SWE-455: Cloud Application Engineering

Homework 01

Deploying a simple Cloud-based application

Ali Elmatarwy

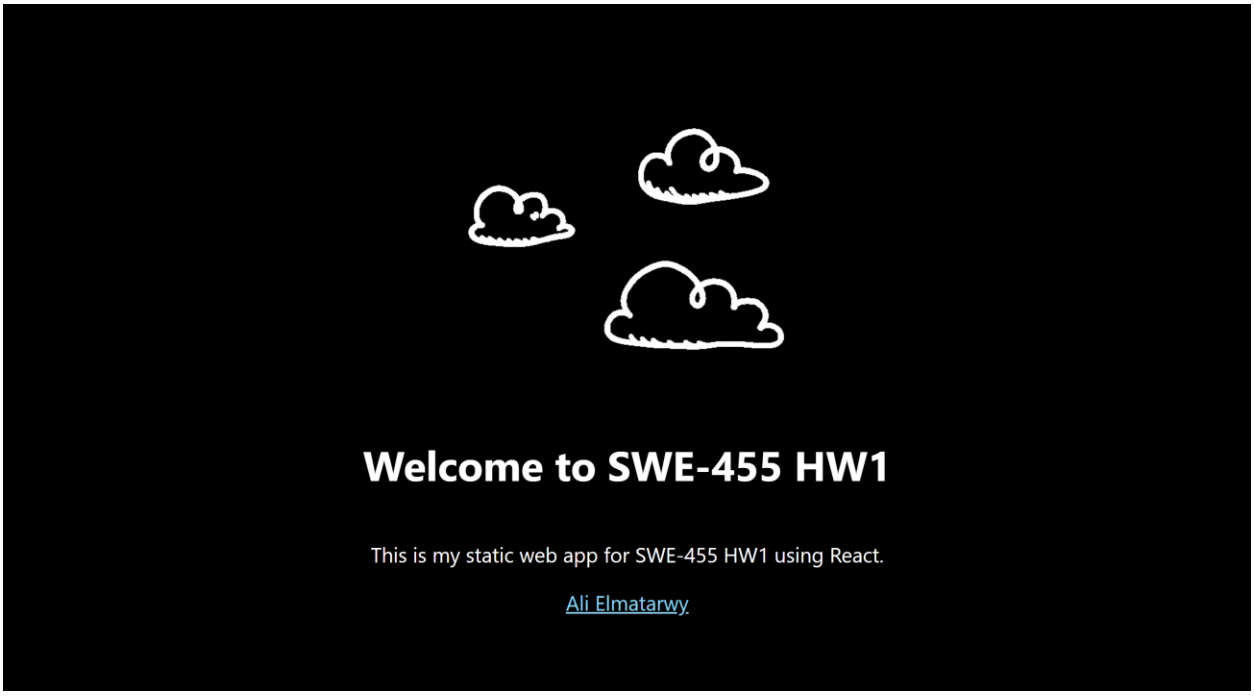
202045300

Task 1: Develop a sample application

For this assignment, I developed a simple web application using React. React was chosen because of its component-based architecture, ease of development, and strong community support. The application consists of a simple homepage displaying a welcome message.

Steps to Develop the Application:

1. Initializing React Project
    - Used npx create-react-app my-app to generate the project structure.
2. Develop the UI
    - Modified App.js to display a simple welcome page.
3. Test Locally
    - Ran npm start to ensure the application worked correctly before deployment.
4. Build the Project
    - Used npm run build to generate optimized static files for deployment.



