Monitoring Application Metrics Using AWS CloudWatch

A Course Project Report Submitted in partial fulfillment of the course requirements for the award of grades in the subject of

CLOUD BASED AIML SPECIALITY
(22SDCS07A)

by

Vaibhav L Jawalkar

2210030065

Under the esteemed guidance of

Ms. P. Sree Lakshmi
Assistant Professor,
Department of Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

K L Deemed to be UNIVERSITY

Aziznagar, Moinabad, Hyderabad,
Telangana, Pincode: 500075

April 2025

K L Deemed to be UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Certificate

This is Certified that the project entitled **"Monitoring Application Metrics Using AWS CloudWatch"** which is a experimental &/ theoretical &/ Simulation&/ hardware work carried out by Vaibhav L Jawalkar (2210030065), in partial fulfillment of the course requirements for the award of grades in the subject of **CLOUD BASED AIML SPECIALITY**, during the year **2024-2025**. The project has been approved as it satisfies the academic requirements.

Ms.P.Sree Lakshmi	**Dr. Arpita Gupta**
Course Coordinator	**Head of the Department**

Ms. P. Sree Lakshmi

Course Instructor

CONTENTS

Page No.

1. INTRODUCTION

Importance of Cloud Monitoring

In today's cloud-centric world, monitoring application performance is crucial for ensuring reliability, efficiency, and cost-effectiveness. With businesses increasingly relying on cloud-based infrastructure, it becomes essential to proactively track system performance and ensure that applications run smoothly without unexpected downtime. Any service disruption can result in lost revenue, decreased customer satisfaction, and potential security vulnerabilities.

Overview of AWS CloudWatch

Amazon Web Services (AWS) provides a robust set of tools for monitoring and managing cloud resources. One of the most powerful tools in its suite is AWS CloudWatch, which allows users to collect and track performance metrics, monitor logs, and set alarms for various cloud services. This project focuses on using AWS CloudWatch to monitor the performance of an Amazon EC2 (Elastic Compute Cloud) instance, specifically tracking CPU utilization. By setting up alarms and notifications, we can proactively manage our applications and respond to performance issues in real-time.

Project Scope and Objectives

The main objectives of this project are:

- Deploy an EC2 instance and enable monitoring.
- Use AWS CloudWatch to track CPU utilization.
- Set up alarms based on predefined thresholds.
- Configure Amazon SNS to send notifications when alarms are triggered.
- Understand the importance of cloud monitoring in maintaining application reliability.

2. AWS Services Used as part of the project

Amazon EC2 (Elastic Compute Cloud)

Amazon EC2 is a web service that provides resizable compute capacity in the cloud. It allows users to run virtual servers and manage applications efficiently. Key features include:

- Scalability to handle varying workloads.
- Customizable configurations, including instance types and AMIs.
- Integrated monitoring with AWS CloudWatch.

Amazon CloudWatch

Amazon CloudWatch is a monitoring and observability service that provides data and insights to track application performance. Key benefits include:

- Real-time monitoring of EC2 instances.
- Alarm setup for detecting performance anomalies.
- Logging and metric visualization.

Amazon SNS (Simple Notification Service)

Amazon SNS is a fully managed messaging service that enables sending notifications to subscribers. In this project, SNS is used to notify users when predefined CPU utilization thresholds are breached.

3. Steps involved in solving project problem statement

Step 1: Log into AWS Free Tier Account

- Access the AWS Management Console using credentials.
- Ensure AWS Free Tier eligibility for cost-effective usage.

Step 2: Create an EC2 Instance

- Navigate to the EC2 dashboard and select "Launch Instance."
- Choose an appropriate AMI (Amazon Machine Image), such as Amazon Linux 2.
- Select an instance type based on performance requirements.
- Enable CloudWatch monitoring during instance creation.

Step 3: Check Instance Status

- Verify that the instance is running without issues.

- Ensure instance health checks are passed.

Step 4: Access CloudWatch

- Navigate to the CloudWatch service from the AWS Management Console.
- Select EC2 metrics and monitor CPU utilization in real-time.

Step 5: Connect to the EC2 Instance

- Use SSH to establish a secure connection.
- Execute commands to monitor CPU usage.

Step 6: Set Up Amazon SNS for Notifications

- Create an SNS topic in AWS.
- Subscribe an email or phone number for receiving notifications.
- Ensure SNS is configured to trigger alerts from CloudWatch alarms.

Step 7: Terminate the EC2 Instance

- To avoid unnecessary charges, terminate the instance after testing.
- Verify resource cleanup in the EC2 dashboard.

Step 8: Launch a New EC2 Instance

- Repeat the instance creation process to set up monitoring and alarms effectively.

Step 9: Create an Alarm in CloudWatch

- Define threshold limits for CPU utilization.
- Configure CloudWatch to trigger alerts via SNS when limits are breached.

Step 10: Monitor Alarm Status

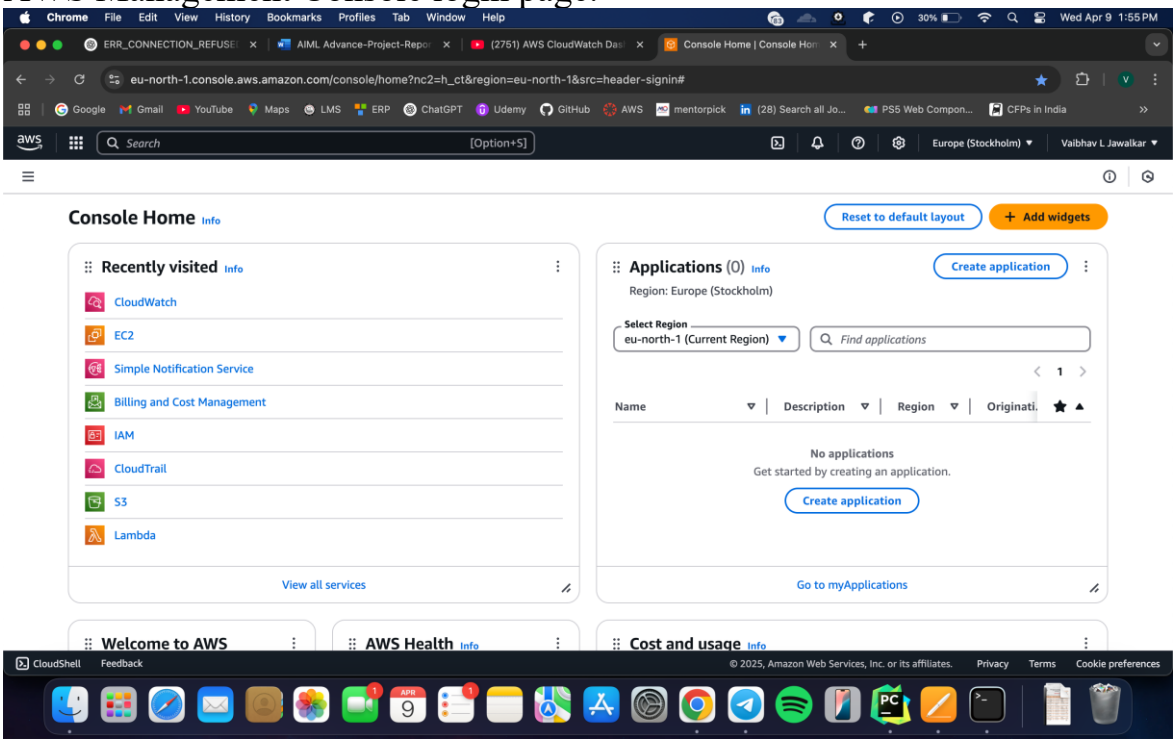
- Observe the alarm's state and response to different workload scenarios.
- Fine-tune alarm settings for optimal performance tracking.

Step 11: Receive Notifications

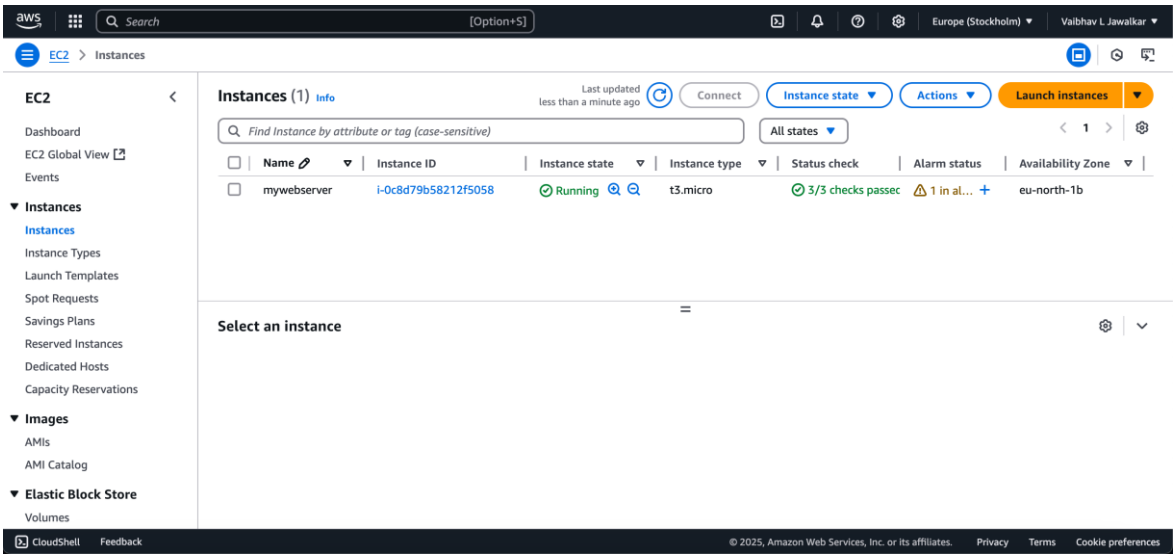
- Confirm receipt of SNS alerts when the alarm is triggered.
- Verify notification accuracy and responsiveness.

4. Stepwise Screenshots with brief description

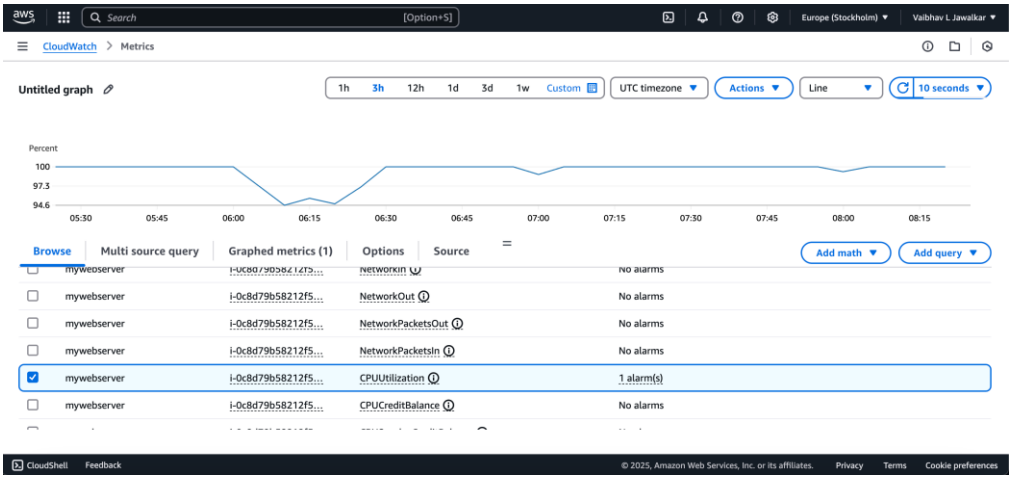
- AWS Management Console login page.



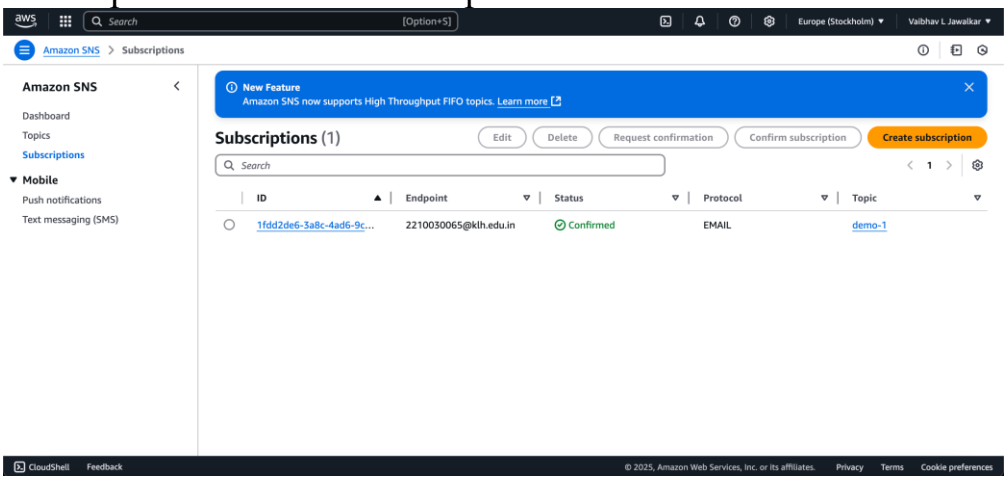
- EC2 instance creation settings.



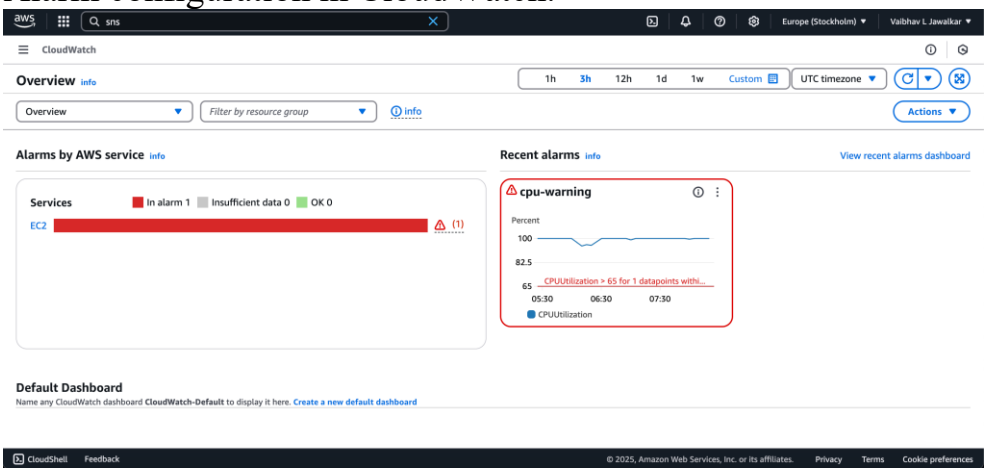
- CloudWatch metrics dashboard.



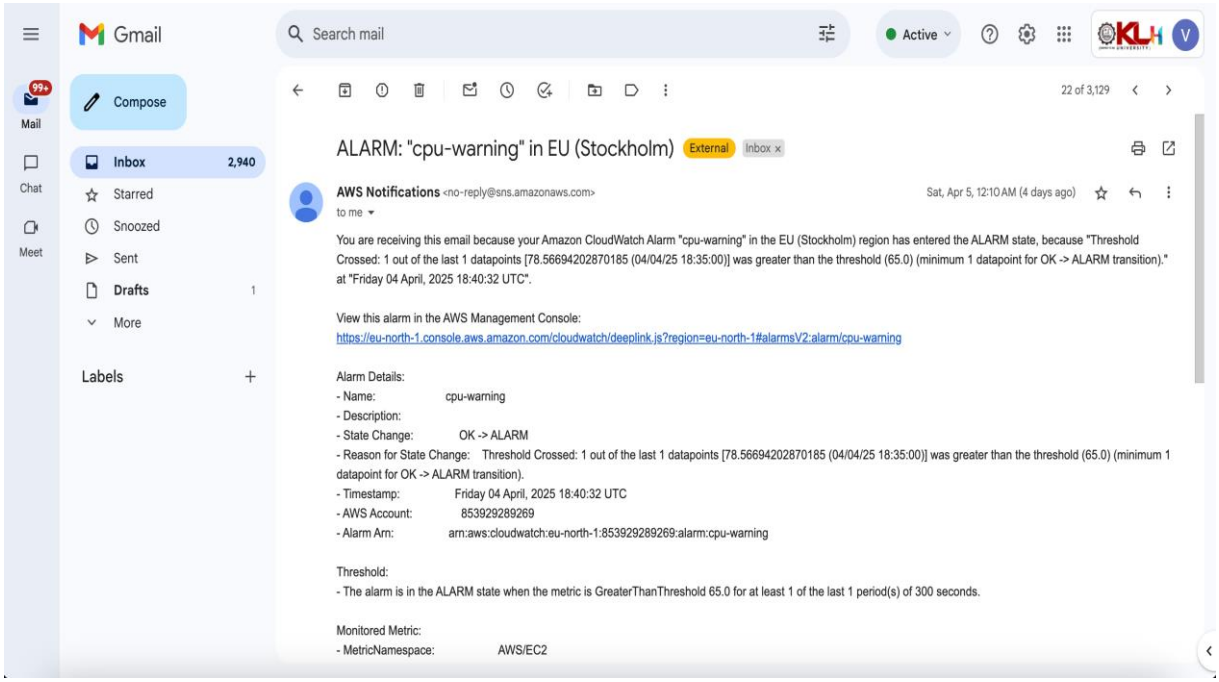
- SNS topic creation and subscription verification.



- Alarm configuration in CloudWatch.



- Notification received via email.



5. Learning Outcomes

By completing this project, users will:

- Understand the importance of monitoring application performance in the cloud.
- Gain hands-on experience with AWS services such as EC2, CloudWatch, and SNS.
- Learn how to set up alarms and notifications to proactively manage application performance.
- Develop skills in using the AWS Management Console for resource management.
- Enhance their ability to respond to real-time performance issues in cloud environments.

6. Conclusion

This project successfully demonstrates how to monitor application metrics using AWS CloudWatch. By creating an EC2 instance, enabling monitoring, and setting up alarms, we ensure that applications run efficiently and respond to performance issues in real-time.

The integration of Amazon SNS for notifications further enhances the monitoring process, allowing administrators to receive instant alerts and take timely actions. By mastering these monitoring techniques, businesses can improve application reliability, optimize resource usage, and enhance overall system performance. Furthermore, the project underscores the importance of cloud-based observability tools in modern infrastructure management, highlighting how real-time monitoring contributes to proactive decision-making, operational efficiency, and cost savings in cloud computing environments.

7. References

- **Amazon Web Services Documentation**

1. AWS CloudWatch: https://docs.aws.amazon.com/cloudwatch/
2. AWS QuickSight: https://docs.aws.amazon.com/quicksight/
3. AWS IAM: https://docs.aws.amazon.com/IAM/
4. Amazon S3: https://docs.aws.amazon.com/s3/
5. AWS Glue: https://docs.aws.amazon.com/glue/
6. AWS EventBridge: https://docs.aws.amazon.com/eventbridge/

- **AWS Architecture Center**

1. https://aws.amazon.com/architecture/

- **AWS Blogs – Data Analytics**

1. https://aws.amazon.com/blogs/big-data/

- **AWS YouTube Channel – Tutorials & Walkthroughs**

1. https://www.youtube.com/user/AmazonWebServices

- **Official AWS Well-Architected Framework**

1. https://aws.amazon.com/architecture/well-architected/