

Capstone Project : MatterMost

Team No : 11

Submitted By :

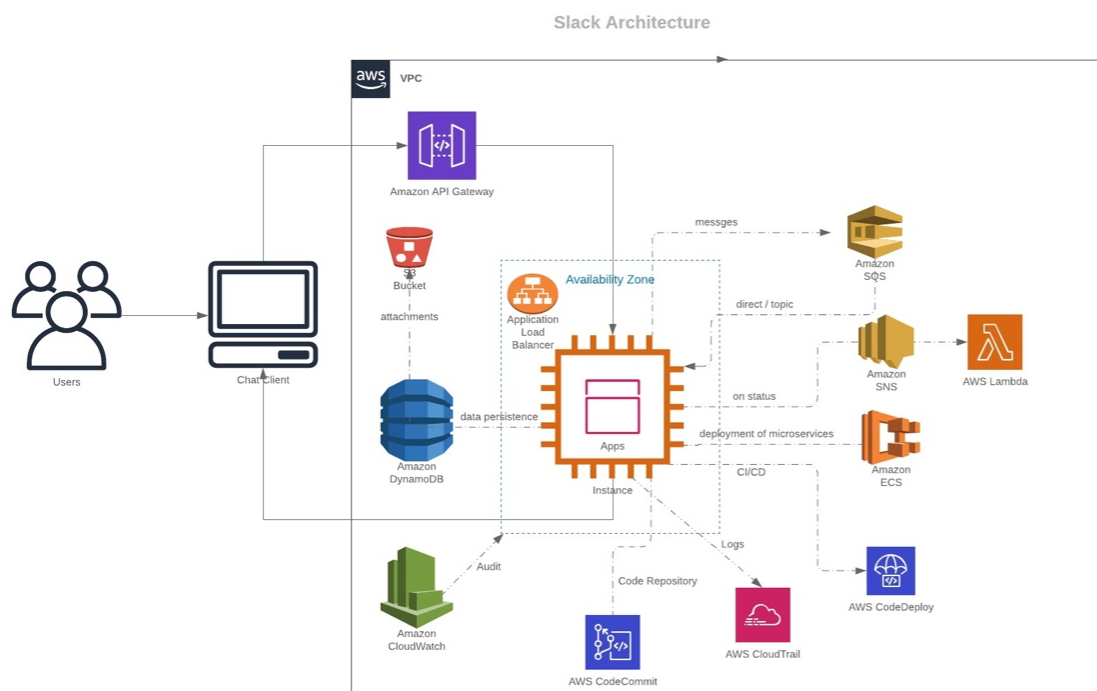
Rex Roy

Rahul Chatterjee

Nagendra

Prakhar

Architecture Diagram:



Implementation Steps :

1. Create custom VPC and Subnets
2. Create EC2 instance
3. Deploy the MatterMost application
4. Integration with various AWS services
5. Code maintenance

Step 1 : Create custom VPC and Subnets

VPC : GL_Capstone

The screenshot shows the AWS VPC Dashboard. On the left, the 'Virtual Private Cloud' section is expanded, showing 'Your VPCs'. The main area displays a table of VPCs. The VPC 'GL_Capstone' with ID 'vpc-08dc1545777d0e004' is highlighted with a red box. Below the table, the details for this VPC are shown, including its state (available), IPv4 CIDR (10.0.0.0/16), DHCP options set (dopt-c38b9a8), Main Route table (rtb-0446d046f6b9f13b), Main Network ACL (acl-0b2f071eeccd58406), Tenancy (default), and Owner (1217729391).