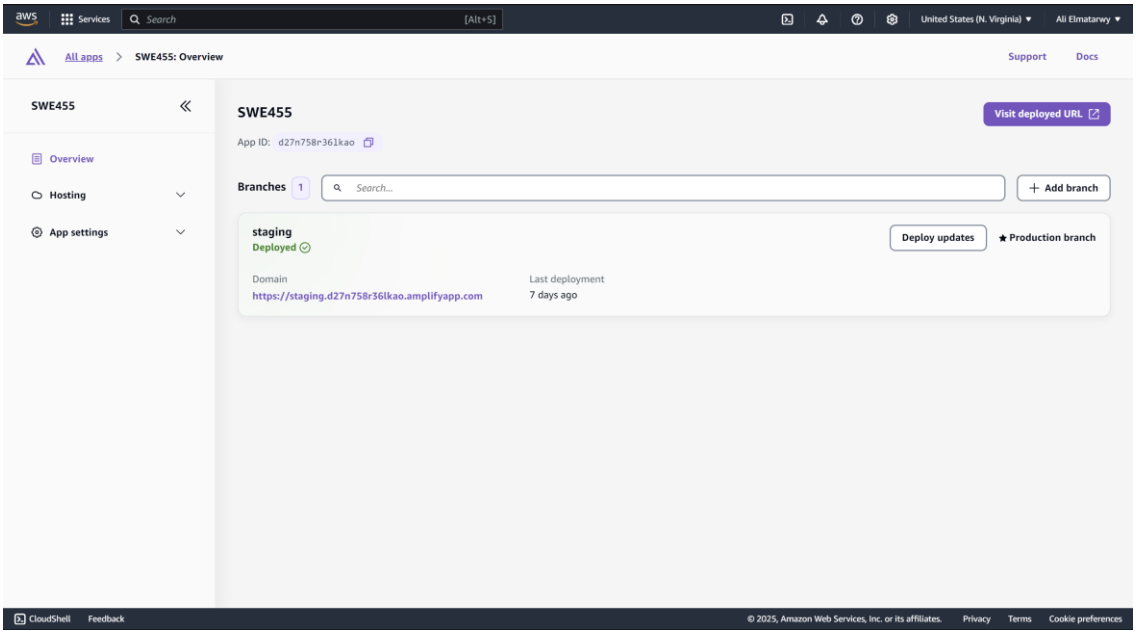
Task 2: Cloud deployment

The application was deployed using AWS Amplify, a cloud service that simplifies front-end deployment. Below are the steps I followed:

Steps to Deploy with AWS Amplify:

1. Create an AWS Amplify App
    - Logged into AWS and navigated to Amplify Console.
    - Clicked "New App" and selected "Host a web app."
2. Connect the Repository
    - Linked the GitHub repository containing my React application.
    - Configured build settings automatically detected by Amplify.
3. Deploying the Application
    - Used AWS's automated deployment pipeline to build and host the React app.
4. Public URL of the Deployed App:
    - Application Link

AWS Amplify Dashboard Showing Deployment:



Task 3: Scalability and Monitoring

To ensure scalability and monitoring, I configured AWS Amplify with the following:

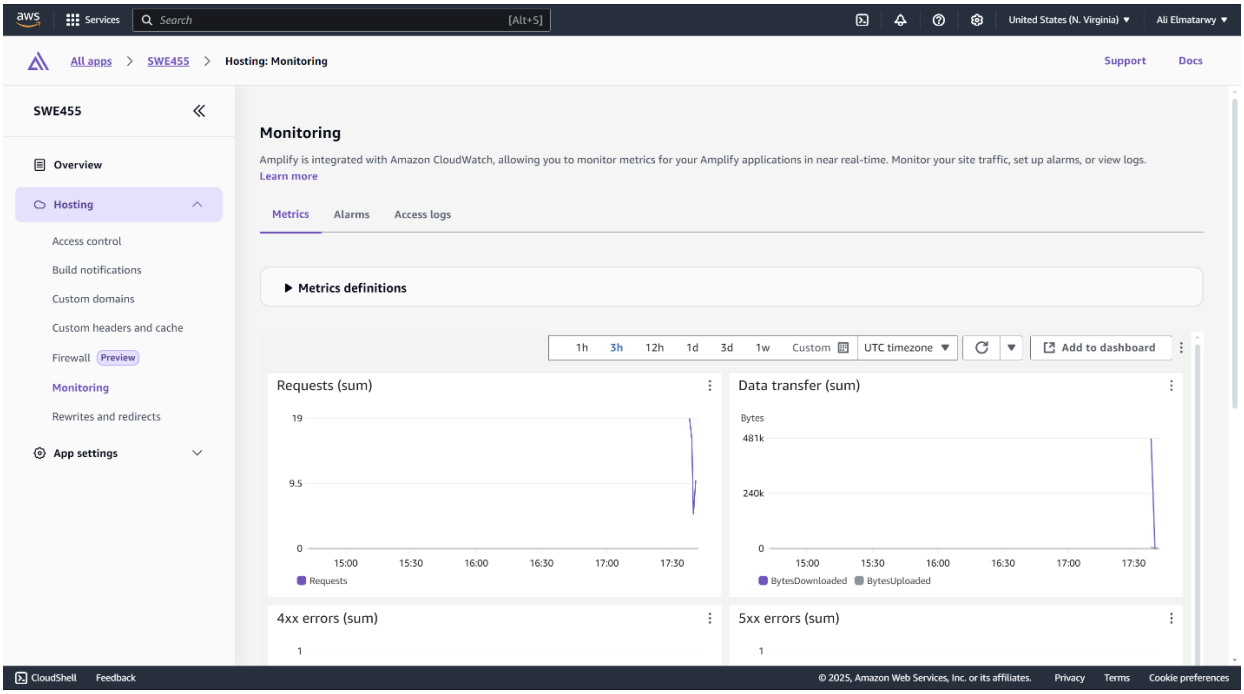
Scalability Setup:

