analysis microservice that will record and analyse clicks on the website using AWS Lambda and Amazon Kinesis Firehose.

**Step by Step Procedure :**

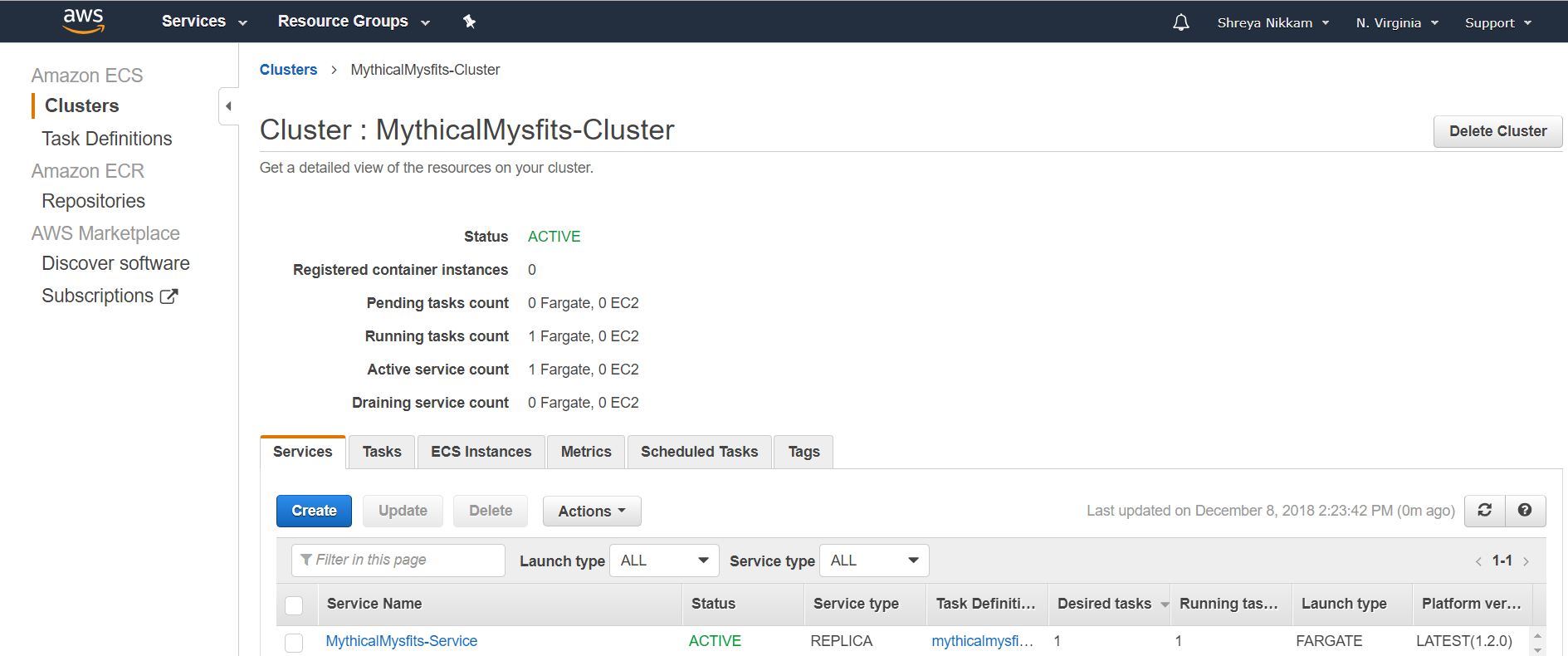
*Create Static Website using Cloud9 IDE*



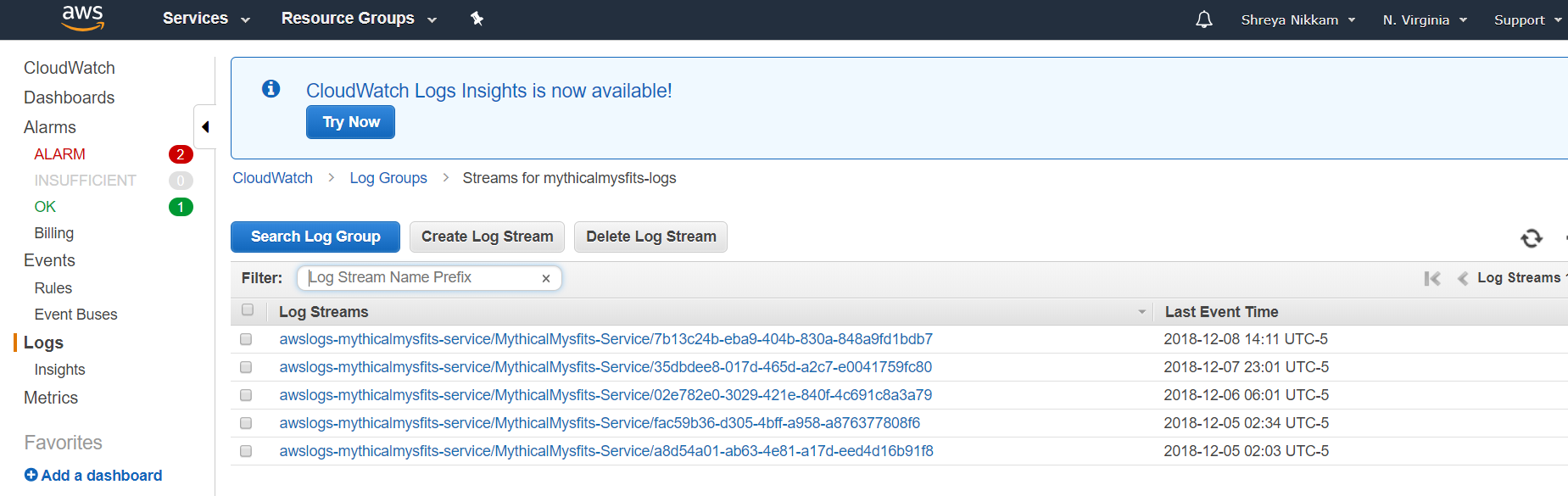
*Create S3 Bucket and Configure for Web Hosting*