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set	Main Route table	Main Network ACL	Tenancy	Default VPC	Owner
GL_Capstone	vpc-08dc1545777d0e004	available	10.0.0.0/16	-	dopt-c38b9a8	rtb-0446d046f6b9f13b	acl-0b2f071eeccd58406	default	No	1217729391

VPC: vpc-08dc1545777d0e004

Description	CIDR Blocks	Flow Logs	Tags
VPC ID vpc-08dc1545777d0e004 State available IPv4 CIDR 10.0.0.0/16 DNS resolution Enabled DNS hostnames Disabled Route table rtb-0446d046f6b9f13b			Tenancy default Default VPC No IPv6 CIDR - Network ACL acl-0b2f071eeccd58406 DHCP options set dopt-c38b9a8 Owner 1217729391

Subnets : Capstone_private, Capstone_Public

The screenshot shows the AWS Subnet Dashboard. On the left, the 'Virtual Private Cloud' section is expanded, showing 'Your VPCs'. The main area displays a table of subnets. The subnets 'capst_private' and 'capst_public' are highlighted with a red box. Below the table, the details for these subnets are shown, including their state (available), VPC (vpc-08dc1545777d0e004), IPv4 CIDR, Available IPv4, IPv6 CIDR, Availability Zone, Availability Zone ID, Route table, and Network ACL.

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID	Route table	Network ACL
capst_private	subnet-016acdb4ef06311dc	available	vpc-08dc1545777d0e004	10.0.2.0/24	250	-	us-east-2c	use2-az3	rtb-0446d046f6b9f13b	acl-0b2f071
capst_public	subnet-031bdcf959a365e0	available	vpc-08dc1545777d0e004	10.0.1.0/24	250	-	us-east-2b	use2-az2	rtb-056823493c671db4 capst...	acl-0b2f071
gl-public-su...	subnet-0c2fa0c99c124fb0b	available	vpc-08dc1545777d0e004	10.0.3.0/24	251	-	us-east-2a	use2-az1	rtb-0446d046f6b9f13b	acl-0b2f071
subnet-1cda8866	subnet-1cda8866	available	vpc-1bcd3170	172.31.16.0/20	4091	-	us-east-2b	use2-az2	rtb-c5c440ae	acl-4d271c2
subnet-413dd92a	subnet-413dd92a	available	vpc-1bcd3170	172.31.0.0/20	4091	-	us-east-2a	use2-az1	rtb-c5c440ae	acl-4d271c2
subnet-bcd05ef0	subnet-bcd05ef0	available	vpc-1bcd3170	172.31.32.0/20	4090	-	us-east-2c	use2-az3	rtb-c5c440ae	acl-4d271c2