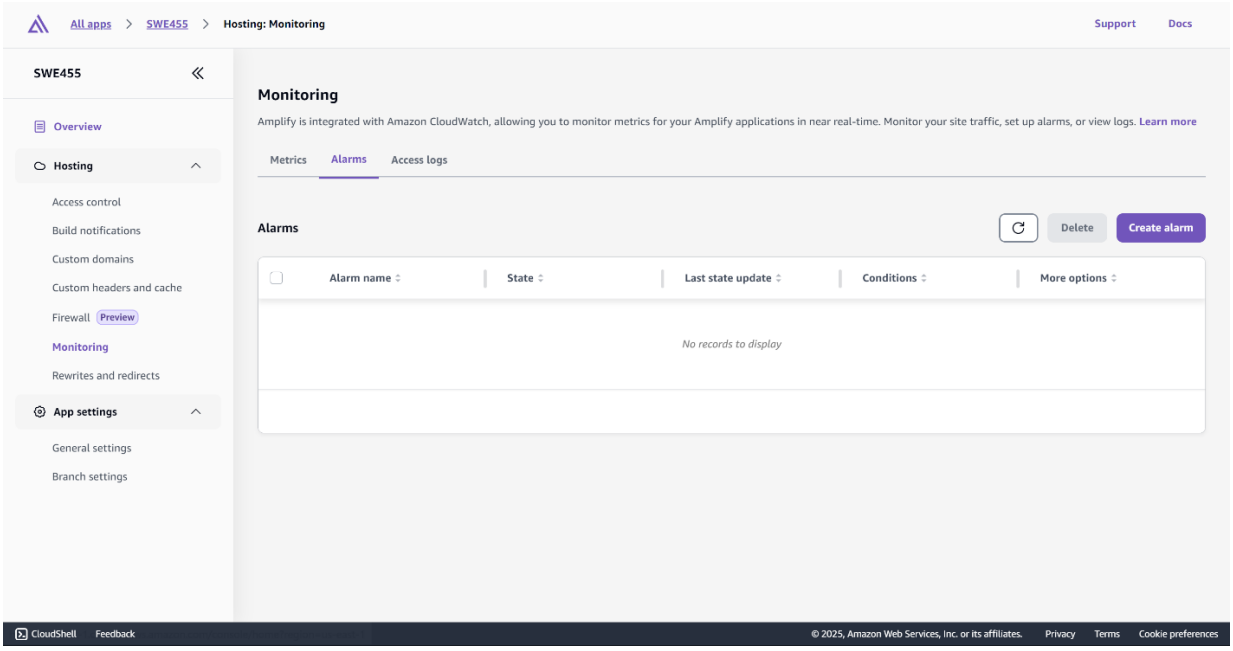
- AWS Amplify automatically scales based on traffic demands.
- The hosting service ensures multiple instances can handle traffic spikes.