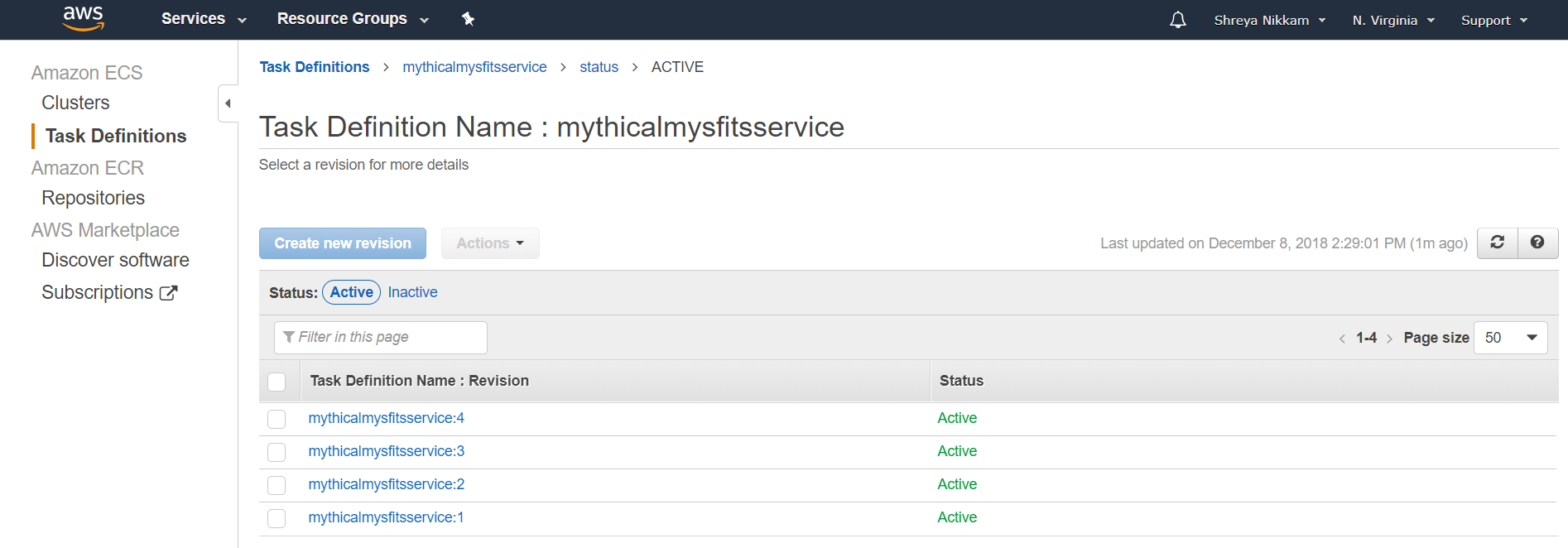
*Create an AWS Fargate Cluster*



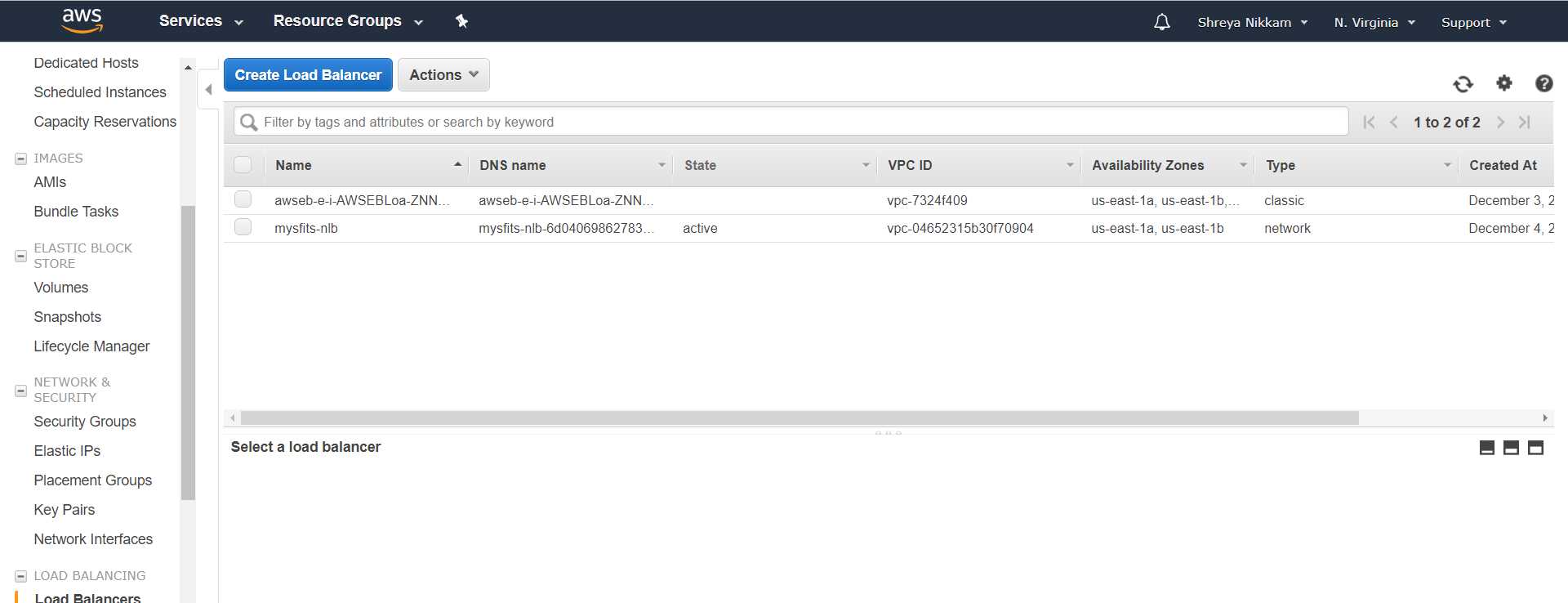
*Create an AWS Cloudwatch Logs Group*



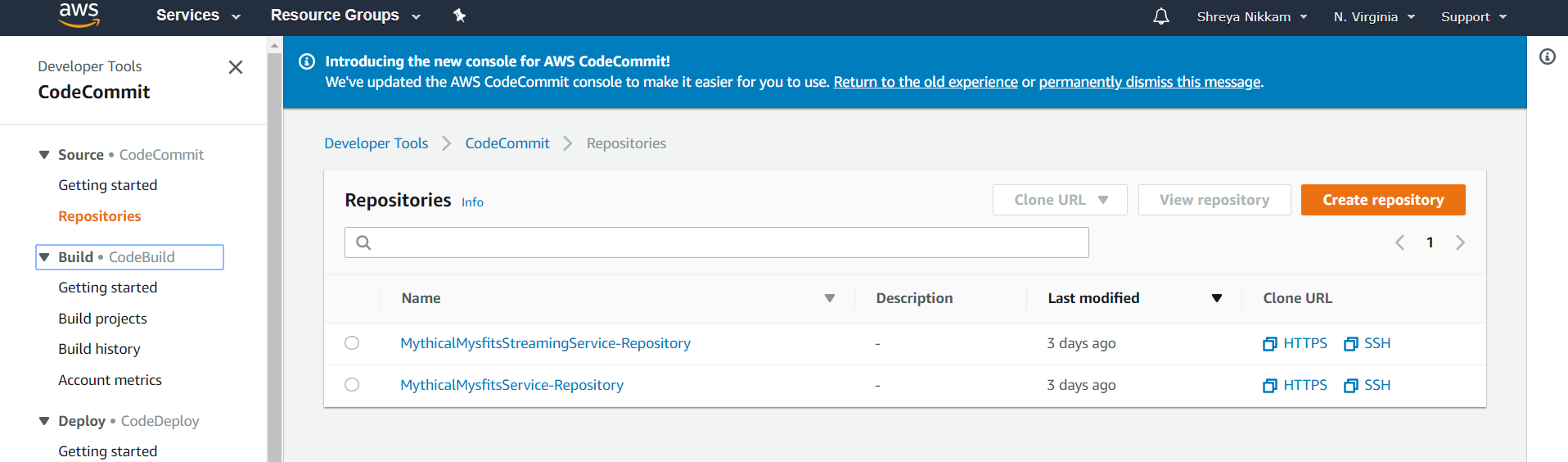
*Register an ECS Task Definition*



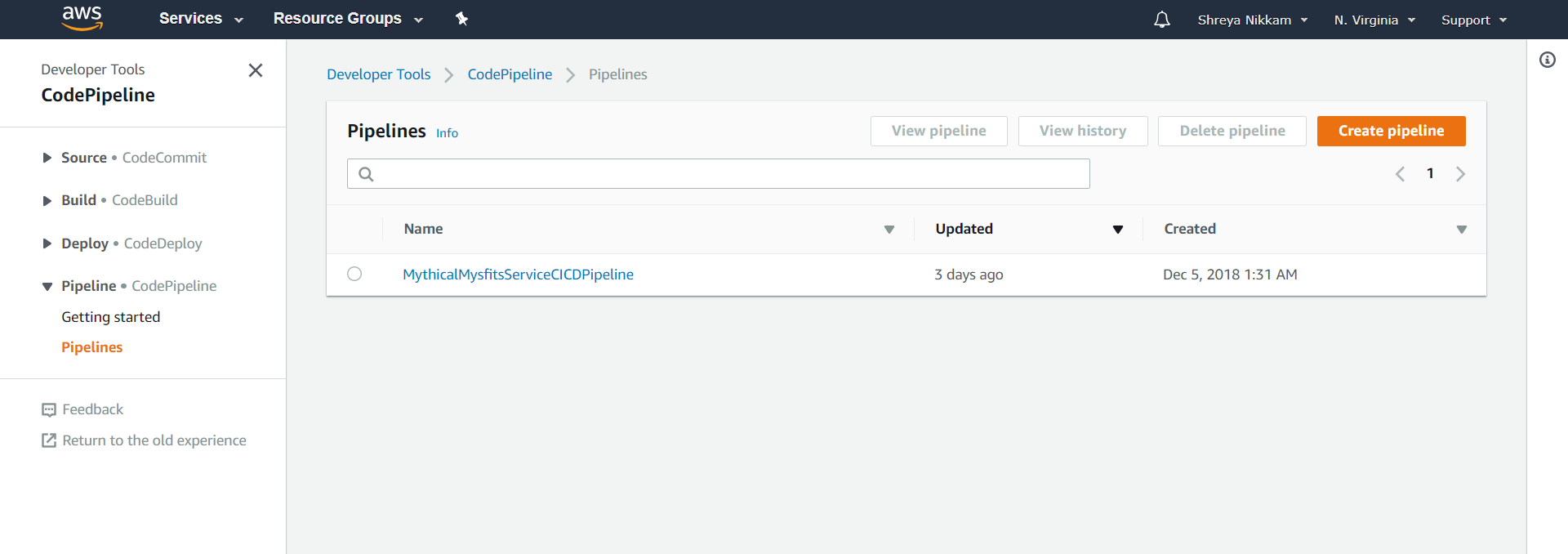
*Enable a Load Balancer Fargate Service*



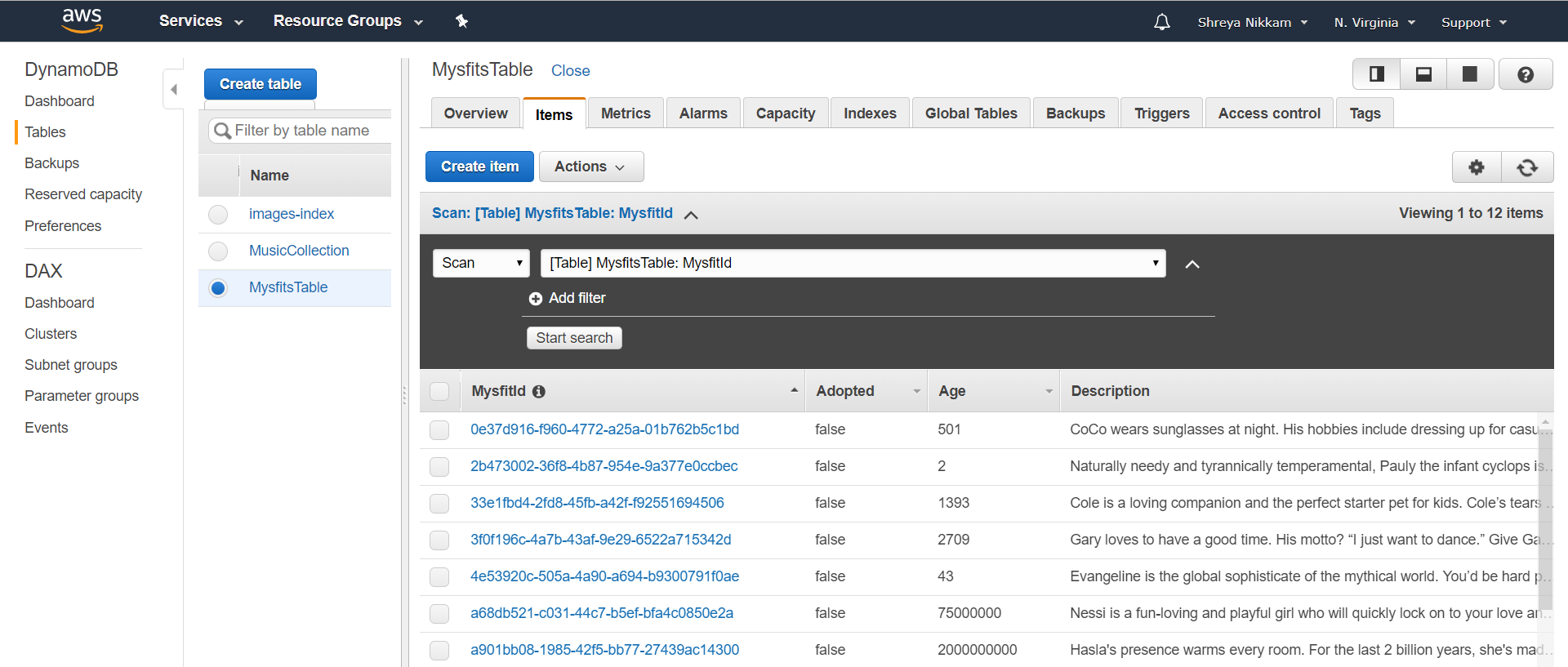
*Create a CodeCommit Repository*



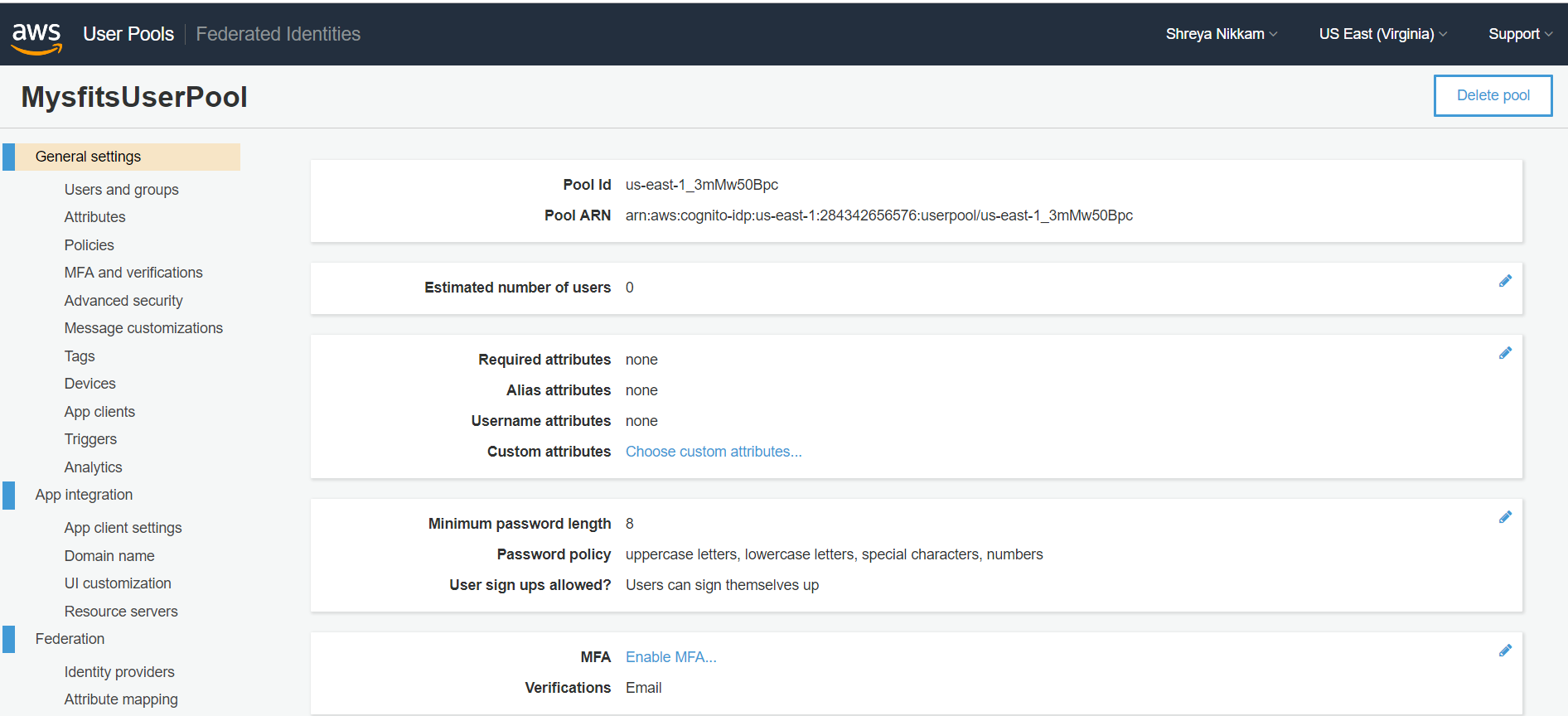
*Testing the CI/CD Pipeline*



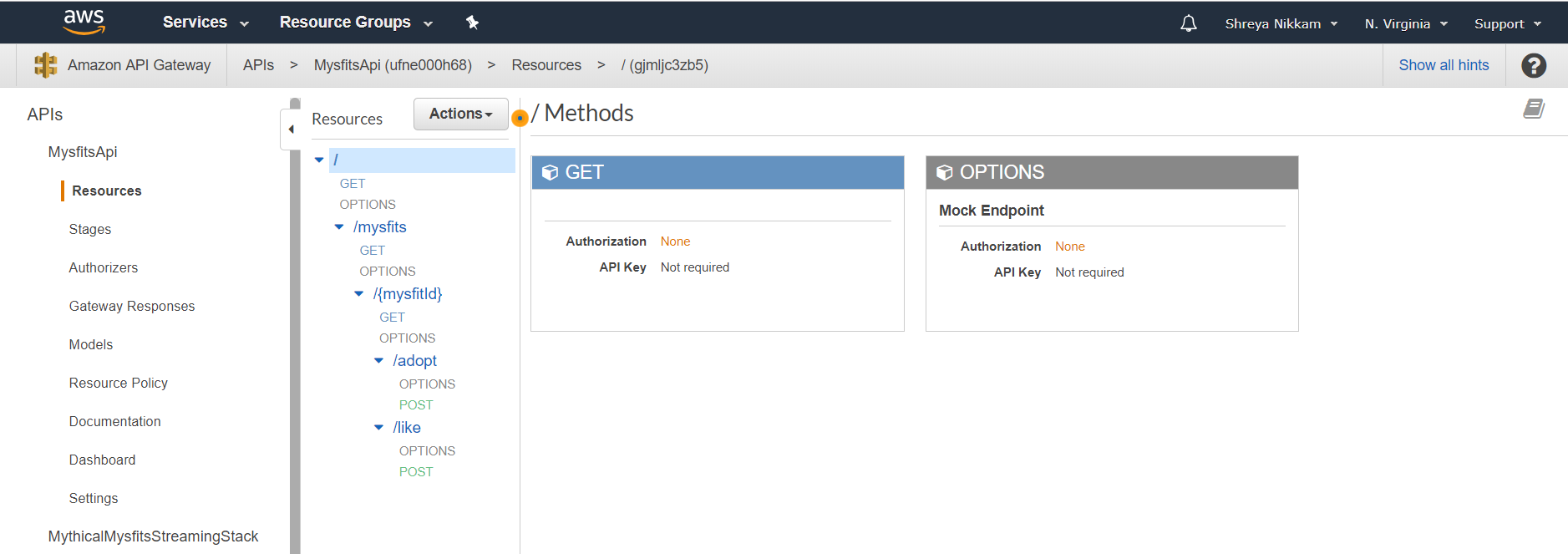
*Create and add items to Dynamo DB Table*



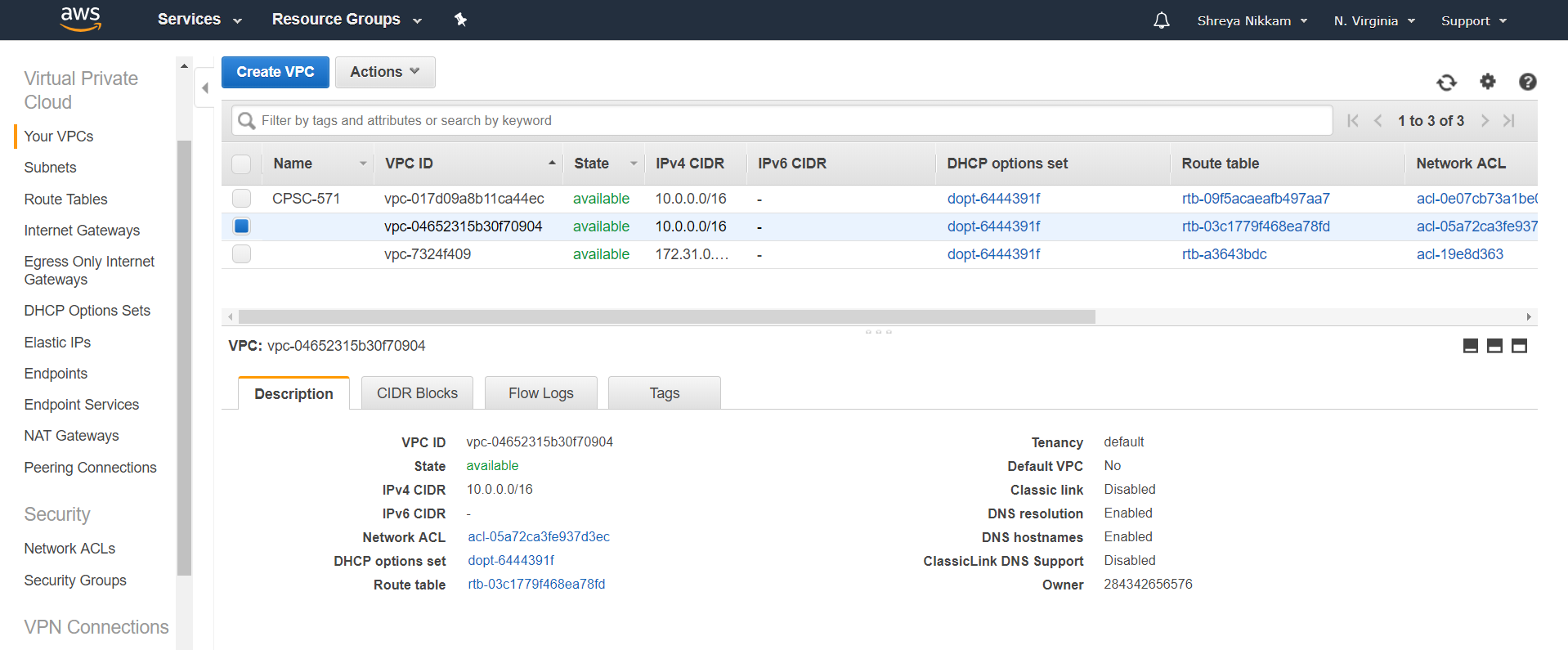
*Creating a Cognito User Pool for Website Users*



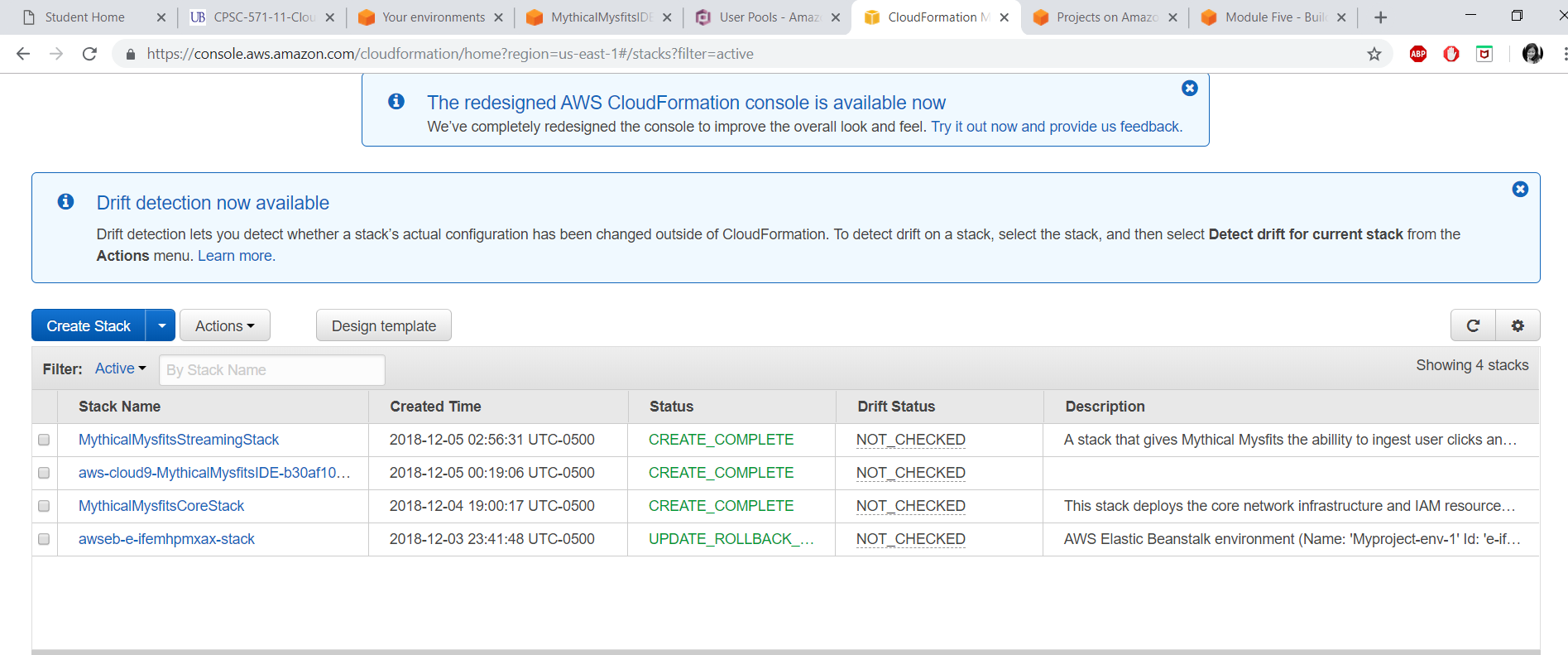
*Creating an API Gateway VPC Link*



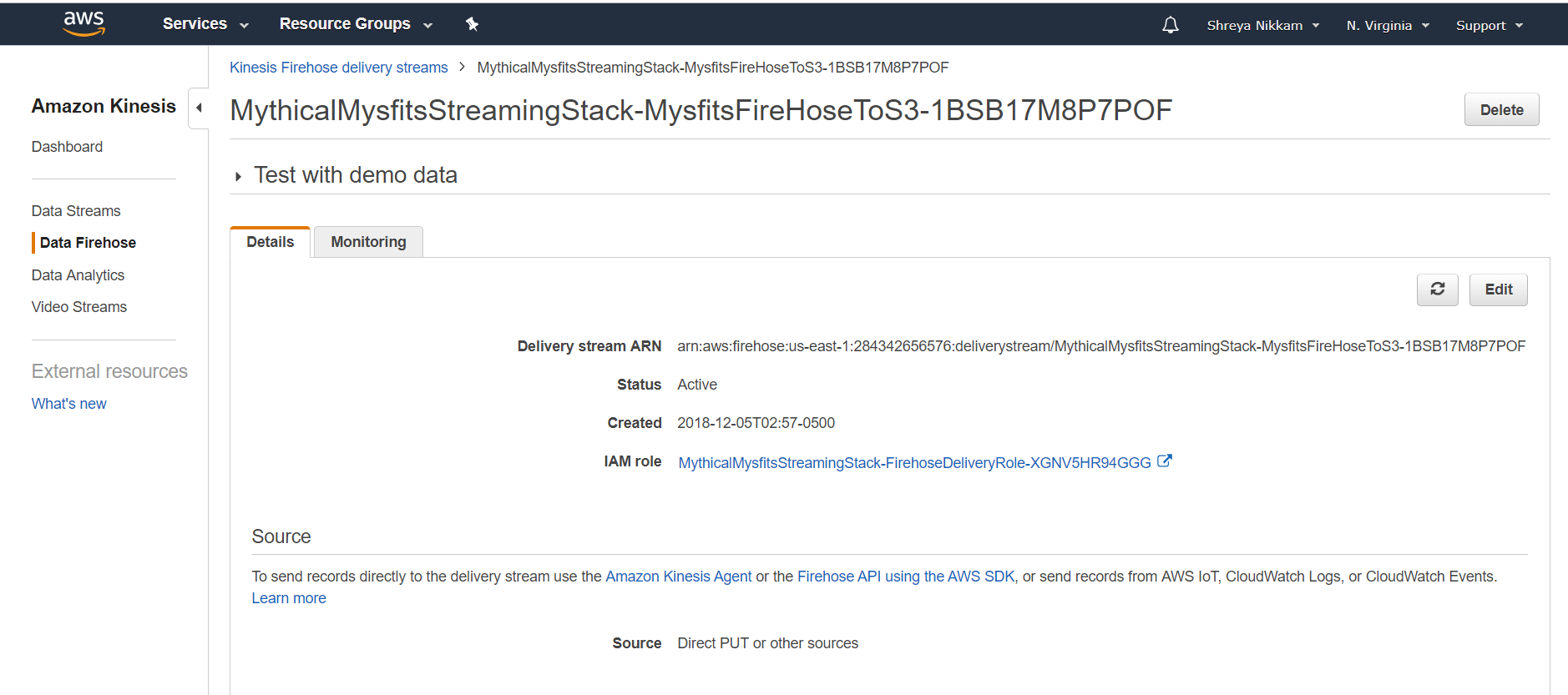
*Linking various VPC’s*



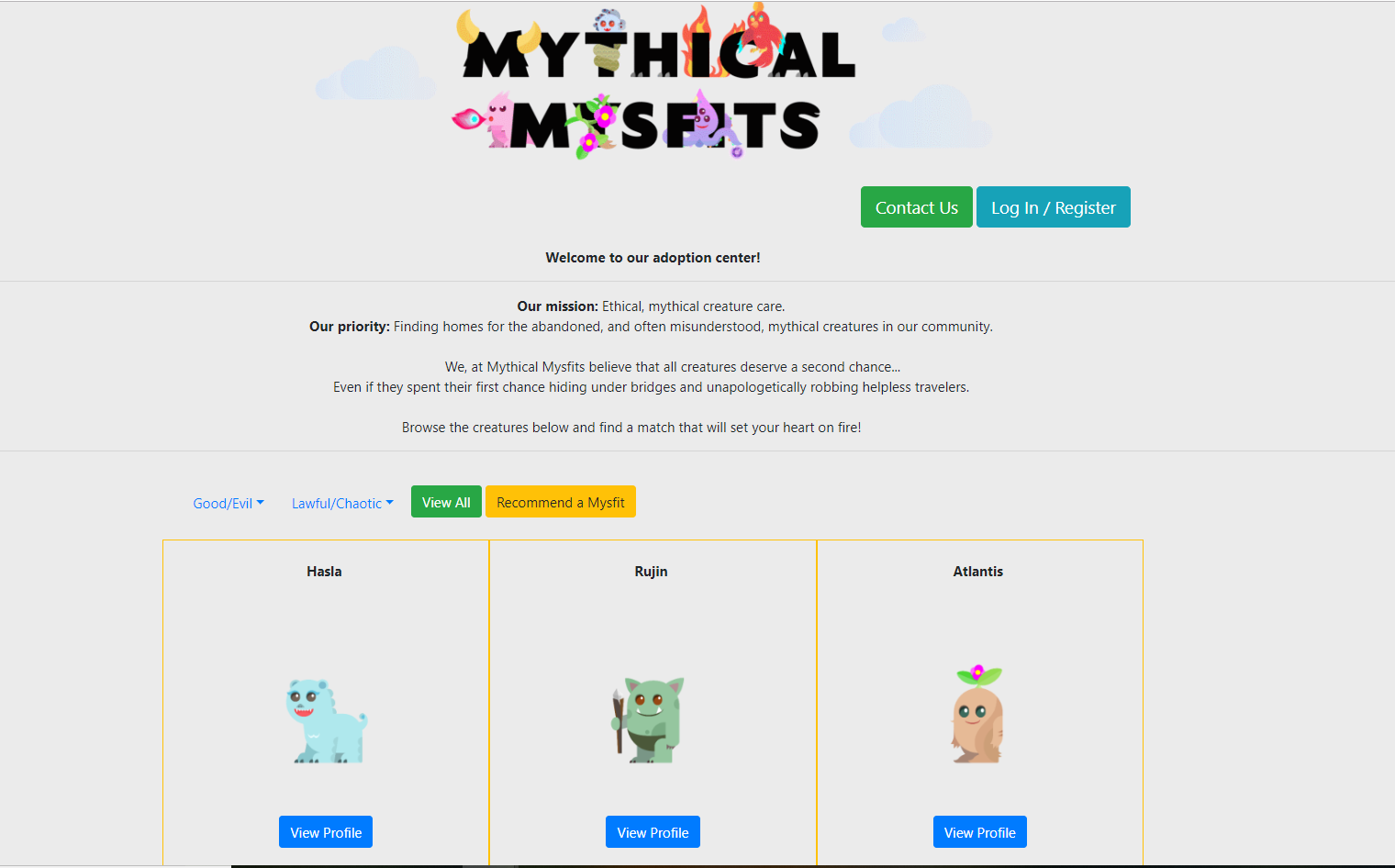
*Creating Streaming Service and Deploying Stack*



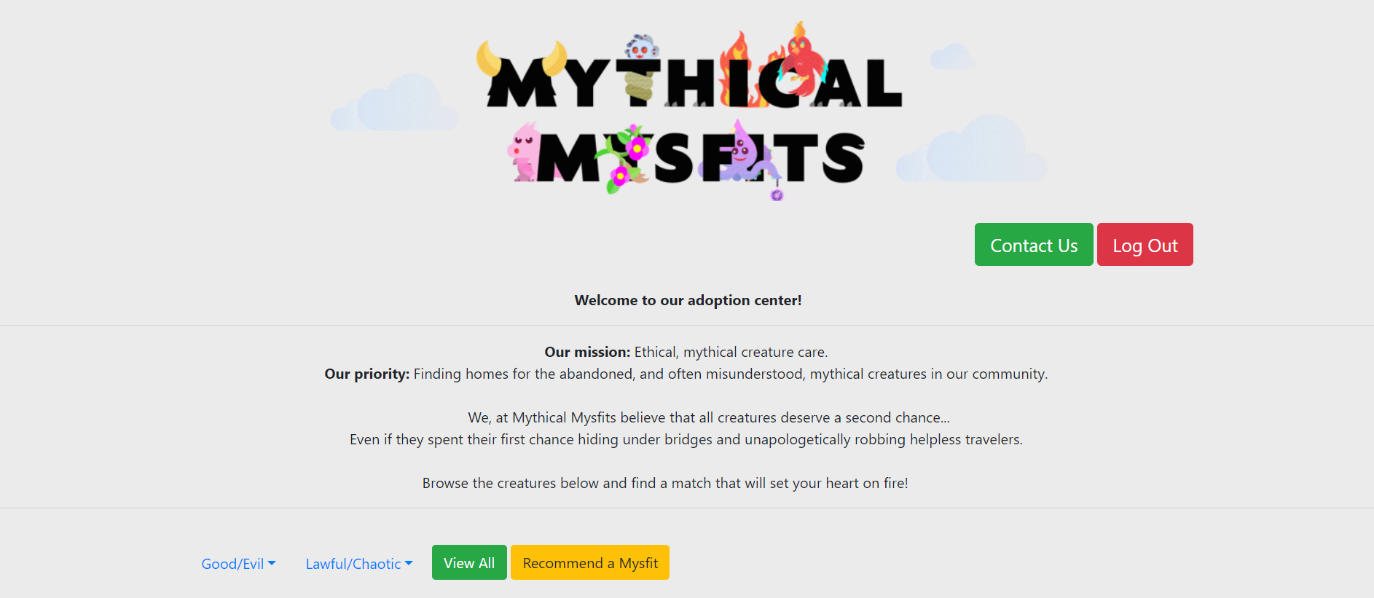
*Update website content and streaming real-time data using Amazon Kinesis*



*This is first snapshot of the Website, on deployment*



*Website after Users Log-in*



*Display of various Mysfit Creatures available for Adoption*

