



# **BS-Software Engineering**

## **Software Development & Construction**

### **AWS Architecture Diagram**

**Instructor: Dr Zunnurain Hussain**

**Teaching Assistant: Umair Makhdoom**

**Project Title: Cloud Based E – Commerce Platform**

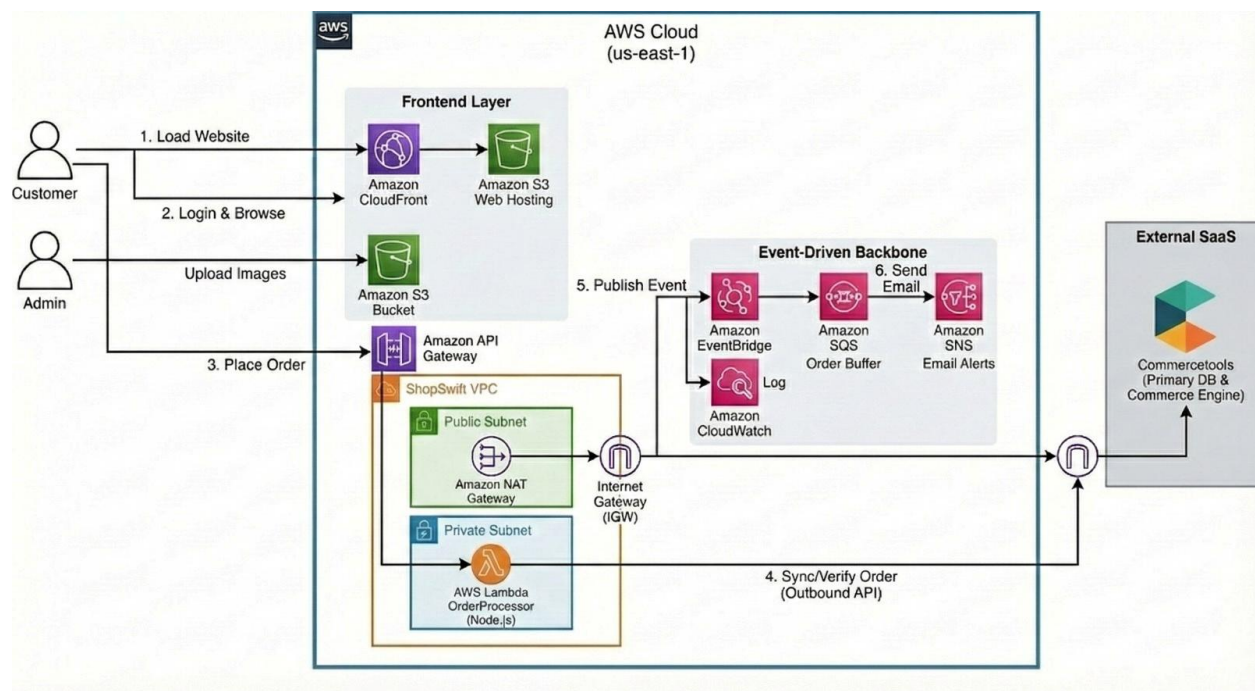


#### **Group Members:**

- Muhammad Numan Tahir – BSSE23069
- Muhammad Zubair Akhter – BSSE23079

## AWS Architecture Diagram:

A professional, AWS-standard architecture diagram was designed using correct AWS icons and clear data flows. The diagram accurately represents frontend delivery through Amazon S3 and CloudFront, backend processing via API Gateway and Lambda, secure networking using VPC subnets, and event-driven integration with EventBridge, SQS, SNS, and external Commercetools services. All components and interactions are logically connected and easy to understand.



## Implementation Steps:

1. An **Amazon S3 bucket** was created to store all website-related static resources, including HTML files, CSS, JavaScript, and product images. S3 was chosen due to its high durability, scalability, and cost-effective storage. This allowed the website content to be securely stored and accessed without the need for managing physical servers.
2. **Amazon CloudFront** was configured with the S3 bucket as its origin to deliver website content to users with low latency. CloudFront uses a global

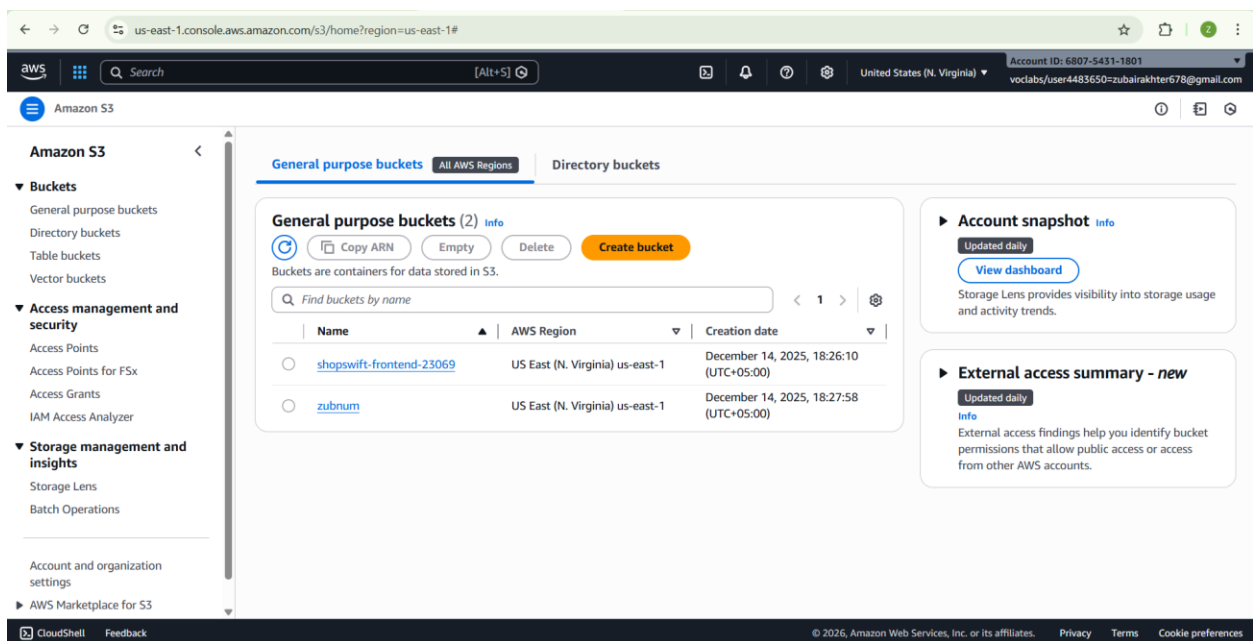
network of edge locations to cache content closer to users, which significantly improved website loading speed and reduced the load on the origin server.

3. A **Virtual Private Cloud (VPC)** was set up with both public and private subnets to ensure network security and isolation. Public subnets were used for services that needed internet access, while private subnets were used for backend services. This separation improved security by limiting direct access to critical components.
4. **Amazon API Gateway** was deployed to act as the main entry point for all backend requests from the frontend. It handled request routing, authentication, and traffic management. API Gateway enabled secure and scalable communication between the frontend application and backend services.
5. Backend business logic was implemented using **AWS Lambda (or Amazon ECS)** within private subnets. Lambda allowed serverless execution of backend code without managing servers, automatically scaling based on demand. This approach reduced operational overhead and ensured high availability during traffic spikes.
6. **Commercetools APIs** were integrated to manage core e-commerce functionalities such as product catalog, shopping carts, order processing, and payments. Using Commercetools provided a headless commerce solution that allowed ShopSwift to scale and update features independently from the frontend.
7. Customer session data and profile information were stored in **Amazon DynamoDB**, a fully managed NoSQL database service. DynamoDB was selected for its low latency, automatic scaling, and high availability, ensuring fast access to user data even during high traffic periods.
8. **Amazon EventBridge** was configured to publish events whenever a new order was placed. This enabled an event-driven architecture where different services could react to order events independently, improving system flexibility and decoupling services.

9. **Amazon SQS** was used to buffer order messages to ensure reliable processing, even during high load. At the same time, **Amazon SNS** was used to send notifications such as order confirmations to customers. This combination ensured fault tolerance and reliable message delivery.
10. Finally, **Amazon CloudWatch** was enabled to monitor system performance, collect logs, and track metrics across all services. CloudWatch provided visibility into application health, allowed real-time monitoring, and helped in quickly identifying and resolving issues.

## Screenshots:

### 1) S3:



### 2) VPC

**aws** [Search] [Alt+S] United States (N. Virginia) Account ID: 6807-5431-1801 voclabs/user4483650-zubairakhter678@gmail.com

**VPC** > Your VPCs

**VPC dashboard** < AWS Global View [i] Filter by VPC [v]

▼ **Virtual private cloud**

- Your VPCs
- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- NAT gateways
- Peering connections
- Route servers

▼ **Security**

- Network ACLs
- Security groups

**Your VPCs** VPCs VPC encryption controls

Find VPCs by attribute or tag

Last updated less than a minute ago [Actions] [Create VPC]

<input type="checkbox"/>	Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public...	IPv...
<input type="checkbox"/>	--	vpc-0336d178b2525f00b	Available	--	--	Off	172
<input type="checkbox"/>	shopswift-vpc	vpc-035ab7348922026af	Available	--	--	Off	10.0

Select a VPC above

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

### 3) Subnets:

**aws** [Search] [Alt+S] United States (N. Virginia) Account ID: 6807-5431-1801 voclabs/user4483650-zubairakhter678@gmail.com

**VPC** > Subnets

**VPC dashboard** < AWS Global View [i] Filter by VPC [v]

▼ **Virtual private cloud**

- Your VPCs
- Subnets**
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- NAT gateways
- Peering connections
- Route servers

▼ **Security**

- Network ACLs
- Security groups

**Subnets (8)** Info

Find subnets by attribute or tag

Last updated 1 minute ago [Actions] [Create subnet]

<input type="checkbox"/>	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
<input type="checkbox"/>	--	subnet-06dfacd0b7d87ce86	Available	vpc-0336d178b2525f00b	Off	172.31.16.0/2
<input type="checkbox"/>	--	subnet-082f6ff8311a1d275	Available	vpc-0336d178b2525f00b	Off	172.31.80.0/2
<input type="checkbox"/>	--	subnet-05d91bf1cf4acb16	Available	vpc-0336d178b2525f00b	Off	172.31.64.0/2
<input type="checkbox"/>	--	subnet-0f1dd9af40c3656f2	Available	vpc-0336d178b2525f00b	Off	172.31.0.0/20
<input type="checkbox"/>	--	subnet-08b474276dc0d1c5d	Available	vpc-0336d178b2525f00b	Off	172.31.32.0/2
<input type="checkbox"/>	shopswift-public-subnet	subnet-0d276448e41810603	Available	vpc-035ab7348922026af   sho...	Off	10.0.1.0/24

Select a subnet

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

### 4) Route Tables:

Account ID: 6807-5431-1801  
voclabs/user4483650=zubairakhter678@gmail.com

United States (N. Virginia)

VPC > Route tables

**Route tables (3)** Info

Find route tables by attribute or tag

Last updated 2 minutes ago

Actions Create route table

<input type="checkbox"/>	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input type="checkbox"/>	shopswift-public-rt	rtb-0be9e8903bc87090e	subnet-Od276448e41810...	-	No	vpc-035ab7348922026af   sho..
<input type="checkbox"/>	-	rtb-011d4911c1418df82	-	-	Yes	vpc-0336d178b2525f00b
<input type="checkbox"/>	-	rtb-0d2968c9cf7169e7	-	-	Yes	vpc-035ab7348922026af   sho..

Select a route table

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 5) Internet Gateway:

us-east-1.console.aws.amazon.com/vpconsole/home?region=us-east-1#igws:

Account ID: 6807-5431-1801  
voclabs/user4483650=zubairakhter678@gmail.com

United States (N. Virginia)

VPC > Internet gateways

**Internet gateways (2)** Info

Find internet gateways by attribute or tag

Actions Create internet gateway

<input type="checkbox"/>	Name	Internet gateway ID	State	VPC ID	Owner
<input type="checkbox"/>	shopswift-igw	igw-0148de7d8b79c673e	Attached	vpc-035ab7348922026af   shopswift-vpc	680754311801
<input type="checkbox"/>	-	igw-017c94f89af32c4ea	Attached	vpc-0336d178b2525f00b	680754311801

Select an internet gateway above

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 6) Security Groups:

Account ID: 6807-5431-1801  
voclabs/user4483650=zubairakhter678@gmail.com

United States (N. Virginia)

VPC > Security Groups

**Security Groups (3)** Info

Find security groups by attribute or tag

Actions Export security groups to CSV Create security group

<input type="checkbox"/>	Name	Security group ID	Security group name	VPC ID	Description
<input type="checkbox"/>	-	sg-04f7932cf77ae2e7c	default	vpc-0336d178b2525f00b	default VPC security
<input type="checkbox"/>	-	sg-0112749070323bb45	launch-wizard-1	vpc-035ab7348922026af	launch-wizard-1 cre
<input type="checkbox"/>	-	sg-006bb7cb3d49733f7	default	vpc-035ab7348922026af	default VPC security

## 7) Lambda

**Lambda > Functions**

Functions (6) Last fetched 1/4/2026, 12:46:14 PM

Search by attributes or search by keyword

Function name	Description	Runtime	Last modified
<a href="#">RedshiftEventSubscription</a>	Create Redshift event subscription to SNS Topic.	Python 3.10	4 weeks ago
<a href="#">MainMonitoringFunction</a>	-	Python 3.10	4 weeks ago
<a href="#">ModLabRole</a>	updates LabRole to allow it to assume itself	Python 3.10	4 weeks ago
<a href="#">RoleCreationFunction</a>	Create SLR if absent	Python 3.10	4 weeks ago
<a href="#">RedshiftOverwatch</a>	Deletes Redshift Cluster if the count is more than 2.	Python 3.10	4 weeks ago
<a href="#">ShopSwift-Backend</a>	-	Node.js 20.x	2 weeks ago

## 8) Tables (DynamoDB):

**DynamoDB > Tables**

Tables (1) Info Last updated January 4, 2026, 12:46 (UTC+5:00)

Find tables Filter by tag Any tag key Filter by tag value Any tag value

Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion protection	Favorite	Read c
<a href="#">ShopSwift-UserData</a>	Active	userid (S)	-	0 0		Off	☆	On-de

## 9) EC2 Instances:

**EC2 > Instances**

Instances (1) Info

Find Instance by attribute or tag (case-sensitive) All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
<a href="#">ShopSwift-We...</a>	i-0750f674c4792d2c7	Running	t3.micro	3/3 checks pass	View alarms +	us-east-1a	-

## 10) SNS

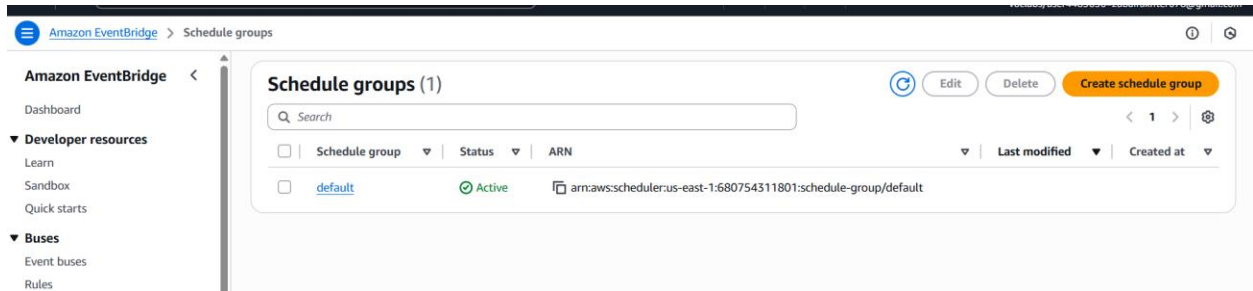
**Amazon SNS > Topics**

Topics (2)

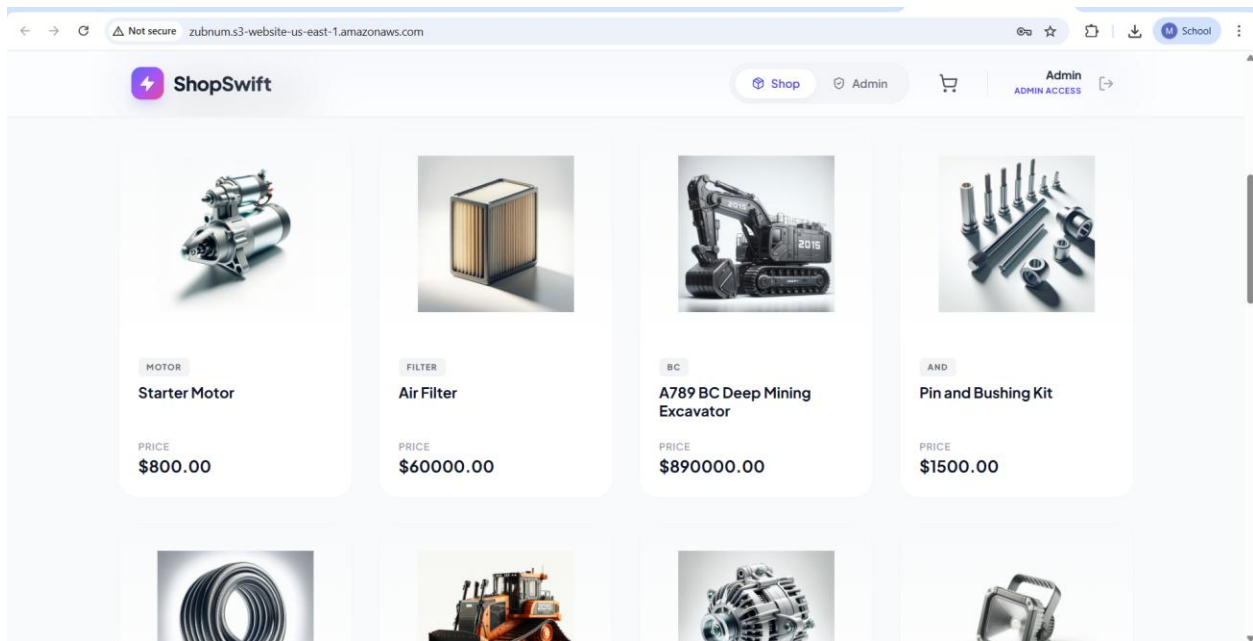
Search

Name	Type	ARN
<a href="#">RedshiftSNS</a>	Standard	arn:aws:sns:us-east-1:680754311801:RedshiftSNS
<a href="#">ShopSwift-Alerts</a>	Standard	arn:aws:sns:us-east-1:680754311801:ShopSwift-Alerts

## 11) Event Bridge:



## 12) Final Output



## Security & IAM:

Security best practices were followed throughout the system. IAM roles were configured using the principle of least privilege, VPC isolation was implemented with public and private subnets, and outbound access was controlled using a NAT Gateway. Logging and monitoring were enabled through CloudWatch, and data access was secured to ensure reliability and compliance.