

CCA ASSIGNMENT

Name :- Maanav Singh

SAP ID :- 500108304

Batch :- CCVT (B-7)

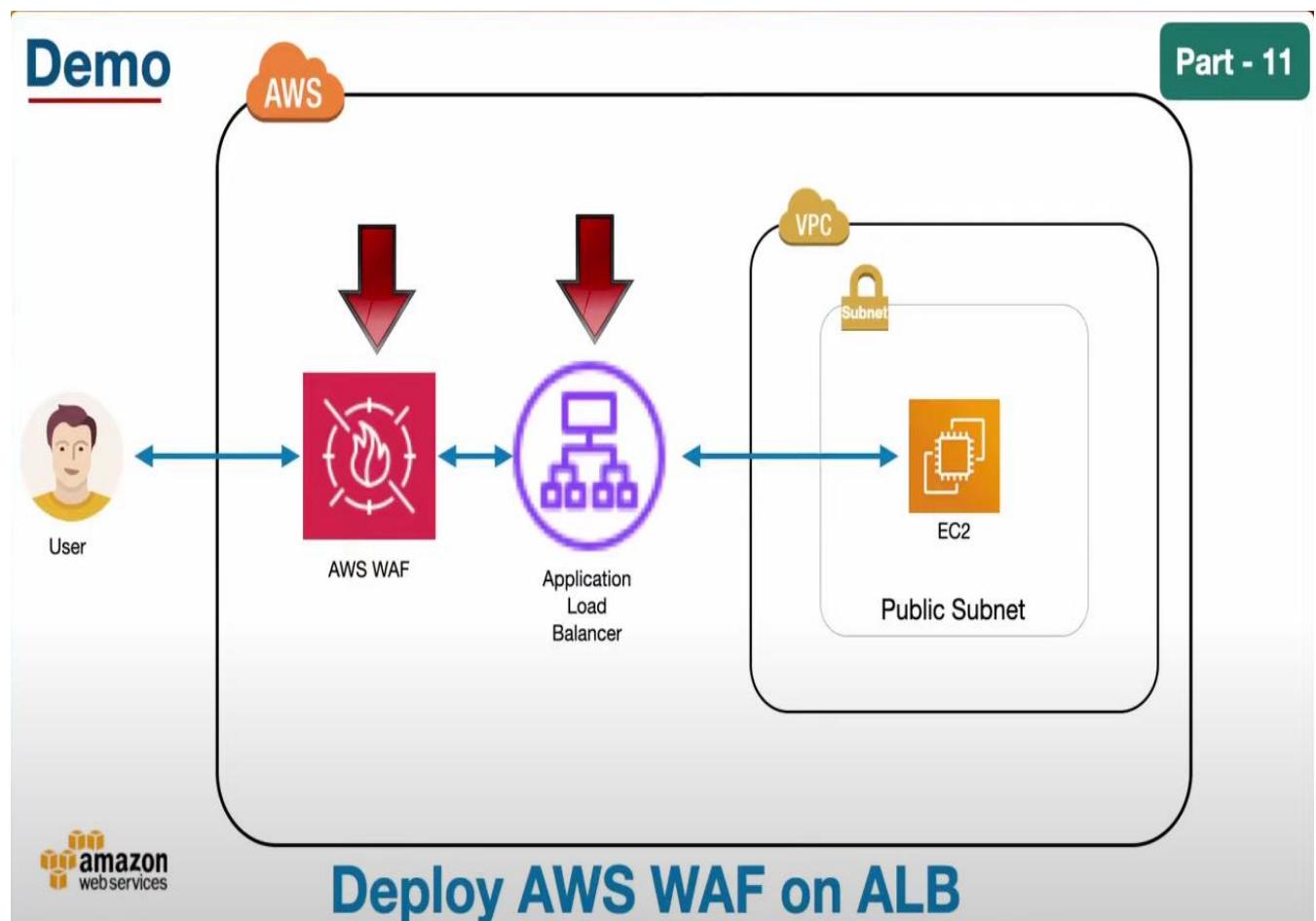
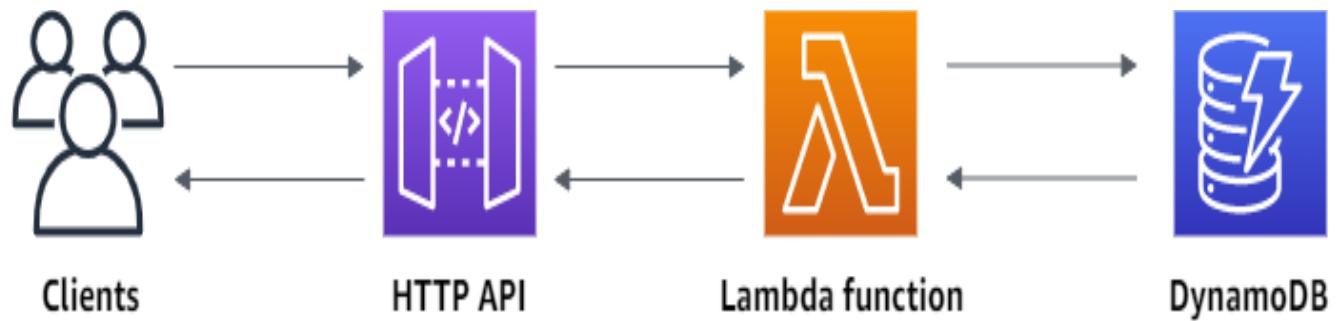
Enrollment no. :- R2142220977

Faculty :- Avita Katal

Q 1 - Design and implement a Software as a Service (SaaS) application. Provide a comprehensive plan covering the conceptualization, design, and development stages. Emphasize the unique value proposition, customer acquisition, user experience, multitenant design, pricing strategy and potential market impact.

Solution :-

- 1) Architectures Used in this Project



- 2) Create table in dynamoDB

The screenshot shows the AWS DynamoDB console interface. On the left, there is a navigation sidebar with options like Dashboard, Tables, Explore items, PartiQL editor, Backups, Exports to S3, Imports from S3, Integrations, Reserved capacity, and Settings. Below this is a section for DAX with Clusters, Subnet groups, Parameter groups, and Events. The main content area is titled 'Tables (1) Info' and shows a table named 'db-signup-table'. The table details are as follows:

Name	Status	Partition key	Sort key	Indexes	Deletion protection	Read capacity mode	Write capacity mode
db-signup-table	Active	uid (S)	-	0	Off	Provisioned (1)	Provisioned (1)

At the bottom of the main content area, there are tabs for Overview, Indexes, Monitor, Global tables, Backups, Exports and streams, and Per. The 'Overview' tab is selected. The 'General information' section shows the following details:

Partition key	Sort key	Capacity mode	Table status
uid (String)	-	Provisioned	Active
Alarms	Point-in-time recovery (PITR)	Resource-based policy	Not active
No active alarms	PITR	Off	

Below this, there is an 'Additional info' section and an 'Items summary' section which states: 'DynamoDB updates the following information approximately every six hours.' There is a button to 'Get live item count'.

The browser toolbar at the top includes links for Gmail, YouTube, Maps, Asus Offer Redemp..., CodeSandbox, Asus Offer Redemp..., Ticket Status, Untitled site, site google.com/u..., chatgpt, What's New, Reexam online regi..., and All Bookmarks. The status bar at the bottom shows the date (27-04-2024), time (12:22), and language (ENG IN).

3) Create Roles in IAM and select amazonDynamoDBFullAccess also.

The screenshot shows the AWS IAM Roles page. The left sidebar includes options like Dashboard, Access management, Policies, Identity providers, Account settings, Access reports, External access, Unused access, Analyzer settings, and Credential report. The main content area displays a table titled "Roles (11) info" with columns for Role name, Trusted entities, and Last activity. Three rows are visible:

Role name	Trusted entities	Last activity
Powerfunction2-role-3wn5oe6u	AWS Service: lambda	-
Powerfunction2-role-dw9okjcv	AWS Service: lambda	81 days ago
role-signup	AWS Service: lambda	4 days ago

Below the table, there's a section titled "Roles Anywhere" with three cards: "Access AWS from your non AWS workloads", "X.509 Standard", and "Temporary credentials".

4) Now Create Lambda function

The screenshot shows the AWS Lambda Functions page. The left sidebar includes options like Dashboard, Applications, Functions, Additional resources, Event source mappings, Layers, Replicas, and Related AWS resources. The main content area displays a table titled "Functions (1)" with columns for Function name, Description, Package type, Runtime, and Last modified. One row is visible:

Function name	Description	Package type	Runtime	Last modified
lambda-signup	-	Zip	Python 3.12	5 days ago

The screenshot shows the AWS Lambda console interface. At the top, there are tabs for 'Items | Amazon DynamoDB' and 'Roles | IAM | Global'. The main tab is 'lambda-signup - Lambda'. Below the tabs, there's a search bar and a navigation bar with 'Services' and 'Search'.

The main content area displays the 'Function overview' for 'lambda-signup'. It shows a diagram where 'lambda-signup' is triggered by 'API Gateway'. There are buttons for '+ Add destination' and '+ Add trigger'. On the right side, there's a sidebar with 'Description', 'Last modified (5 days ago)', 'Function ARN' (arn:aws:lambda:ap-south-1:654654341426:function:lambda-signup), and 'Function URL' (Info).

Below the overview, there are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. The 'Configuration' tab is currently selected.

At the bottom, there's a footer with links for 'CloudShell', 'Feedback', and social media icons. The status bar shows the date and time as '27-04-2024 12:24'.

After creating lambda function, create environment variables.

This screenshot shows the 'Configuration' tab for the 'lambda-signup' function. The left sidebar has a list of configuration options: General configuration, Triggers, Permissions, Destinations, Function URL, Environment variables (which is selected and highlighted in blue), Tags, VPC, RDS databases, Monitoring and operations tools, and Concurrency.

The main content area shows the 'Environment variables (1)' section. It includes a search bar labeled 'Find environment variables' and a table with one entry:

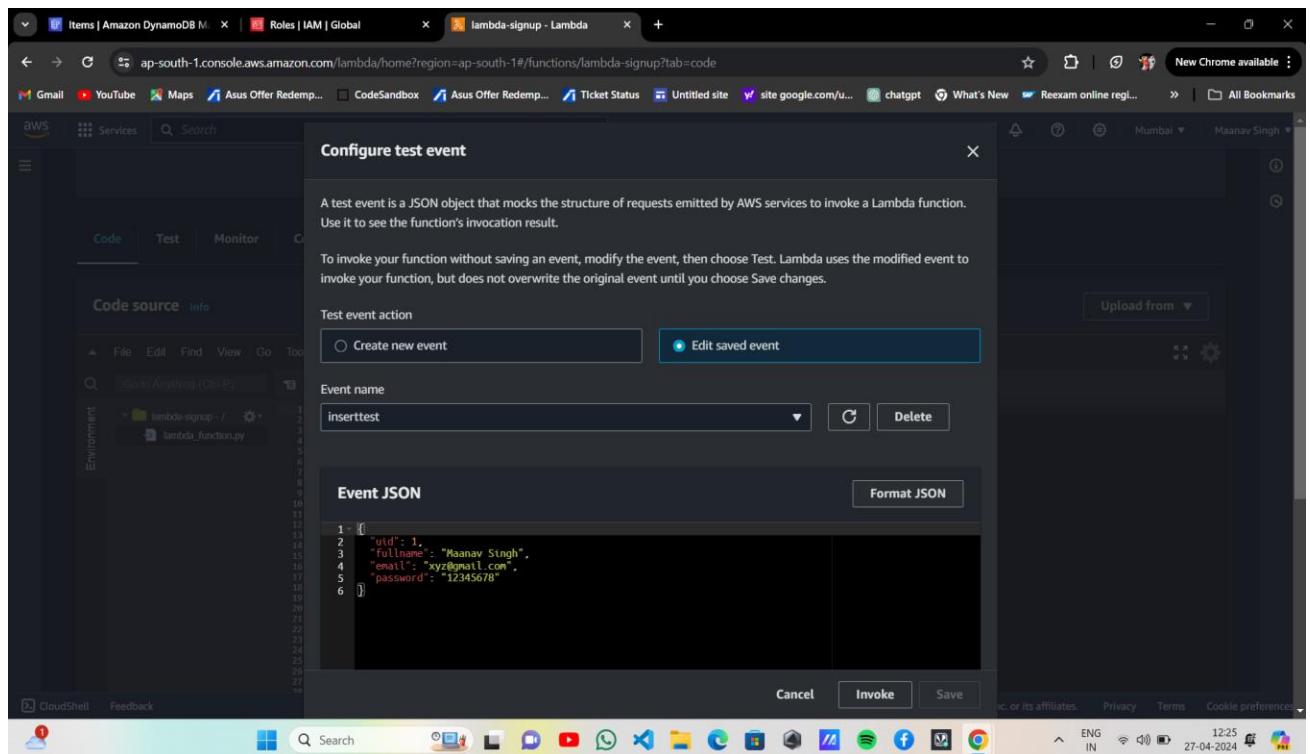
Key	Value
DYNAMODB_TABLE_NAME	db-signup-table

At the bottom, there's a footer with links for 'CloudShell', 'Feedback', and social media icons. The status bar shows the date and time as '27-04-2024 12:25'.

After creating environment variable, now write your python script and test it.

The screenshot shows the AWS Lambda function code editor. The current tab is 'lambda_function.py'. The code is as follows:

```
1 import json
2 import uuid
3 import boto3
4
5 dynamodb = boto3.resource('dynamodb')
6 table = dynamodb.Table('db-signup-table')
7
8 def lambda_handler(event, context):
9     try:
10         # Extract values from the event
11         fullname = event.get('fullname', '')
12         email = event.get('email', '')
13         password = event.get('password', '')
14
15         # Generate unique UID
16         uid = str(uuid.uuid4())
17
18         # Insert record into DynamoDB
19         table.put_item(
20             Item={
21                 'uid': uid,
22                 'fullname': fullname,
23                 'email': email,
24                 'password': password
25             }
26         )
27
28         return {
29             'statusCode': 200,
30             'body': json.dumps(['message': 'Data saved successfully'])
31         }
32     except Exception as e:
33         return {
34             'statusCode': 500,
35             'body': json.dumps(['message': f'Error saving data to DynamoDB: {str(e)}'])
36     }
37
```



5) Create API Gateway

Screenshot of the AWS API Gateway - APIs page showing a single API named "api-signup".

APIs (1/1)

Name	Description	ID	Protocol	API endpoint type	Created
api-signup	olunuzjrce	REST	Regional	2024-04-22	

Actions: Delete, Create API

CloudShell Feedback

Feedback: CloudShell, Feedback

System: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 12:26 27-04-2024

Screenshot of the AWS API Gateway - Resources page showing the configuration for the "api-signup" API.

API actions: Deploy API

Resource details:

Path: /	Resource ID: r5u1us5n2e
---------	-------------------------

Methods (2):

Method type	Integration type	Authorization	API key
OPTIONS	Mock	None	Not required
POST	Lambda	None	Not required

Actions: Delete, Create method

CloudShell Feedback

Feedback: CloudShell, Feedback

System: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 12:27 27-04-2024

The screenshot shows the AWS API Gateway Resources page for the 'api-signup' API. The left sidebar shows the 'Resources' section is selected. The main area displays the 'OPTIONS' method execution flow. The ARN is arn:aws:execute-api:ap-south-1:654654341426:olunuzjrc/*/*OPTIONS/. The flow diagram shows the sequence from Client to Method request, then to Integration request, then to Integration response, and finally to Method response. A 'Mock integration' button is present. Below the diagram, tabs for Method request, Integration request, Integration response, Method response, and Test are visible. A 'Test method' button is at the bottom.

The screenshot shows the AWS API Gateway Resources page for the 'api-signup' API. The left sidebar shows the 'Resources' section is selected. The main area displays the 'POST' method execution flow. The ARN is arn:aws:execute-api:ap-south-1:654654341426:olunuzjrc/*/*POST/. The flow diagram shows the sequence from Client to Method request, then to Integration request, then to Integration response, and finally to Method response. A 'Lambda integration' button is present. Below the diagram, tabs for Method request, Integration request, Integration response, Method response, and Test are visible. A 'Test method' button is at the bottom.

6) Create VPC

The screenshot shows the AWS VPC dashboard. On the left, a sidebar lists various network-related services: EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections), and Security (Network ACLs). The main pane displays a table titled "Your VPCs" with two entries:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options
aws_vpc	vpc-009f1e3c295410f59	Available	172.31.0.0/16	-	dopt-02
test-vpc	vpc-01da18d1ffb09900c	Available	12.0.0.0/16	-	dopt-02

The screenshot shows the "VpcDetails | VPC Console" page for the VPC with ID vpc-01da18d1ffb09900c. The top navigation bar includes links for Items, Roles, Lambda, API Gateway, and vpc. The sidebar on the left is identical to the one in the previous screenshot.

The main content area is titled "VPC > Your VPCs > vpc-01da18d1ffb09900c / test-vpc". It displays detailed information about the VPC:

VPC ID	State	DNS hostnames	DNS resolution
vpc-01da18d1ffb09900c	Available	Disabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	dopt-02a9bcc8d46ac5164	rtb-043a121be4e4a939a	acl-0e2f13210a8a3bfa5
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR (Network border group)
No	12.0.0.0/16	-	-
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	
Disabled	-	654654341426	

Below the details table, there are tabs for Resource map, CIDRs, Flow logs, Tags, and Integrations. At the bottom, there are buttons for VPC Show details, Subnets (2), Route tables (2), and Network ACLs (2).

7) Create subnets

Screenshots of the AWS VPC Subnets console showing a list of subnets and their details.

Subnets (6) Info

Name	Subnet ID	State	VPC	IPv4 CIDR
-	subnet-0a17bf4d35e8147e5	Available	vpc-009f1e3c295410f59 aws...	172.31.32.0/20
my-subnet-1	subnet-0835592c85ad6de3d	Available	vpc-009f1e3c295410f59 aws...	172.31.48.0/20
-	subnet-0153ffbb3ac0854dd	Available	vpc-009f1e3c295410f59 aws...	172.31.16.0/20
-	subnet-0db309ce8e28fea5e	Available	vpc-009f1e3c295410f59 aws...	172.31.0.0/20
test-public-subnet-1b	subnet-0cc0a398c794912b1	Available	vpc-01da18d1ffb0990c test-...	12.0.2.0/24
test-public-subnet-1a	subnet-0167065bfc6890117	Available	vpc-01da18d1ffb0990c test-...	12.0.1.0/24

SubnetDetails | VPC Console

subnet-0167065bfc6890117 / test-public-subnet-1a

Details			
Subnet ID subnet-0167065bfc6890117	Subnet ARN arn:aws:ec2:ap-south-1:654654341426:subnet/subnet-0167065bfc6890117	State Available	IPv4 CIDR 12.0.1.0/24
Available IPv4 addresses 250		Availability Zone ap-south-1a	Availability Zone ID aps1-az1
Network border group ap-south-1	IPv6 CIDR -	Route table rtb-0de52e2b22cd63df test-public-RT	Network ACL acl-0e2f13210a8a3bfa5
Default subnet No	VPC vpc-01da18d1ffb0990c test-vpc	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Auto-assign public IPv4 address No	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Outpost ID -	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	Hostname type IP name Owner 654654341426		

The screenshot shows the AWS VPC Subnets details page for a specific subnet. The subnet ID is `subnet-0cc0a398c794912b1`, and its name is `test-public-subnet-1b`. The subnet ARN is `arn:aws:ec2:ap-south-1:654654341426:subnet/subnet-0cc0a398c794912b1`. The state is `Available`. It is associated with the availability zone `ap-south-1b` and the network ACL `acl-0e2f13210a8a3bfa5`. The VPC is `vpc-01da18d1ff0b09900c`. The subnet has 250 available IPv4 addresses and no default subnet assigned. It is part of the test-public-RT route table. There are no IPv6 CIDR reservations or auto-assign IPv6 address options. The owner is listed as `654654341426`.

8) Create Route Table

The screenshot shows the AWS Route Tables list page. It displays three existing route tables: `rtb-043a121be4e4a939a`, `rtb-0426fab6f79133c09`, and `rtb-0de52e2b22cdb63df`. The third route table, `rtb-0de52e2b22cdb63df`, is associated with the `test-public-RT` route table and connects to two subnets. The main column indicates that the first two route tables are the main route table for their respective VPCs, while the third one is not.

The screenshot shows the AWS VPC Route Table Details page. The route table ID is rtb-0de52e2b22cdb63df, associated with the VPC vpc-01da18d1ff09900c | test-vpc. It has two explicit subnet associations: 2 subnets. There are two routes listed:

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0d28d36e05a423d17	Active	No
12.0.0.0/16	local	Active	No

9) Create Internet Gateway

The screenshot shows the AWS Internet Gateways list page. There are two internet gateways listed:

Name	Internet gateway ID	State	VPC ID	Owner
igw-test	igw-0d28d36e05a423d17	Attached	vpc-01da18d1ff09900c test-vpc	654654341426
igw-aws-default	igw-0fcf7f74ae980da063	Attached	vpc-009f1e3c295410f59 aws_vpc	654654341426

The screenshot shows the AWS VPC Internet gateway configuration page. The URL in the address bar is `ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#InternetGateway:internetGatewayId=igw-0d28d36e03a423d17`. The main content area displays the details for an Internet gateway named `igw-0d28d36e03a423d17 / igw-test`. The gateway is attached to a VPC with ID `vpc-01da18d1ffb0990c` and owner `654654341426`. A single tag named `igw-test` is listed under the Tags section. The left sidebar shows the VPC dashboard and various VPC-related options like Subnets, Route tables, and Internet gateways.

10) Create S3 Bucket and upload your website code (HTML,CSS,JS etc.)

The screenshot shows the AWS S3 buckets list. The URL in the address bar is `ap-south-1.console.aws.amazon.com/s3/home?region=ap-south-1#`. The main content area shows a single bucket named `buckypajji` in the General purpose buckets section. The bucket was created on April 15, 2024, at 15:54:30 (UTC+05:30). The left sidebar shows the Amazon S3 navigation menu with options like Buckets, Access Grants, and Storage Lens.

The screenshot shows the AWS S3 console interface. On the left, a sidebar lists various services like Amazon DynamoDB, IAM, Lambda, API Gateway, VPC, and CloudWatch Metrics. The main area displays the 'buckypaji' bucket. The 'Objects' tab is selected, showing one item: 'AWS_Website_Project.zip' (Type: zip, Last modified: April 22, 2024, 15:14:37 (UTC+0:30), Size: 856.3 KB, Storage class: Standard). The top navigation bar shows the URL as 'ap-south-1.console.aws.amazon.com/s3/buckets/buckypaji?region=ap-south-1&bucketType=general&tab=objects'.

11) Now create EC2 instance and Host your website on it.

The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 Dashboard, Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. The main table shows one instance: 'temp-website' (Instance ID: i-00f09a039296c982, Status: Running, Instance type: t2.micro, Public IPv4 DNS: ap-south-1b). A modal window titled 'Select an instance' is open at the bottom. The top navigation bar shows the URL as 'ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1&instancesv=3:\$case=tags:true%5Cclient:false:\$regex=tagsfalse%5Cclientfalse'.

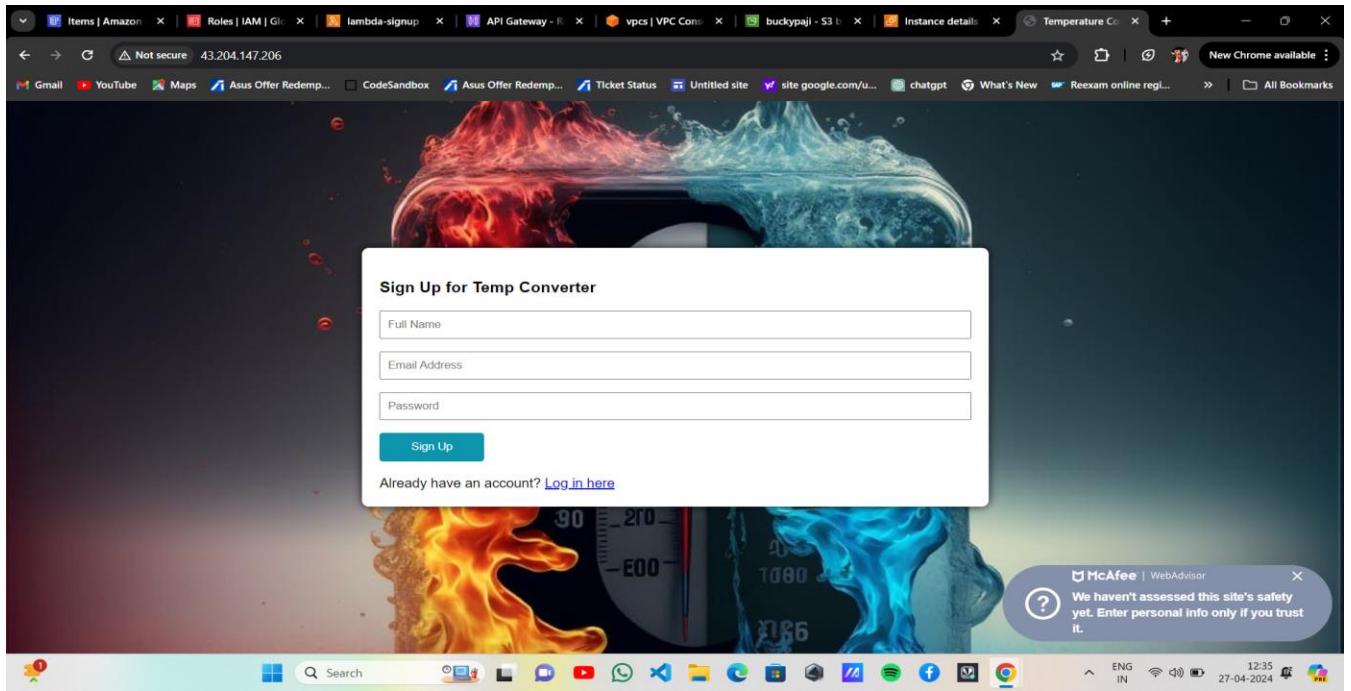
The screenshot shows the AWS EC2 Instances page. The left sidebar has sections for EC2 Dashboard, EC2 Global View, Events, Instances (with sub-options like Instances, Instance Types, Launch Templates, etc.), Images, AMIs, AMI Catalog, and Elastic Block Store (with Volumes and Snapshots). The main content area is titled "Instance summary for i-00f09a0392960c982 (temp-website)". It displays the following details:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-00f09a0392960c982 (temp-website)	43.204.147.206 [open address]	12.0.2.226
IPv6 address	Instance state	Public IPv4 DNS
-	Running	-
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-12-0-2-226.ap-south-1.compute.internal	ip-12-0-2-226.ap-south-1.compute.internal	-
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
-	t2.micro	Opt-in to AWS Compute Optimizer for recommendations.
Auto-assigned IP address	VPC ID	[Learn more]
43.204.147.206 [Public IP]	vpc-01da18d1ffb09900c (test-vpc)	
IAM Role	Subnet ID	Auto Scaling Group name
-	subnet-0cc0a398c794912b1 (test-public-subnet-1b)	-
IMDSv2		
Required		

At the bottom of the main content area, there is a note: "Auto-assigned IP address copied". The status bar at the bottom right shows "ENG IN" and the date "27-04-2024".

After creating EC2 instance, copy the public ip and paste it on another tab to see your website properly hosted or not.

This screenshot is identical to the one above, showing the EC2 Instances page for the same instance. The key difference is the message "Auto-assigned IP address copied" is displayed prominently in the center of the main content area, indicating the public IP has been copied.



12) Create target group for load balancer.

Name	ARN	Port	Protocol	Target type	Load balancer
tg-temp	arn:aws:elasticloadbalancing:ap-south-1:123456789012:targetgroup/tg-temp/12345678901234567890	80	HTTP	Instance	lb-temp

tg-temp

Details

arn:aws:elasticloadbalancing:ap-south-1:654654341426:targetgroup/tg-temp/3607df65efef665a

Target type	Protocol : Port	Protocol version	VPC
instance	HTTP: 80	HTTP1	vpc-01da18d1ff09900c

Total targets	Healthy	Unhealthy	Unused	Initial	Draining
1	1	0	0	0	0

Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets Monitoring Health checks Attributes Tags

13) After creating target group, it's time to create load balancer.

Create load balancer

Name	DNS name	State	VPC ID	Availability Zones	Type
lb-temp	lb-temp-597939375.ap-so...	Active	vpc-01da18d1ff09900c	2 Availability Zones	application

0 load balancers selected

lb-temp

Details

Load balancer type	Status	VPC	IP address type
Application	Active	vpc-01da18d1ff0990c	IPv4
Scheme	Hosted zone	Availability Zones	Date created
Internet-facing	ZP97RAFLXTNZK	subnet-0cc0a398c794912b1 ap-south-1b (aps1-az3)	April 22, 2024, 15:37 (UTC+05:30)
		subnet-0167065bfc6890117 ap-south-1a (aps1-az1)	
Load balancer ARN	DNS name	Listeners and rules (1)	
arn:aws:elasticloadbalancing:ap-south-1:654654341426:loadbalancer/app/lb-temp/ead83fd9e27b7653	lb-temp-597939375.ap-south-1.elb.amazonaws.com (A Record)	Manage rules	Manage listener

Listeners and rules (1)

CloudShell Feedback

14) Now we are going to create Web ACLs in WAF.

WAF & Shield

AWS WAF > Web ACLs

Web ACLs (1)

Name	Description	ID
waf-temp	-	b243e716-bb35-4481-80d7-5379625a1e1b

Asia Pacific (Mumbai)

Copy ARN Delete Create web ACL

CloudShell Feedback

The screenshot shows the AWS WAF & Shield console with the URL us-east-1.console.aws.amazon.com/wafv2/homev2/web-ac/waf-temp/b243e716-bb35-4481-80d7-5379625a1e1b/overview?region=ap-south-1. The left sidebar is expanded to show the AWS WAF section, specifically the Web ACLs category. The main content area is titled "waf-temp" and includes tabs for "Traffic overview", "Rules", "Associated AWS resources", "Custom response bodies", "Logging and metrics", "Sampled requests", and "CloudWatch". A "Data filters" section allows selecting time ranges and terminating rule actions like "Blocked", "Allowed", "Captcha", and "Challenge". Below this, a "Bot Control" section is visible. A summary table at the bottom shows "Action totals for the specified time range - all traffic" with columns for Total, Blocked, Allowed, and Captcha. The status bar at the bottom right indicates the date as 27-04-2024.

15) Create IP sets

The screenshot shows the AWS WAF & Shield console with the URL us-east-1.console.aws.amazon.com/wafv2/homev2/ip-sets?region=ap-south-1. The left sidebar is expanded to show the AWS WAF section, specifically the IP sets category. The main content area is titled "IP sets (1)" and includes a search bar and a table. The table has columns for Name, Description, ID, and ARN. One entry is listed: "ip-temp" with ID ce70e3f9-ee5b-4818-9617-02da53667e96. The status bar at the bottom right indicates the date as 27-04-2024.

The screenshot shows the AWS WAF IP sets configuration page. The left sidebar is titled "WAF & Shield" and includes sections for "AWS WAF" (Getting started, Web ACLs, Bot control dashboard, Application integration, IP sets, Regex pattern sets, Rule groups, AWS Marketplace managed rules) and "AWS Shield" (Getting started, Overview). The main content area is titled "ip-temp" under "AWS WAF > IP sets > ip-temp". It displays an "Info" section with fields: Name (ip-temp), Region (Asia Pacific (Mumbai)), Description (empty), and IP version (IPv4). Below this is a table titled "IP addresses (1)" showing one entry: 122.161.72.120/32.

The screenshot shows the AWS WAF Web ACL configuration page for "waf-temp". The left sidebar is identical to the previous screenshot. The main content area is titled "waf-temp" under "AWS WAF > Web ACLs > waf-temp". It includes tabs for Traffic overview, Rules (selected), Associated AWS resources, Custom response bodies, Logging and metrics, Sampled requests, CloudWatch, and Download web ACL as JSON. The "Rules (1)" section shows a single rule named "block-my-ip-temp" with the action "CAPTCHA" and priority 0. Below this is a section titled "Web ACL capacity units (WCUs) used by your web ACL" stating "1/5000 WCUs".

16) After establishing WAF successfully now we will go back to EC2 instance and run the website and enter the required details and we're able to see the filled details in DynamoDB.

Screenshot of the AWS EC2 Instances page showing the details for instance i-00f09a0392960c982.

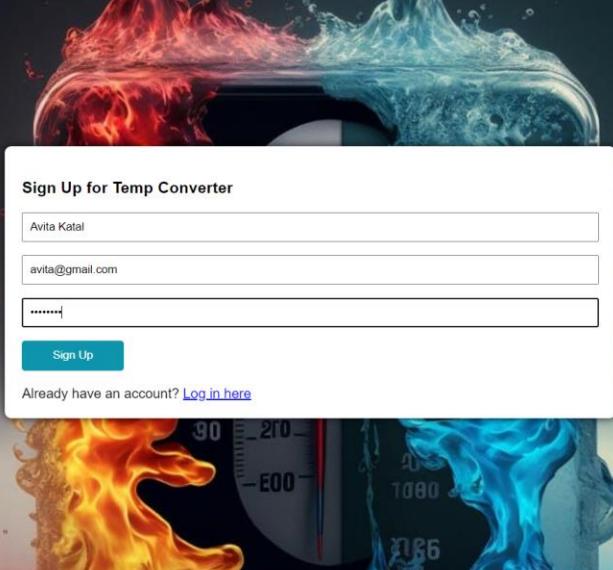
Instance summary for i-00f09a0392960c982 (temp-website)

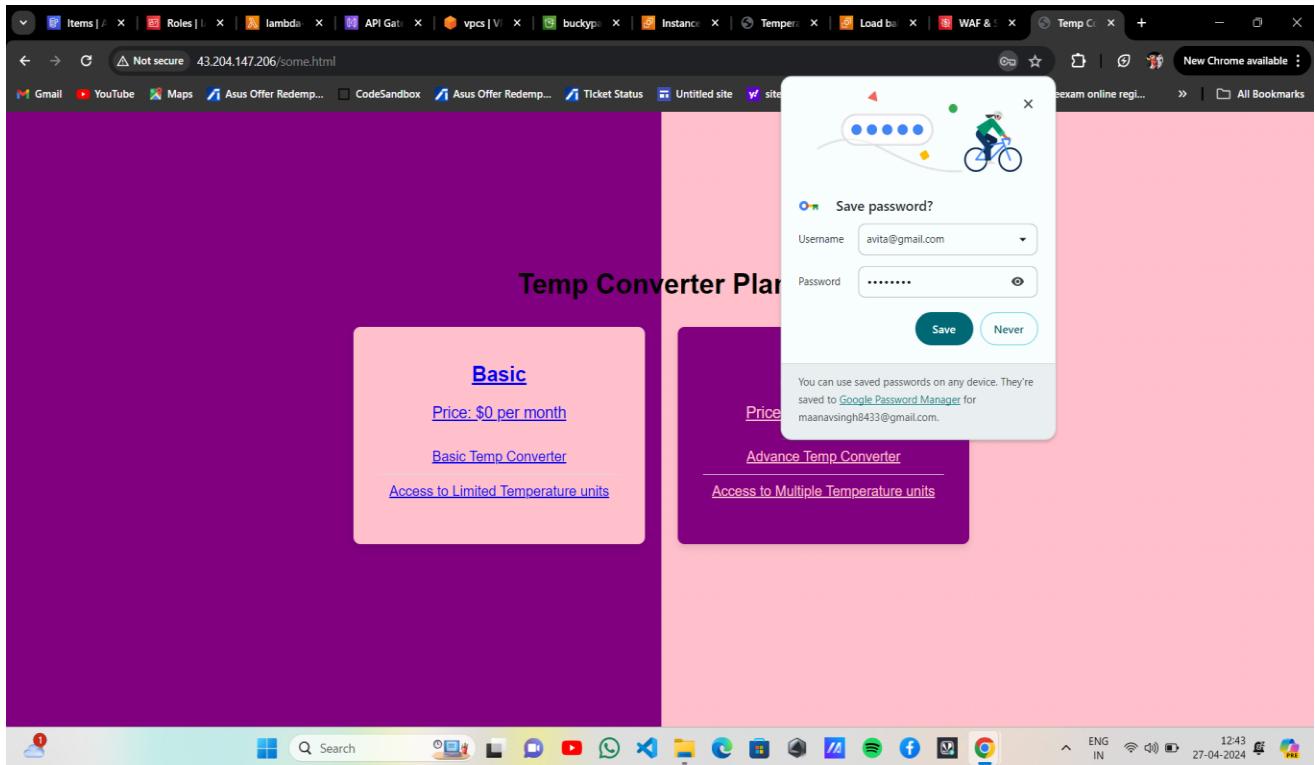
Attribute	Value
Instance ID	i-00f09a0392960c982 (temp-website)
IPv6 address	-
Hostname type	IP name: ip-12-0-2-226.ap-south-1.compute.internal
IAM Role	-
IMDSv2	Required
Public IPv4 address	43.204.147.206 [open address]
Private IP DNS name (IPv4 only)	ip-12-0-2-226.ap-south-1.compute.internal
Private IP IPv4 address	12.0.2.226
Private IP IPv4 DNS	-
Private IP DNS name (IPv6 only)	-
Private IP IPv6 address	-
Private IP IPv6 DNS	-
Instance type	t2.micro
VPC ID	vpc-01da18d1ff09900c (test-vpc)
Subnet ID	subnet-0cc0a398c794912b1 (test-public-subnet-1b)
Elastic IP addresses	-
AWS Compute Optimizer finding	Opt-in to AWS Compute Optimizer for recommendations.
Auto Scaling Group name	-

Screenshot of a web browser displaying a sign-up form for "Temp Converter".

Sign Up for Temp Converter

Already have an account? [Log in here](#)





A screenshot of the AWS DynamoDB console. The URL in the address bar is 'ap-south-1.console.aws.amazon.com/dynamodbv2/home?region=ap-south-1#item-explorer?maximize=true&operation=SCAN&table=db-signup-table'. The left sidebar shows navigation options like 'Dashboard', 'Tables', 'Explore items', and 'DAX'. The main area shows a table named 'db-signup-table' with one item listed. A success message at the top says 'Selected items have been deleted successfully.' Below the table, there are sections for 'Scan or query items' (with 'Scan' selected), 'Filters', and a results table titled 'Items returned (1/5)'. The results table shows one item with columns 'id', 'email', and 'password'. At the bottom, there are buttons for 'Actions' and 'Create item'. The status bar at the bottom indicates '© 2024, Amazon Web Services, Inc. or its affiliates.' and shows the date '27-04-2024'.

The screenshot shows the AWS DynamoDB console interface. On the left, a sidebar lists various options: Dashboard, Tables, Explore items (which is selected), PartiQL editor, Backups, Exports to S3, Imports from S3, Integrations, Reserved capacity, Settings, DAX (Clusters, Subnet groups, Parameter groups, Events), CloudShell, and Feedback. The main area is titled "DynamoDB" and contains a search bar and a "Filters" section with "Run" and "Reset" buttons. A green notification box at the top right says "Completed. Read capacity units consumed: 0.5". Below it, a table titled "Items returned (1/5)" shows five rows of data:

	uid (String)	email	fullname	password
<input type="checkbox"/>	2fa9f1bf-fb12-4fa6-8...	xyz@gmail....	Maanav Singh	12345678
<input type="checkbox"/>	e96a3203-0bbe-4de3...	abc@gmail....	Raghav Aga...	87654321
<input type="checkbox"/>	ea7f136c-e01b-4b4b...	maanavbea...	Maanav Singh	kdnfngos
<input checked="" type="checkbox"/>	0050e66e-b2eb-48aa...	avita@gmai...	Avita Katal	12345678
<input type="checkbox"/>	4a106be7-50dd-46bf...	tarushi010...	Tarushi cha...	iosuiafb

At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information and language settings.

THANK YOU !!