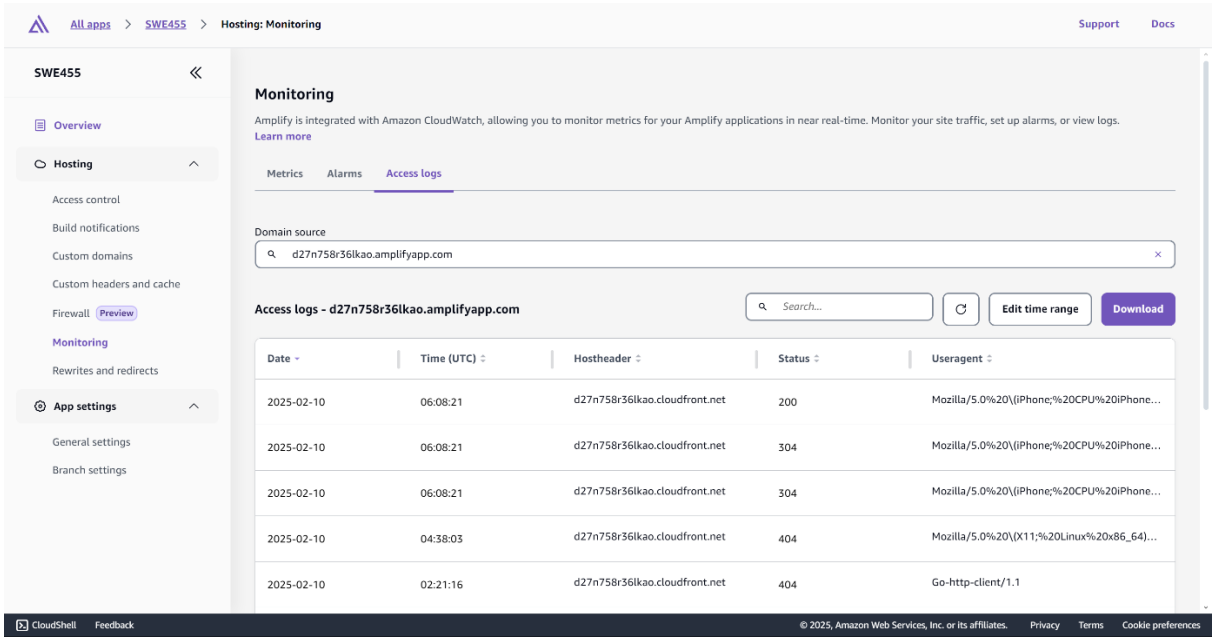
Monitoring with AWS CloudWatch:

- Enabled AWS CloudWatch Monitoring to track application performance.
- Metrics Collected:
    - Deployment status
    - Build duration
    - Request count

Screenshot of AWS Amplify Monitoring Dashboard:




All apps > SWE455 > Hosting: Monitoring
Support
Docs
SWE455
Overview
Hosting
Access control
Build notifications
Custom domains
Custom headers and cache
Firewall Preview
Monitoring
Rewrites and redirects
App settings
General settings
Branch settings
Monitoring
Amplify is integrated with Amazon CloudWatch, allowing you to monitor metrics for your Amplify applications in near real-time. Monitor your site traffic, set up alarms, or view logs. Learn more
Metrics
Alarms
Access logs
Alarms
Delete
Create alarm
Alarm name
State
Last state update
Conditions
More options
No records to display
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


All apps > SWE455 > Hosting: Monitoring
Support
Docs
SWE455
Overview
Hosting
Access control
Build notifications
Custom domains
Custom headers and cache
Firewall Preview
Monitoring
Rewrites and redirects
App settings
General settings
Branch settings
Monitoring
Amplify is integrated with Amazon CloudWatch, allowing you to monitor metrics for your Amplify applications in near real-time. Monitor your site traffic, set up alarms, or view logs.
Learn more
Metrics
Alarms
Access logs
Domain source
d27n758r36lkao.amplifyapp.com
Access logs - d27n758r36lkao.amplifyapp.com
Search...
Edit time range
Download
Date
Time (UTC)
Hostheader
Status
Useragent
2025-02-10 06:08:21 d27n758r36lkao.cloudfront.net 200 Mozilla/5.0%20\(iPhone;%20CPU%20iPhone...
2025-02-10 06:08:21 d27n758r36lkao.cloudfront.net 304 Mozilla/5.0%20\(iPhone;%20CPU%20iPhone...
2025-02-10 06:08:21 d27n758r36lkao.cloudfront.net 304 Mozilla/5.0%20\(iPhone;%20CPU%20iPhone...
2025-02-10 04:38:03 d27n758r36lkao.cloudfront.net 404 Mozilla/5.0%20\(X11;%20Linux%20x86_64)...
2025-02-10 02:21:16 d27n758r36lkao.cloudfront.net 404 Go-http-client/1.1
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Challenges Faced and Solutions:

1. Build Errors in AWS Amplify
    - Issue: Initially, the building failed due to missing dependencies.
    - Solution: Added missing dependencies in package.json and retried deployment.
2. Public Access Issue
    - Issue: The deployed app was not accessible due to incorrect AWS IAM settings.
    - Solution: Adjusted Amplify permissions to make the app publicly accessible.

Application Running in Browser: