Lab No. 1

Welcome to AWS Academy Cloud Developing

Objective

→ To understand the course structure, objectives, the business case (Cafe scenario) and different roles in cloud computing that will be referenced throughout the AWS Academy Cloud Developing Course

Theory

→ This Module provides an overview of cloud computing and how AWS support developers in building scalable applications. It introduces the Cafe business case - a fictional company used throughout the course to simulate real world cloud development challenges. The Module also explains key roles in cloud environments, such as cloud developers, solution architects, devops engineers and cloud administrators.

Tools / Services used:

- AWS Academy Learner Lab Platform
- Student Guide (AWS Academy)
- AWS Academy Learning Portal
- External Learning Tools & Videos

Procedure / Steps:

1. Accessed AWS Academy Cloud Developing Course through the learner Lab.

2. Reviewed the course objectives and learning roadmap in Part 1 & Part 2.
3. Studied the Cafe Business Cafe introduction, learning how AWS services will be applied to solve business problems.
4. Explored the Roles in cloud computing section to understand responsibilities of different cloud professionals.
5. Watched the Module 1 wrap up video summarizing key learnings.

Output / Observations

- Understand the main focus areas of the AWS Academy cloud Developing course.
- Learned about fictional Cafe company and how AWS services can modernize business applications.
- Identified core roles in the cloud industry and how they collaborate in real-world cloud projects.

Conclusion

→ Module 1 provided a foundational understanding of what to expect from the AWS cloud Developing course. It helped me recognize how cloud developers, architects and operations teams work together to build scalable, secure and efficient applications using AWS services.

12/11/25

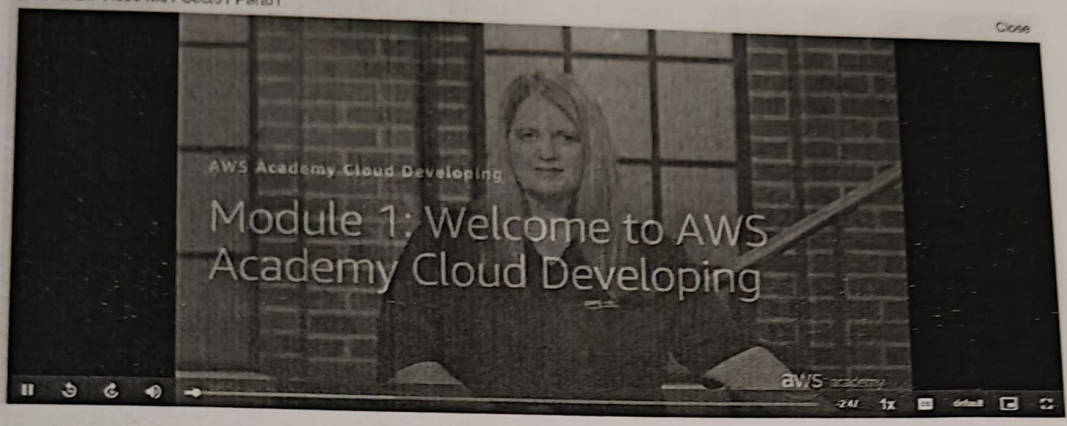
Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Module 1: Welcome to AWS Academy Cloud Developing

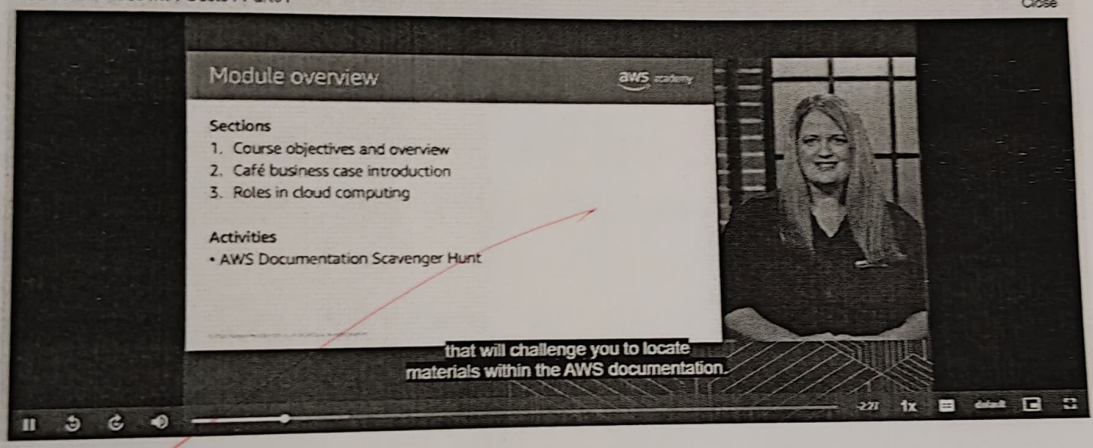
ACDv2 EN Video M01 Sect01 Part01

Close



ACDv2 EN Video M01 Sect01 Part01

Close



Wednesday, 17 September 2025

Exploring AWS Cloudshell and an IDE

Objective

To understand and explore the use of AWS Cloudshell and an Integrated Development Environment (IDE) for developing, testing and Managing AWS resources efficiently through the command line and code editor.

Theory

AWS Cloudshell is a browser-based shell that makes it easy to securely manage, interact and automate AWS resources. It provides the AWS command Line Interface (CLI) pre-installed, removing the need for Local configurations.

An IDE (like Vs Code or cloud9) helps developers write, debug and deploy applications faster by integrating AWS SDKs and extensions for easier cloud development.

Tools / services Used

- AWS Cloudshell
- AWS CLI
- AWS SDKs (Software Development Kits)
- Visual Studio Code
- Amazon Q Developer

Procedure / Steps

1. Logged into AWS Academy Learner Lab
2. Launched AWS Cloudshell from the Management Console.
3. Explored build in AWS CLI (commands) (aws s3 ls, aws ec2

describe instance etc)

4. Set up AWS SDK environment for Python or Node.js within CloudShell.
5. Installed and configure Visual Studio code with AWS Toolkit extension.
6. Connected the IDE to AWS account to run simple Lambda or S3 operations.
7. Explored Amazon Q Developer

Output / Results.

- Successfully executed AWS CLI commands inside CloudShell.
- Verified SDK functionality by listing AWS resources through code.
- Connected IDE to AWS and performed basic cloud operations
- Screenshots of CloudShells and IDE configuration attached.
- Knowledge Check completed with a score of

Conclusion

→ I Learned how to use AWS CloudShell for command-line access to AWS services and how to integrate an IDE for development and testing. This lab enhanced my understanding of cloud-based development workflows and the convenience of managing AWS resources without local setup.

12/11/25

Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 2.1: Exploring AWS CloudShell and an IDE

Due No Due Date **Points** 100 **Submitting** an external tool

Submit | Details | AWS | Start Lab | End Lab | --:-- | Instructions | Grades | Actions

Files ☐ README ☑ Terminal ☑ Source ☐

EN-US

Submission

Sep 10 at 10:28am

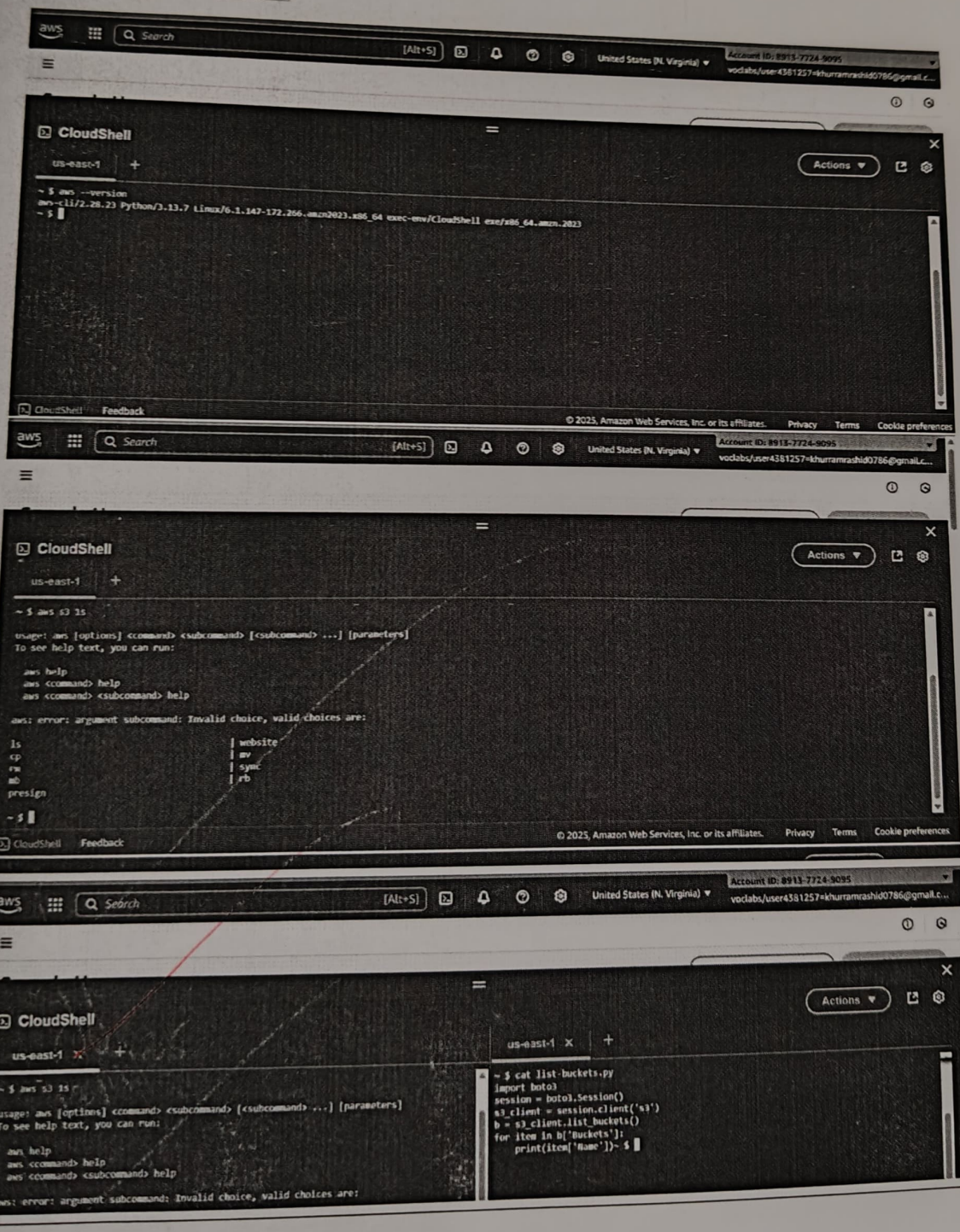
Submission Details

Grade: 100 (100 pts possible)

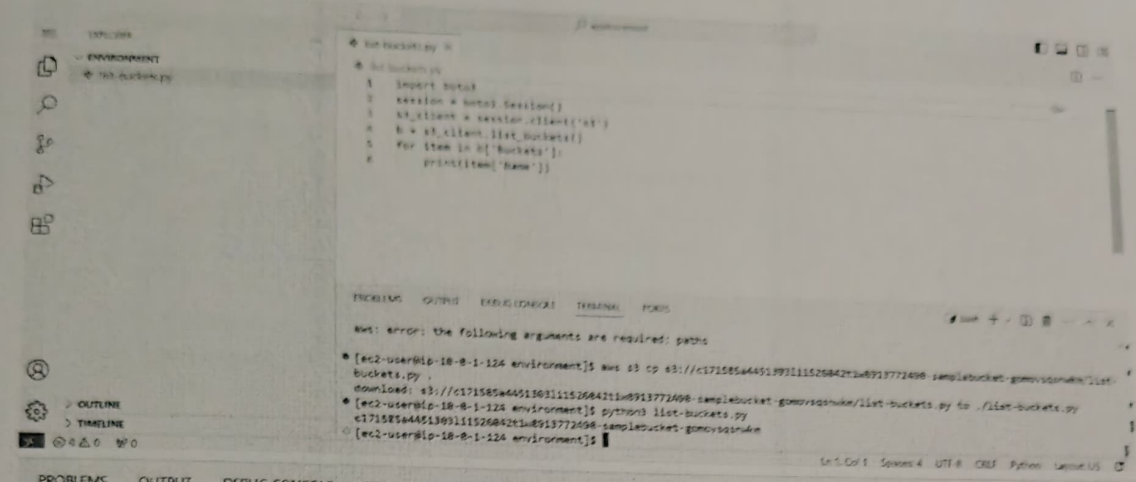
Graded Anonymously: no

Comments:

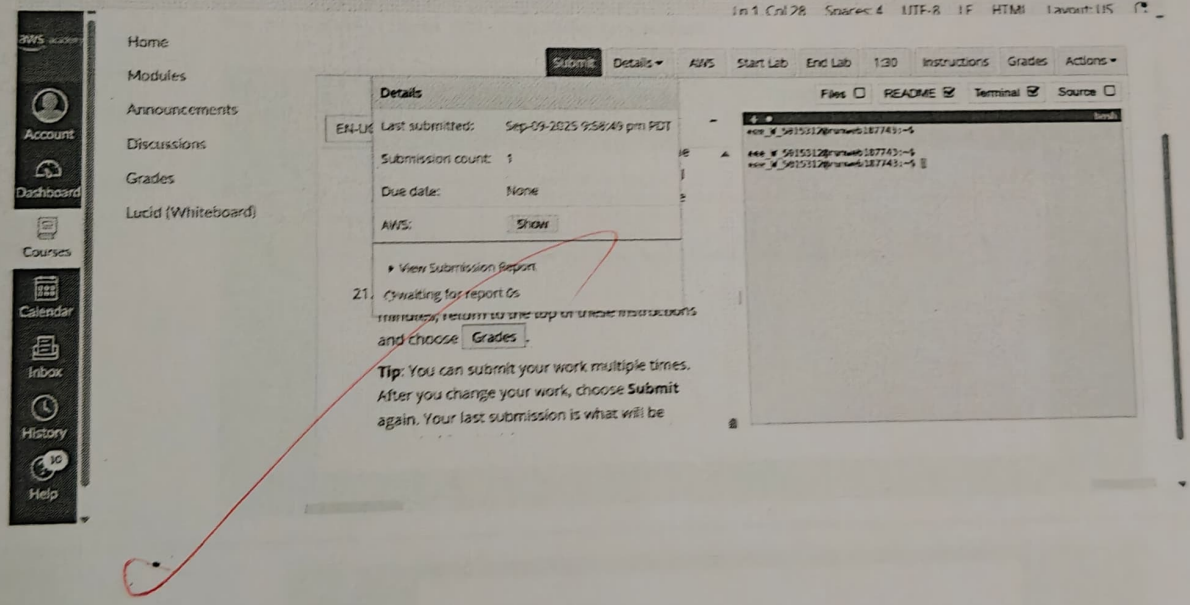
No Comments



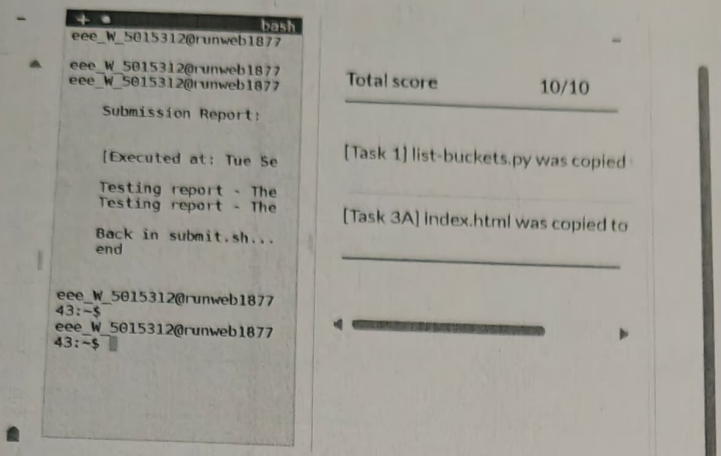
Wednesday, September 10, 2025

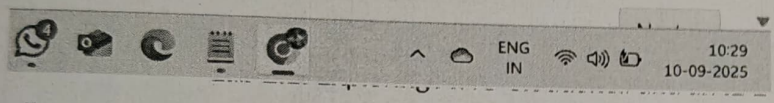


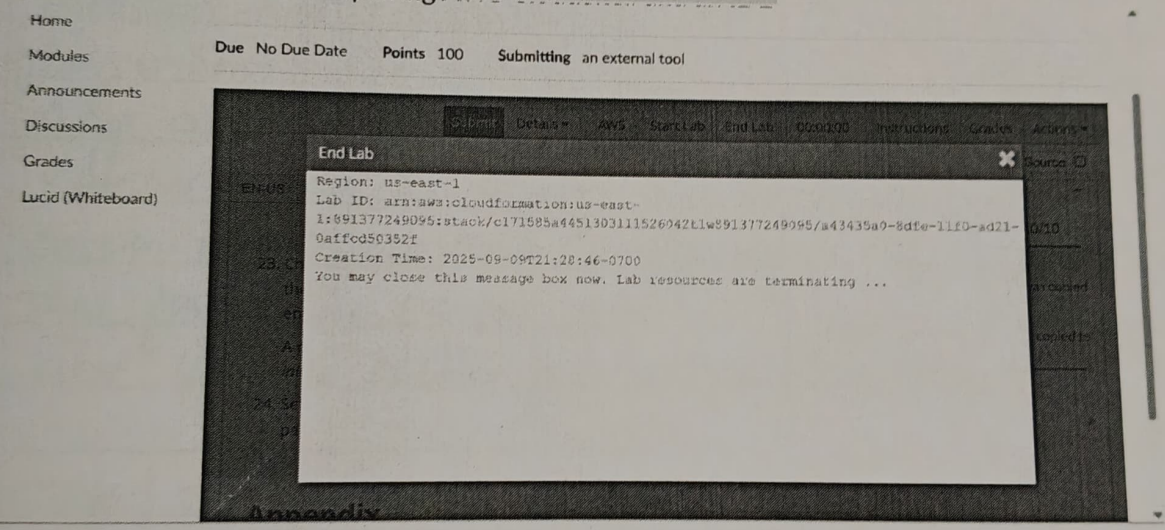
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

buckets.py .
download: s3://c171585a44513031115260421w8913772490-samplebucket-gomovsqsnwkm/list-buckets.py to ./list-buckets.py
[ec2-user@ip-10-0-1-124 environment]$ python3 list-buckets.py
c171585a44513031115260421w8913772490-samplebucket-gomovsqsnwkm
[ec2-user@ip-10-0-1-124 environment]$ aws s3 cp index.html s3://c171585a44513031115260421w8913772490-samplebucket-gomovsq
snwkm/index.html
upload: ./index.html to s3://c171585a44513031115260421w8913772490-samplebucket-gomovsqsnwkm/index.html
[ec2-user@ip-10-0-1-124 environment]$
```

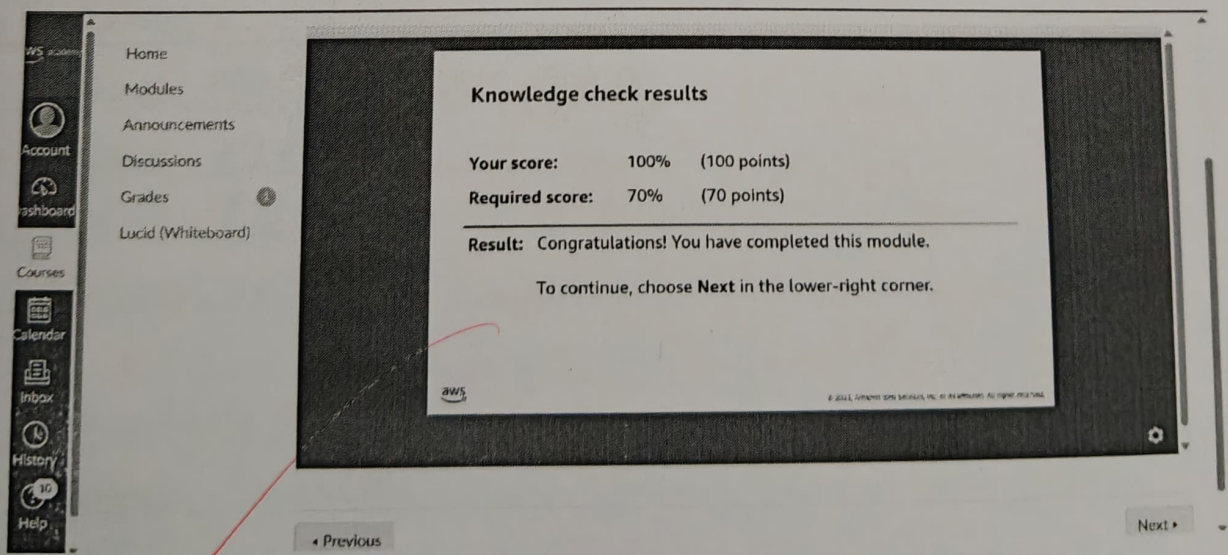


Wednesday, September 10, 2025









Wednesday, September 10, 2025

Working with Amazon S3

Objective

- To learn how to create, configure and manage Amazon S3 (simple storage service) buckets and objects, and to understand how to control access, store data efficiently and ensure durability in cloud-based storage

Theory

→ Amazon S3 is an object storage service that provides industry-leading scalability, data availability, security and performance. It allows developers to store and retrieve any amount of data from anywhere on the web. S3 stores data as objects within buckets each having unique access permissions and configurations. S3 also integrates with IAM for access control and supports versioning, encryption and lifecycle policies for cost optimization and security.

Tools / services used

- Amazon S3
- AWS Management Console
- AWS CLI
- IAM (Identity and Access Management.)
- Bucket policy / Access Control List (ACL)

Procedure / Steps

1. Logged into the AWS Academy Learner Lab environment.

. Navigated to the Amazon S3 service from the AWS Management Console.
. Created a new S3 bucket with a unique global name
. Configure bucket versioning and public access settings.
. Uploaded multiple files (text/images) to the bucket.
. Used the AWS CLI (aws s3 cp, aws s3 ls) to interact with the bucket.
7. Modified bucket policies and access permissions to allow/deny public access.
8. Tested file retrieval using the object URL.
9. Verified bucket configuration and ensure secure access settings

Output / Result

- Successfully created and configured an Amazon S3 bucket.
- Uploaded and accessed multiple objects using the AWS Console and CLI
- Applied and tested access control policies.
- Completed Knowledge check

Conclusion

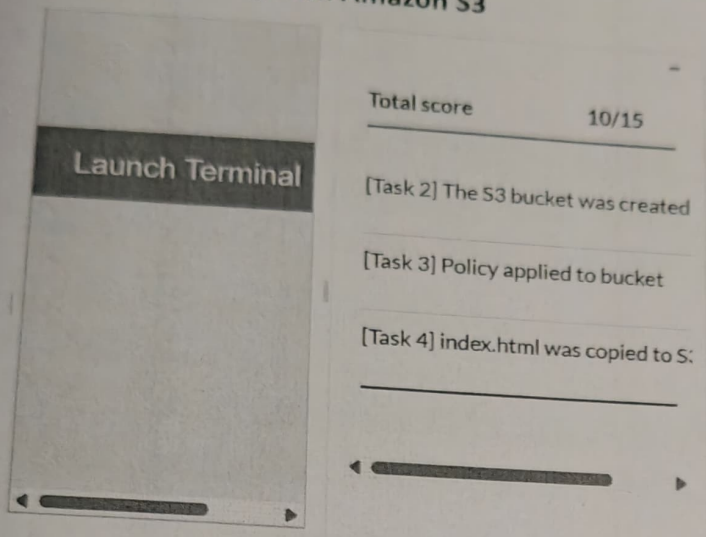
→ This lab enhanced my understanding of cloud storage solutions using Amazon S3. I learned how to create and manage buckets, control permissions and securely store data in the cloud. The lab is also provided practical exposure to managing S3 through both AWS Management Console and AWS CLI.

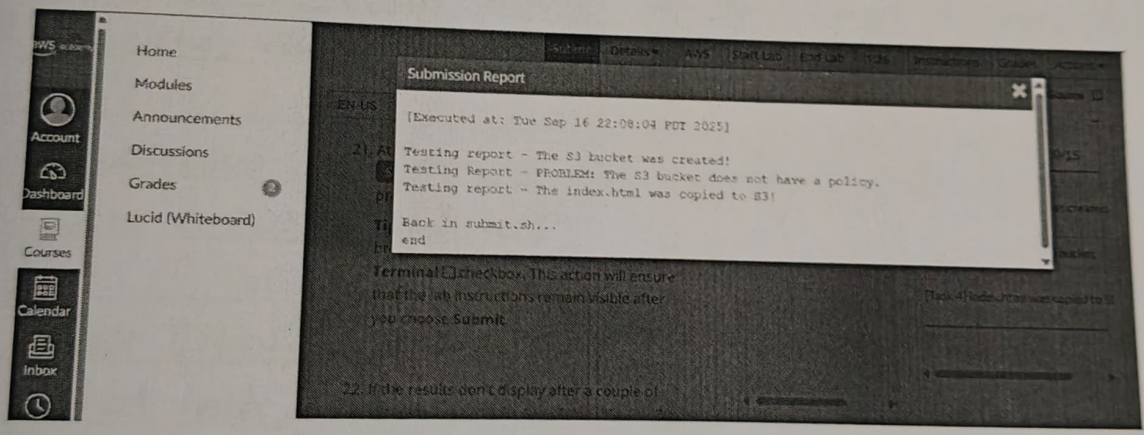
12/11/25

Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 3.1: Working with Amazon S3





Module 3 knowledge check results

Your score:	90%	(90 points)
Required score:	70%	(70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Wednesday, 17 September 2025

Securing Access to cloud Resources

Aim

To understand how to secure access to AWS resource by implementing Identity and Access Management (IAM) Principles and following the AWS shared Responsibility Model.

Theory

In cloud environment, security is a shared responsibility between AWS and the Customer

- AWS Manages the security of the cloud (infrastructure, hardware, regions, data, centers)
- Customers Management the security in the cloud (data encryption, user access, Permissions)

AWS Identity and Access Management (IAM) is a key service that helps organizations securely control access to AWS resources. IAM allows you to:

- Create and manage users, groups and roles.
- Define policies to control permissions.
- Implement Multifactor authentication (MFA)
- Enable cross-account access and fine-grained authorization

Tools / Services Used

- AWS Identity and Access Management (IAM)
- AWS Management Console

Policy JSON Editor
Cross-Account Access Demonstration

Activities / Steps

1. Reviewed the Shared Responsibility Model and identified AWS Vs Customer responsibilities.
2. Studied IAM Components - Users, groups, roles and policies.
3. Observed the demonstration - Configuration Cross Account Access.
4. Viewed the Demonstration.
5. Explored the authorization using IAM Policies (Part 1 and Part 2)
6. Completed the knowledge check

Output / Results

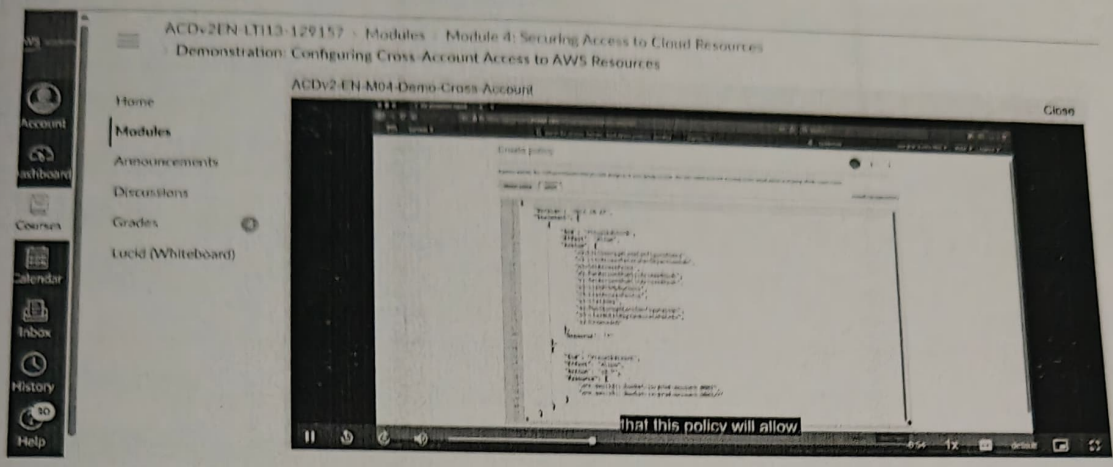
- Gained hands on understanding of how IAM secures AWS environments.
- Completed the knowledge check.

Conclusion

The Module helped me understand the importance of Access Control and Identity Management in cloud environment. I learned how AWS IAM ensures that only authorized users can access specific resources and how policies define granular permissions for secure cloud operations.

12/11/25

Module 4: Securing Access to Cloud Resources



Module 4 knowledge check results

Your score:	80%	(80 points)
Required score:	70%	(70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Monday, 08 October 2025

Working with Amazon DynamoDB

Objective

To learn how to create, configure and manage Amazon DynamoDB tables for building scalable, flexible and high-performance NoSQL database solutions in AWS

Theory

- Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability

It stores data in key-value and document formats, making it ideal for applications requiring high availability and low latency.

Key DynamoDB Concept include:

- Tables
- Items
- Attributes
- Primary key
- secondary Indexes
- Read/write Capacity Units (RCUs, WCUs)

Tools/services Used

- Amazon DynamoDB
- AWS Management Console
- AWS CLI
- IAM for Access management

Procedure / steps

Logged into the Aws Academy Learner Lab environment.
Opened the Amazon DynamoDB from the Aws Console management.
Created a new table named StudentData with a partition key: StudentID
Configure read/write capacity
Added sample data
Queries and scanned the table to retrive items
Created a Global secondary steps.
Explored backup and restore.

Output

- Successfully created and configured a DynamoDB table.
- Performed CRUD operation
- Implemented Secondary Index
- Verified table performance.
- Completed knowledge check

Conclusion

→ This lab enhanced my understanding of how to develop NoSQL database solution on Aws using DynamoDB. I learned to design data models, manage indexes, configure capacity and use advanced features like streams and Backups.

12/11/25

Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Module 5: Developing Flexible NoSQL Solutions

Lab 5.1: Working with DynamoDB

Submission Report

```
[Executed at: Thu Oct 30 23:47:58 PDT 2025]

Testing report - The DynamoDB table was created!
Testing report - The batch load completed successfully!
Testing report - The GSI was added to the table!

Back in submit.sh...
end
```

prted, choose Yes

Lab 5.1: Working with DynamoDB

Due No Due Date **Points** 100 **Submitting** an external tool

Submit | Details | AWS | Start Lab | End Lab | 2:05 | Instructions | Grades | Actions

Files | README | Terminal | Source

EN-US

Submitting your work

65. At the top of these instructions, choose **Submit** to record your progress and when prompted, choose **Yes**.
66. If the results don't display after a couple of minutes, return to the top of these instructions

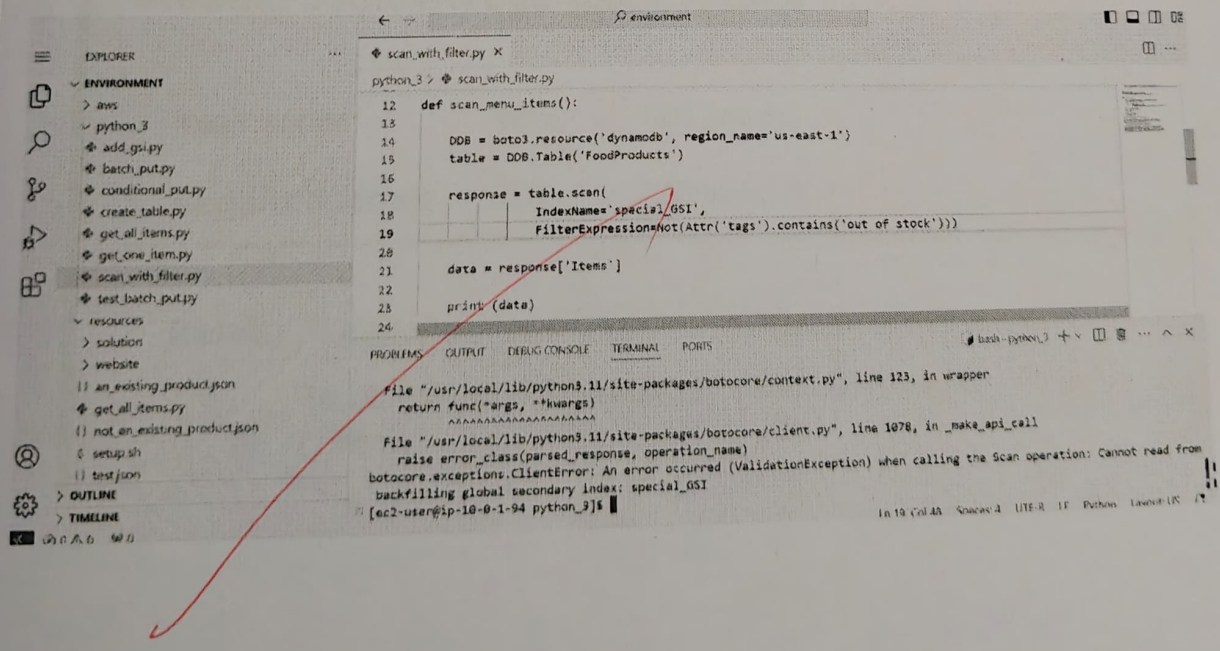
Launch Terminal

Total score 15/15

[Task 2] DynamoDB table created

[Task 5] Batch load completed

[Task 7] GSI added to table



Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Completed · Items returned: **27** · Items scanned: **27** · Efficiency: **100%** · RCUs consumed: **0.5**

Table: FoodProducts - Items returned (27)

Actions ▼ | Create item

Scan started on October 31, 2025, 12:15:32

1

product_name *(String)*	description	price_in_cents
blueberry jelly doughnut	A doughnut w...	295
vanilla glazed doughnut	so good!	295
boston cream doughnut	Boston's favor...	295

Module 5 Knowledge Check

Due No Due Date **Points** 100 **Submitting** an external tool

Module 5 knowledge check results

Your score: 90% (90 points)

Required score: 70% (70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Developing REST APIs with API Gateway

Objective

To learn how to create, configure deploy and secure REST APIs using Amazon API Gateway integrated with AWS backend services.

Theory

Amazon API Gateway is a fully managed service that allows developers to create, publish, maintain, monitor and secure RESTful APIs at scale. It acts as a bridge between clients and backend services such as AWS Lambda, DynamoDB or EC2.

Tools / Services Used:

- Amazon API Gateway
- AWS Lambda
- AWS IAM
- Cloudwatch

Procedure / Steps.

1. Logged into the AWS Academy Learner Lab
2. Created a REST API using API Gateway
3. Defined resources and methods.
4. Integrated API with AWS Lambda function
5. Deployed the API to a stage
6. Tested API endpoint via the invoke URL
7. Configured IAM roles and policies for access control.

Output / Result

- Successfully built and deployed a REST API using API Gateway and Lambda
- Verified working endpoints and access permissions.
- Completed knowledge check

Conclusion

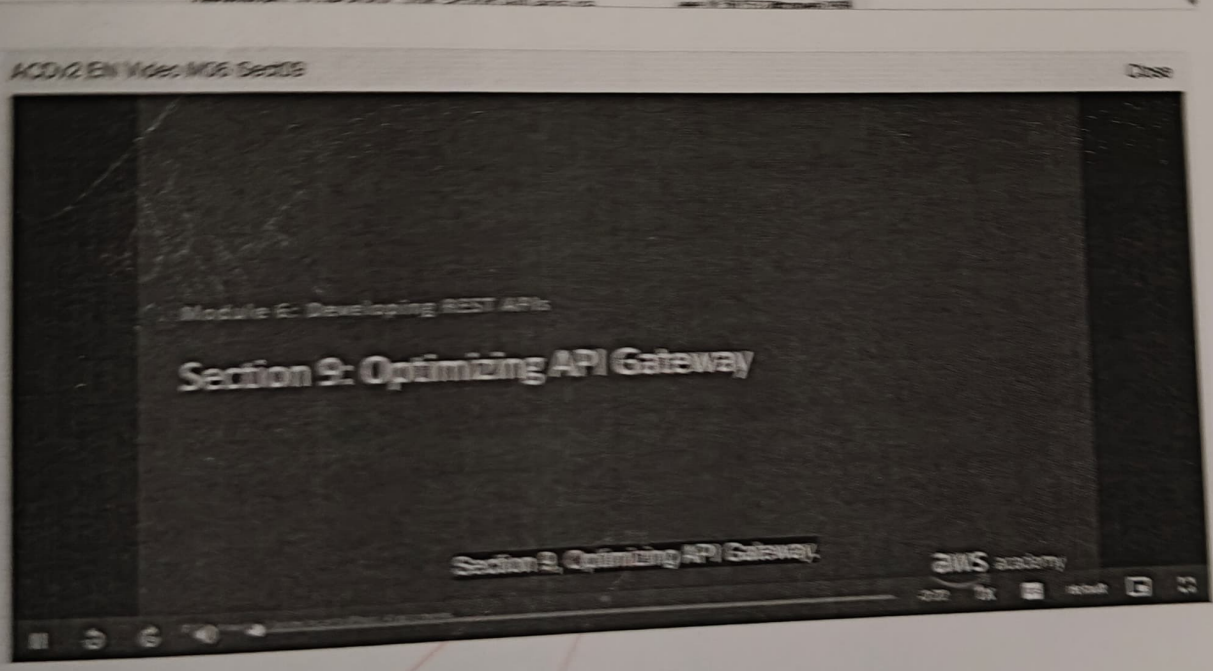
Learned how to develop and deploy secure, scalable REST APIs using API Gateway, integrate with AWS services and monitor performance through cloudwatch.

12/11/25

Module 6: Developing REST APIs

Lab 6.1: Developing REST APIs with API Gateway

Submitting your work



Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Due No Due Date Points 100 Submitting an external tool

Module 6 knowledge check results

Your score:	90%	(90 points)
Required score:	70%	(70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Creating Lambda functions Using the AWS SDK for Python

Objective

To learn how to create, configure and invoke AWS Lambda functions using the AWS SDK for Python in a serverless environment.

Theory

AWS Lambda is a serverless compute service that automatically runs code without managing servers. It execute code in response to events such as API calls, file uploads or database changes. Using Boto3 developers can Programmatically create and manage Lambda functions from Python applications.

Tools / services Used

- AWS Lambda
- AWS SDK for Python (Boto3)
- IAM Roles and policies
- Amazon cloudwatch

Procedure / steps

1. Logged into the AWS Academy Learner Lab
2. Created an IAM role with Lambda execution permissions.
3. Authored a Python functioning using Boto3 to Print a Custom message.
4. Deployed the function to AWS Lambda
5. Invoked the function manually and verified logs.

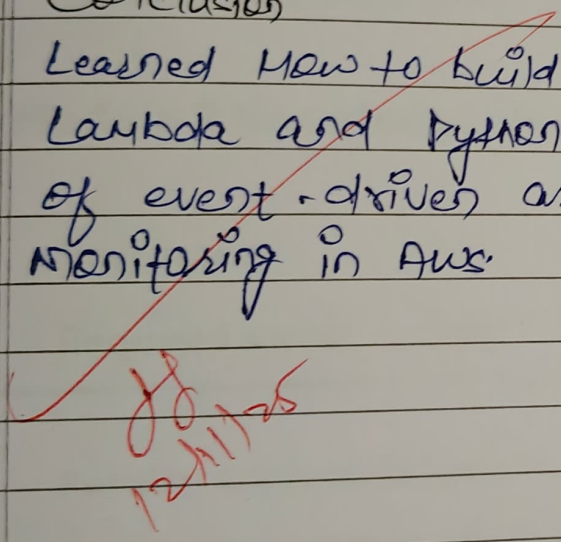
- Modified function configuration and redeployed updates.
- Tested error handling and event responses.

Output / Result

- Successfully created and executed a Lambda function using Python SDK.
- Observed function logs and performance metrics in CloudWatch.
- Completed knowledge check.

Conclusion

- Learned how to build serverless applications using AWS Lambda and Python SDK. This module improved my understanding of event-driven architecture, function deployment and monitoring in AWS.



Software Engineering Lab

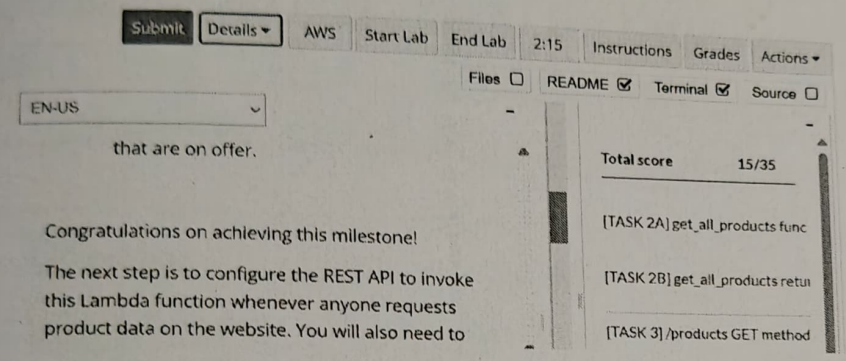
Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 7.1: Creating Lambda Functions Using the AWS SDK for Python

Lab 7.1: Creating Lambda Functions Using the AWS SDK for Python

Due No Due Date Points 100 Submitting an external tool

Submission
Nov 4 at 7:57am
Submission Details
Grade: 0 (100 pts possible)
Graded Anonymously: no
Comments:
No Comments



Module 7 knowledge check results

Your score: 100% (100 points)

Required score: 70% (70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Introducing Containers and Containes Services (Lab-8.1 & 8.2)

Objective

To understand Containerization using Docker and Learn how to deploy and manage containerized applications on Aws using Elastic Beanstalk and other managed Container services.

Theory

Containers Provide a lightweight and portable way to package applications and their dependencies, ensuring consistency across environments.

Docker is the most widely used Containes Platform that allows developers to build, ship and run applications quickly.

Aws supports Containers through services like

- Amazon Elastic Containes services (ECS)
- Elastic Kubernetes Services (EKS)
- Elastic Beanstalk, which automates deployment and scaling of Containerized apps.

3. Tools / services Used

- Docker
- Aws Elastic Beanstalk
- Amazon Ecs (Elastic Container Service)
- Aws management Console / cloudshell

Procedure / steps

Logged into the AWS Academy Learner Lab environment.
Created a Dockerfile for a sample web application.
Built and tested the Docker image locally to verify functionality.
Deployed the containerized application on Elastic Beanstalk as a managed service.
Observed automatic environment provisioning and scaling.
Explored ECS dashboard for understanding container orchestration and management.
Monitored deployment status and application performance metrics.
Completed the module knowledge check.

Output

- Successfully created and ran a Docker container for a web application.
- Deployed the containerized app using AWS Elastic Beanstalk.
- Understood how AWS manage container scaling, deployment and monitoring.

Conclusion

→ This module provided hands on experience in containerization and cloud deployment. I learned how to migrate applications to Docker, deploy them on AWS managed services like Elastic Beanstalk and manage their lifecycle efficiently. This knowledge is essential for modern DevOps and Micro-services based architecture.

Lab 8.1: Migrating a Web Application to Docker Containers

Lab 8.1: Migrating a Web Application to Docker Containers

Due No Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab End Lab 1:13 Instructions Grades Actions

Submission Report

```
[Executed at: Mon Nov 10 5:29:48 BST 2025]

Testing report - The suppliers database contains at least one supplier!
Testing report - The security group was configured correctly!
Testing report - The Dockerfile for MySQL was found!
Testing report - Evidence was found of a successful launch of a MySQL contain
Testing report - The two containers are running as expected.
Testing report - The image node-app was pushed to Amazon ECR!
```

Submit Details AWS Start Lab End Lab 1:13 Instructions Grades Actions

Files README Terminal Source

bash

```
    Found key for Cloud9.
    Result of query for mysql dockerfile: CompletedProcess(args=['ssh',
 '-i', 'c9key.pem', '-o StrictHostKeyChecking=no', 'ec2-user@98.95.134.
71', 'find /home/ec2-user/environment/containers/mysql -name Dockerfile
 |wc -l'], returncode=0, stdout=b'1\n', stderr=b"Warning: Permanently a
dded '98.95.134.71' (ECDSA) to the list of known hosts.\r\n")
    Does mysql Dockerfile exist? (1 = yes, 0 = no): 1
    Excellent! Found the mysql dockerfile.
    Does Docker log show that mysql started in container? (1+ = yes, 0
= no): 2
    Excellent! Found evidence in logs of mysql container starting succe
ssfully.
    ********** TASK 5 **********
    Was mysql_1 container found running? (1=yes, 0=no): 1
    Was node_app_1 container found running? (1=yes, 0=no): 1
    Excellent! Found evidence of mysql container and node app container
both running.
    ********** TASK 6 **********
    Back in submit.sh...
```

Total score 30/30

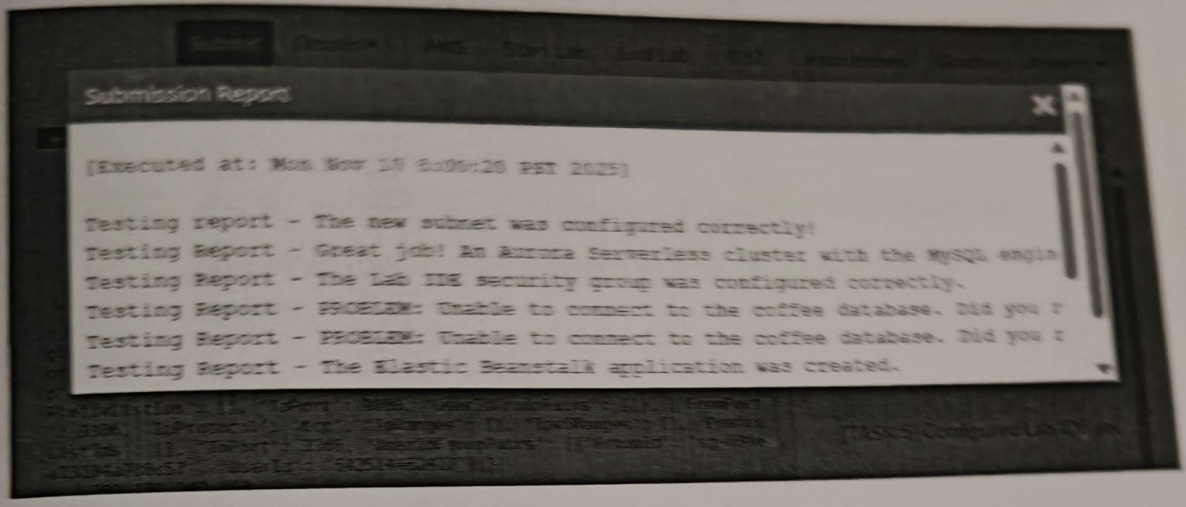
[TASK 2] Created new supplier

[TASK 3] Configured the securit

[TASK 4] Created MySQL Docke

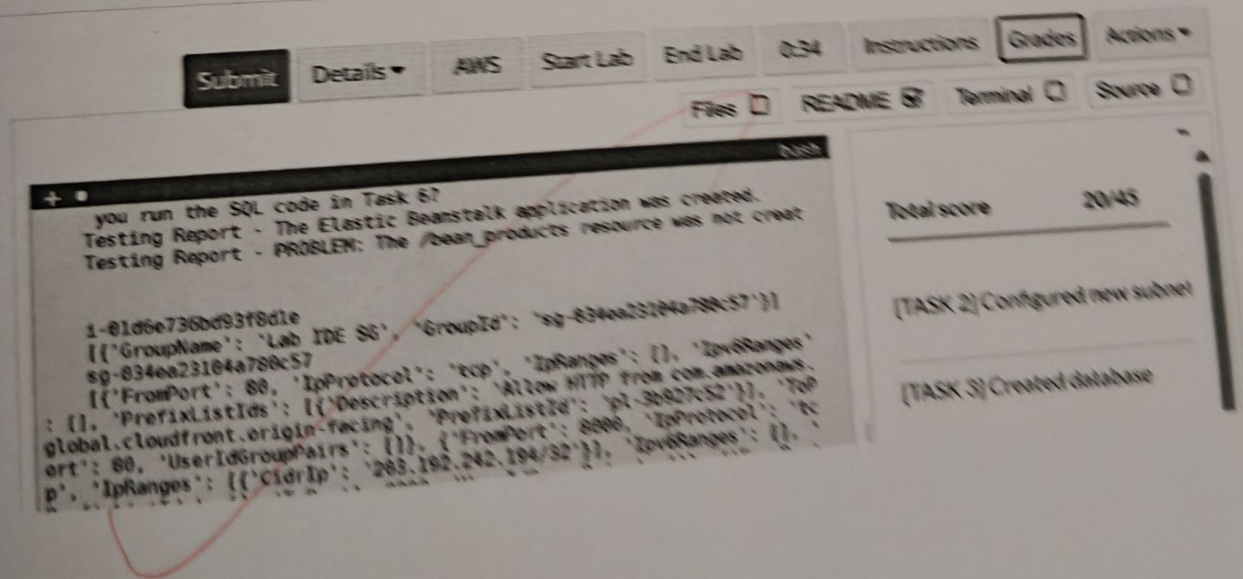
Lab 8.2: Running Containers on a Managed Service

Due No Due Date Points 100 Submitting an external tool



Lab 8.2: Running Containers on a Managed Service

Due No Due Date Points 100 Submitting an external tool



Module 8 knowledge check results

Your score:	100%	(100 points)
Required score:	70%	(70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Caching Information for Scalability
(Lab-9.1 & 9.2)

Objective
To understand how caching improves applications and scalability using Amazon ElastiCache and Amazon Cloudfront, and to implement caching strategies for faster data delivery and reduced latency

Theory
Caching stores frequently accessed data in temporary storage to reduce database load and response time.

- Amazon ElastiCache is a managed in-memory caching service supporting Redis and Memcached engines, ideal for application level caching.

- Amazon Cloudfront is a global Content Delivery Network (CDN) that caches content closer to users for low-latency access and enhanced security.
Together, these services improve application scalability, performance and user experience.

Tools / service used
- Amazon ElastiCache (Redis / Memcached)
- Amazon Cloudfront
- Amazon EC2 (for application testing)
- AWS management Console

Procedure / steps

1. Logged into the AWS Academy Learner Lab.
2. Created an ElastiCache cluster and connected it with an application hosted on EC2.
3. Configured caching for frequently queried data and tested response time improvements.
4. Set up a CloudFront distribution to cache web content from an S3 bucket or web server.
5. Configured origin settings
6. Tested content delivery through cloudfront URL
7. Verified faster response time
8. Completed knowledge check.

Output

- Successfully implemented caching with ElastiCache for database queries.
- Configured cloudfront as a CDN for global content delivery.
- Verified reduced latency and improved application performance.
-

Conclusion

→ This module provided practical understanding of caching strategies using AWS services. I learned how ElastiCache accelerates data retrieval at the application layer, and how cloudfront improves content delivery speed globally. These caching techniques are essential for building scalable, high-performance cloud applications.

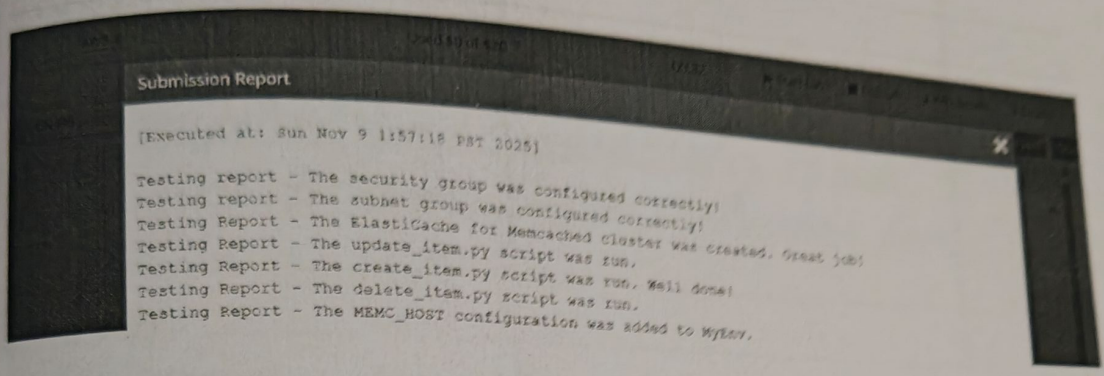
12/11/25

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 9.1: Caching Application Data with ElastiCache

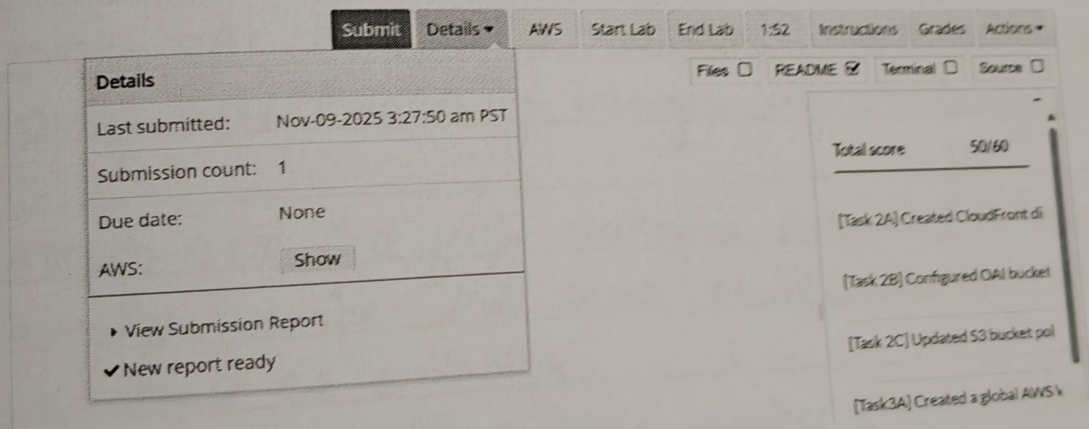
Lab 9.1: Caching Application Data with ElastiCache

Due No Due Date Points 100 Submitting an external tool



Lab 9.2: Implementing CloudFront for Caching and Application Security

Due No Due Date Points 100 Submitting an external tool



Module 9 knowledge check results

Your score:	90%	(90 points)
Required score:	70%	(70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Implementing a Messaging System Using Amazon SNS and Amazon SQS

Objective

To learn how to design and implement asynchronous communication between distributed systems using Amazon Simple Queue Services (SQS) and Amazon Simple Notification Service (SNS)

Theory

Asynchronous Messaging allows application to communicate and process tasks independently without waiting for immediate responses.

Amazon SQS provides a reliable, scalable queue system for storing and processing messages asynchronously.

Amazon SNS is a fully managed pub/sub (publish-subscribe) messaging service that sends notifications to multiple subscribers or endpoints.

→ Combined these services enable decoupled, scalable and fault-tolerant application architectures.

Tools / services Used

- Amazon SQS (Simple Queue Service)
- Amazon SNS (Simple Notification Service)
- AWS Management Console
- AWS CLI

Procedure / steps

Logged into the AWS Academy Learner Lab.
Created a standard SQS queue for message storage.
Created an SNS topic and subscribed an SQS queue to it.
Configured permissions and access policies for SNS-SQS integration.
Published sample messages to the SNS topic.
Verified message delivery to the SQS queue and tested message retrieval.
Completed knowledge check.

Output

Successfully implemented an SNS-SQS integrated messaging system.
Verified message delivery and queue processing.
Demonstrated asynchronous communication between components.

Conclusion

This module helped me understand event-driven and decoupled application design using AWS messaging services. I learned how SQS manages messages queues reliably and how SNS distributes notification efficiently across multiple subscribers.

Lab 10.1: Implementing a Messaging System Using Amazon SNS and Amazon SQS

Lab 10.1: Implementing a Messaging System Using Amazon SNS and Amazon SQS

Due No Due Date **Points** 100 **Submitting** an external tool

Submit Details AWS Start Lab End Lab 1:43

Submission Report

```
[Executed at: Sun Nov 9 5:13:29 PST 2025]

Testing report - Great job creating the DeadLetterQueue.fifo queue!
Testing report - The policy for DeadLetterQueue.fifo was updated.
Testing report - The updated_beans.fifo queue was found. Great job!
Testing Report - PROBLEM: The updated_beans.fifo policy was not updated. Review th
Testing report - Great work! The topic named updated_beans_sns.fifo was created.
Testing report - The policy for the updated_beans_sns.fifo topic was updated.
```

Files README Terminal Source

```
bash
eee_W_5015312@runweb1950
```

Total score 35/40

[Task 2A] Created DeadLetterQu

[Task 2B] Secured DeadLetterQu

Module 10 knowledge check results

Your score: 80% (80 points)

Required score: 70% (70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Orchestrating serverless functions with step functions

Objective

To learn how to coordinate and manage multiple serverless functions using AWS step functions to build reliable automated workflows.

Theory

- AWS step functions is a serverless orchestration services that helps automate multi-step processes by combining AWS services like Lambda, SNS or SQS into a single, visual workflow.

It use state machine to define tasks, sequences and conditions. Each state represents a step - such as invoking a Lambda function or making a decision - allowing developers to manage complex application logic with reliability and error handling.

Tools / services used

- AWS step function
- AWS Lambda
- Amazon cloudwatch
- AWS management console

Procedure / steps

1. Logged into the AWS Academy Learner Lab
2. Opened AWS step functions and created a new state machine.
3. Defined workflow states and transitions using the Amazon States Language (ASL) - JSON based language

Integrated Lambda functions to perform tasks within the workflow.
Executed and tested the workflow to verify state transitions.
Monitored workflow execution and logs in cloudwatch.
Completed the knowledge check.

Output
Successfully created and executed a step functions state machine.
Verified function coordination and error handling.
Observed workflow execution flow and logs.

Conclusion
This module provided practical experience in workflow orchestration using Aws step functions. I learned how to link multiple serverless functions into a coordinated sequence, improving automation, reliability and visibility in distributed systems.

JJ
12/11/25

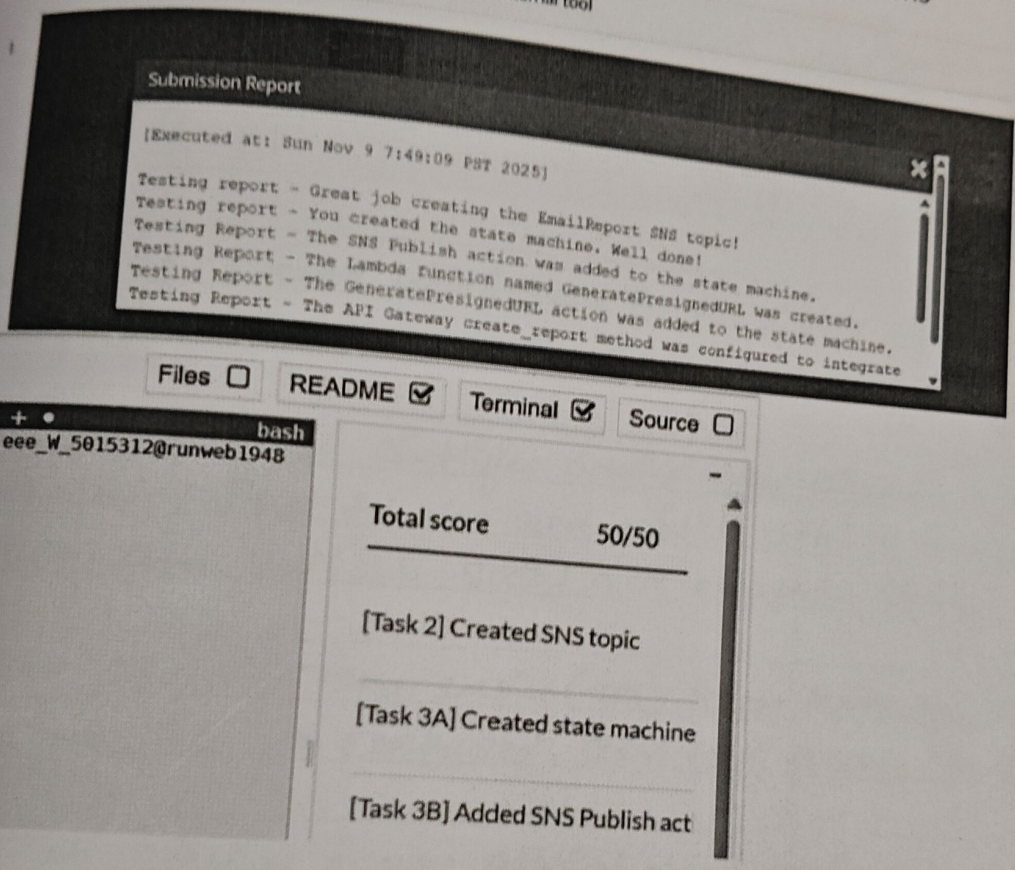
Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 11.1: Orchestrating Serverless Functions with Step Functions

Lab 11.1: Orchestrating Serverless Functions with Step Functions

Due No Due Date **Points** 100 **Submitting** an external tool



Module 11 knowledge check results

Your score: 90% (90 points)

Required score: 70% (70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Implementing Application Authentication using Amazon Cognito

Objective

→ To understand and implement secure user authentication in cloud based application using Amazon Cognito and Aws Security Token Services (STS)

Theory

→ Building secure applications in the cloud requires managing user identities, authentication and access control.

- Amazon Cognito simplifies adding user signup, sign in and access control to web and mobile apps. It integrates with IAM and supports social and enterprise identity providers.

- Aws Security Token service (STS) issues temporary security credentials for controlled, short term access to Aws resources.

These services ensure applications remain secure, scalable and compliant.

Tools / services used

- Amazon Cognito (user pools & identity pools)
- Aws Security Token Services (STS)
- Aws Management Console
- IAM Roles and policies

Procedure / steps

1. Logged into the Aws Academy Learner Lab.

created an Amazon Cognito User Pool to manage user registration and Login.
- Configured API clients and enabled authentication flow.
- Integrated AWS STS to issue temporary credentials for authorized users.
- Linked Cognito authentication with an application
- Tested sign up, Login and token validation processes
- Completed knowledge check

Output / Result

- Successfully implemented user authentication with Amazon Cognito.
- Verified secure sign-up and sign-in functionality
- Observed temporary credential generation via AWS STS.

Conclusion

- This module provided ~~provided~~ practical knowledge of securing AWS applications using Cognito and STS. I learned how to managed user identities, enable secure acess and integrate authentication features into real-world cloud applications

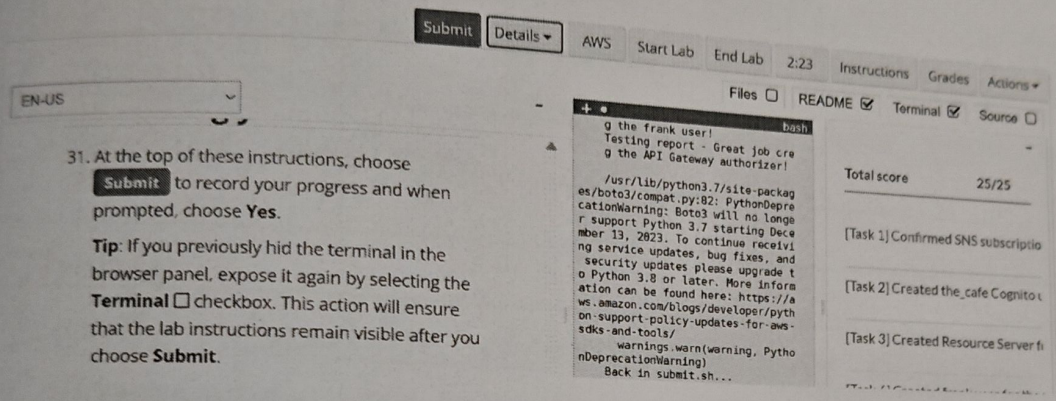
Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 12.1: Implementing Application Authentication Using Amazon Cognito

Lab 12.1: Implementing Application Authentication Using Amazon Cognito

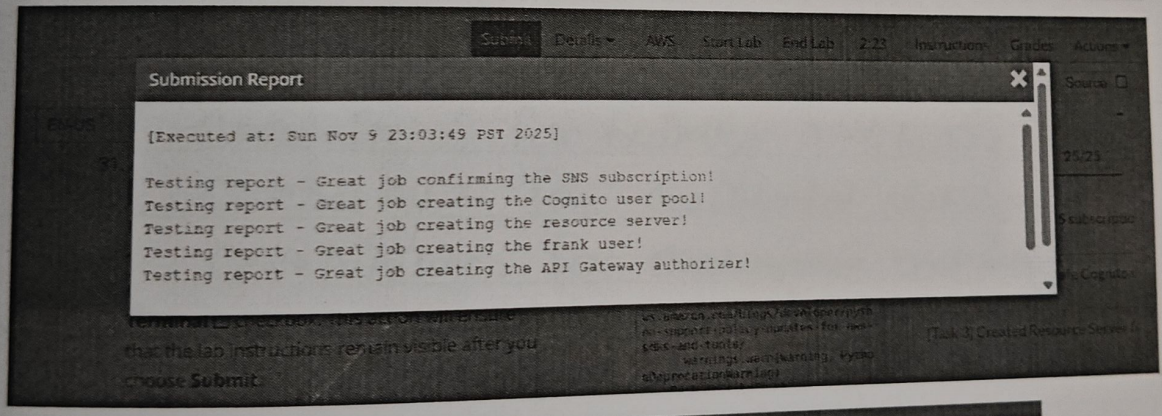
Due No Due Date Points 100 Submitting an external tool



31. At the top of these instructions, choose **Submit** to record your progress and when prompted, choose **Yes**.

Tip: If you previously hid the terminal in the browser panel, expose it again by selecting the **Terminal** ☐ checkbox. This action will ensure that the lab instructions remain visible after you choose **Submit**.

Due No Due Date Points 100 Submitting an external tool



Module 12 knowledge check results

Your score: 100% (100 points)

Required score: 70% (70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

Automating Application Deployment Using a CI/CD Pipeline.

Objective

→ To learn how to automate application deployment using AWS Continuous integration and continuous delivery (CI/CD) services, ensuring faster and more reliable software delivery.

Theory

→ DevOps Combine development and operations to improve Collaboration, automation and delivery speed AWS provides Managed DevOps tools such as

- AWS CodeCommit - for source code management.
- AWS CodeBuild - for building and testing applications.
- AWS CodeDeploy - for automated deployment.
- AWS CodePipeline - for ~~Orchestration~~ orchestrating the entire CI/CD workflow

Together, they help implement a complete CI/CD lifecycle for modern cloud applications

Tools / services Used

- AWS CodePipeline
- AWS CodeCommit
- AWS CodeBuild
- AWS CodeDeploy
- AWS CloudFormation / AWS SAM

Automating Application Deployment Using a CI/CD Pipeline.

Objective

→ To learn how to automate application deployment using AWS Continuous integration and continuous Delivery (CI/CD) services, ensuring faster and more reliable software delivery.

Theory

→ DevOps Combine development and operations to improve Collaboration, automation and delivery speed AWS provides Managed DevOps tools such as

- AWS CodeCommit - for source code management.
- AWS CodeBuild - for building and testing applications.
- AWS CodeDeploy - for automated deployment.
- AWS CodePipeline - for ~~Orchestration~~ orchestrating the entire CI/CD workflow

Together, they help implement a complete CI/CD lifecycle for modern cloud applications

Tools / services Used

- AWS CodePipeline
- AWS CodeCommit
- AWS CodeBuild
- AWS CodeDeploy
- AWS CloudFormation / AWS SAM

Procedure / steps

Logged into the AWS Academy Learner Lab.
Created a CodeCommit repository and pushed sample application code.
Configured a CodeBuild project to compile and test the code
Created a CodeDeploy application to automate deployment to EC2 or Lambda.
Set up a CodePipeline to integrate all stage (Source → Build → Deploy)
Deploy the application automatically through the pipeline.
Monitored build and deployment status in the AWS Console.
Completed the Module knowledge check.

Output

Successfully built and deployed an automated CI/CD pipeline using AWS DevOps tools.
Verified end-to-end automation from Code commit to deployment.

Conclusion

This Module provided hands-on experience in implementing DevOps automation using AWS CI/CD services. I learned how to streamline software delivery, improve deployment reliability and maintain Continuous integration practices for cloud-based applications.

12/11/25

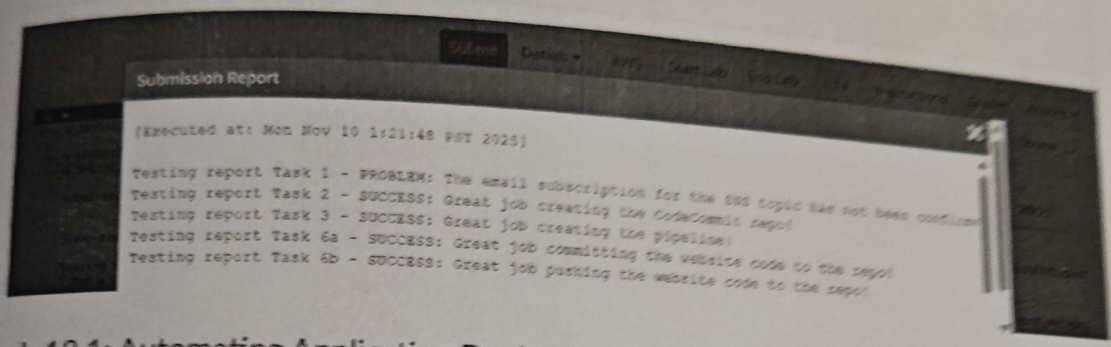
Software Engineering Lab

Khurram Rashid, A70405223016, ASET Div- C, Sem- 5

Lab 13.1: Automating Application Deployment Using a CI/CD Pipeline

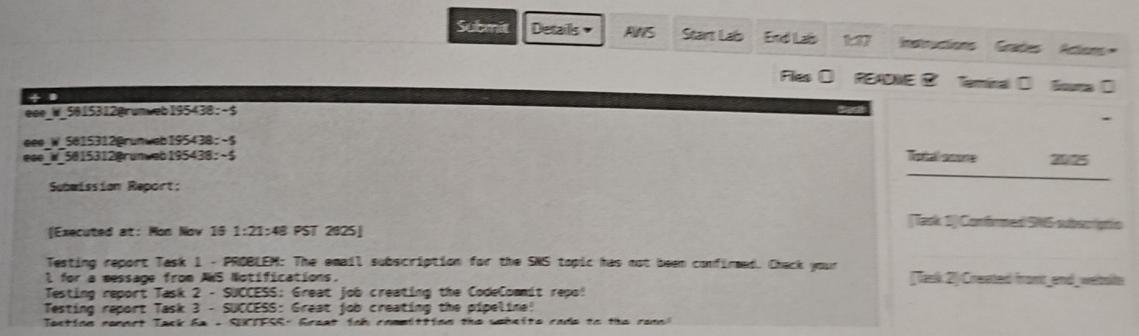
Lab 13.1: Automating Application Deployment Using a CI/CD Pipeline

Due No Due Date Points 100 Submitting an external tool



Lab 13.1: Automating Application Deployment Using a CI/CD Pipeline

Due No Due Date Points 100 Submitting an external tool



Module 13 knowledge check results

Your score:	80%	(80 points)
Required score:	70%	(70 points)

Result: Congratulations! You have completed this module.

To continue, choose **Next** in the lower-right corner.

aws academy

Khurram Rashid

Certificate of Completion for

AWS Academy Graduate - Cloud Developing - Training Badge

Course hours completed

40 hours

Issued on

11/09/2025

Digital badge

https://www.credly.com/go/yNXPbnvp