Step 2 : Create EC2 instance

The screenshot displays the AWS Management Console interface. On the left, the navigation menu includes sections like EC2 Dashboard, INSTANCES, IMAGES, ELASTIC BLOCK STORE, NETWORK & SECURITY, and Key Pairs. The main content area shows the details for an EC2 instance named 'capstone-chat' with ID 'i-00febcd999aeff38'. The instance is in a 'running' state, located in the 'us-east-2b' availability zone. The console provides a comprehensive overview of the instance's configuration, including its instance type (t2.micro), VPC (vpc-08dc1545777d0e004), subnet (subnet-031bcbcf959a365e0), and network interfaces (eth0). It also lists various attributes such as the public IP (3.135.228.53), private IP (10.0.1.43), and the AMI ID (ami-0d5d9d301c853a04a). The instance is associated with the 'Capstone' key pair and has a 'disabled' monitoring status.

```
trroy-a01:Capstone Project - MatterMost trroy$ ssh -i Capstone.pem ubuntu@3.135.228.53
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1051-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Dec 17 11:59:58 UTC 2019

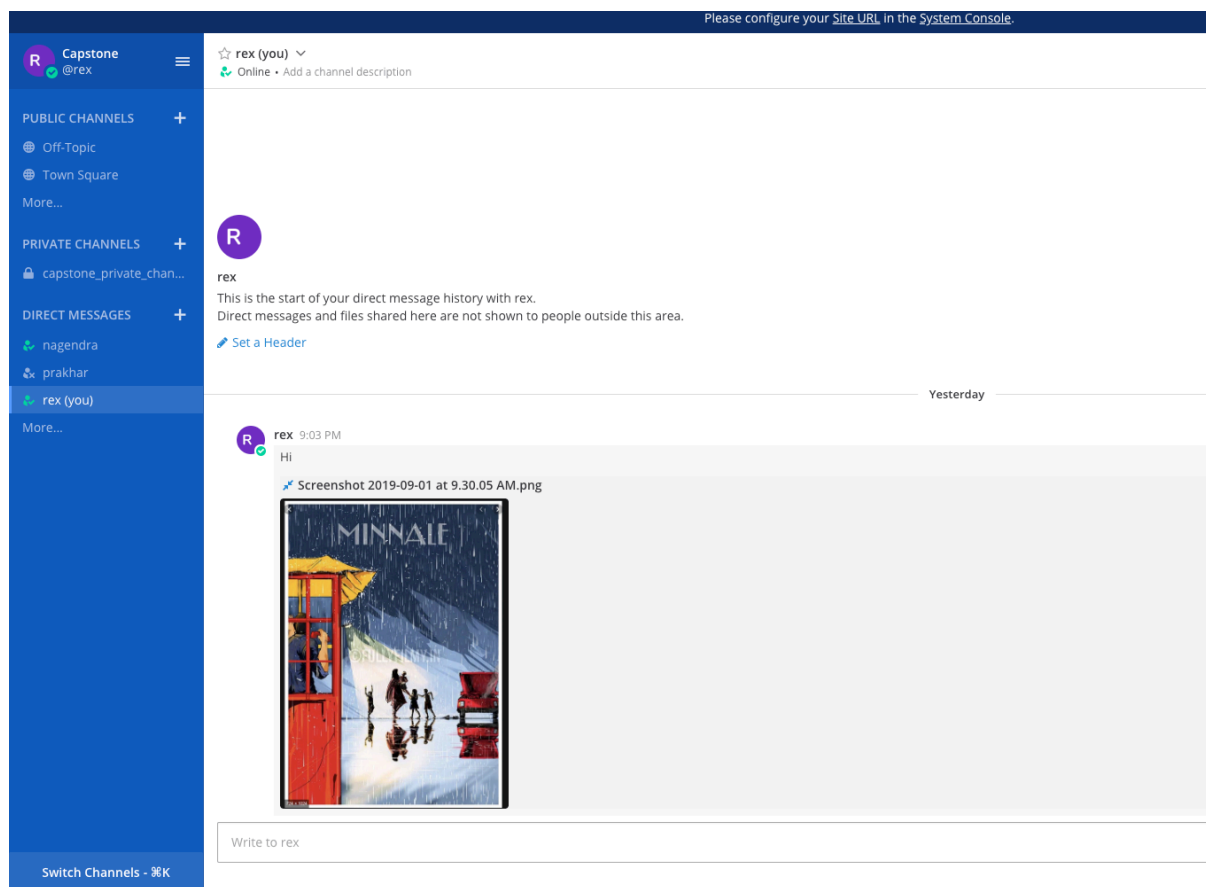
System load:  0.0           Processes:            96
Usage of /:   41.9% of 7.69GB Users logged in:       0
Memory usage: 56%          IP address for eth0:  10.0.1.43
Swap usage:   0%           IP address for docker0: 172.17.0.1

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

36 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Mon Dec 16 15:20:54 2019 from 49.206.11.21
```

Step 3: Deploy the MatterMost application



MatterMost is an open source collaboration tool. We have implemented this messenger based tool in an Ubuntu Instance in AWS. Primarily we have implemented the following features.

Features :

1. Sign-In Pages – Integrated with OAuth
2. Direct Message
3. Private channel message
4. Public channel message

AWS Service Used :

1. S3 Bucket
2. RDS – MySQL DB
3. Cloud Watch
4. Route53
5. Code commit
6. SNS & Lambda
7. Amazon Container Services & ECR

Features :

1. Sign-In Pages – Integrated with OAuth

Mattermost

All team communication in one place,
searchable and accessible anywhere

Let's create your account

Already have an account? [Click here to sign in.](#)

What's your email address?

artidhamija26@gmail.com

Valid email required for sign-up

Choose your username

ArtiDhamija

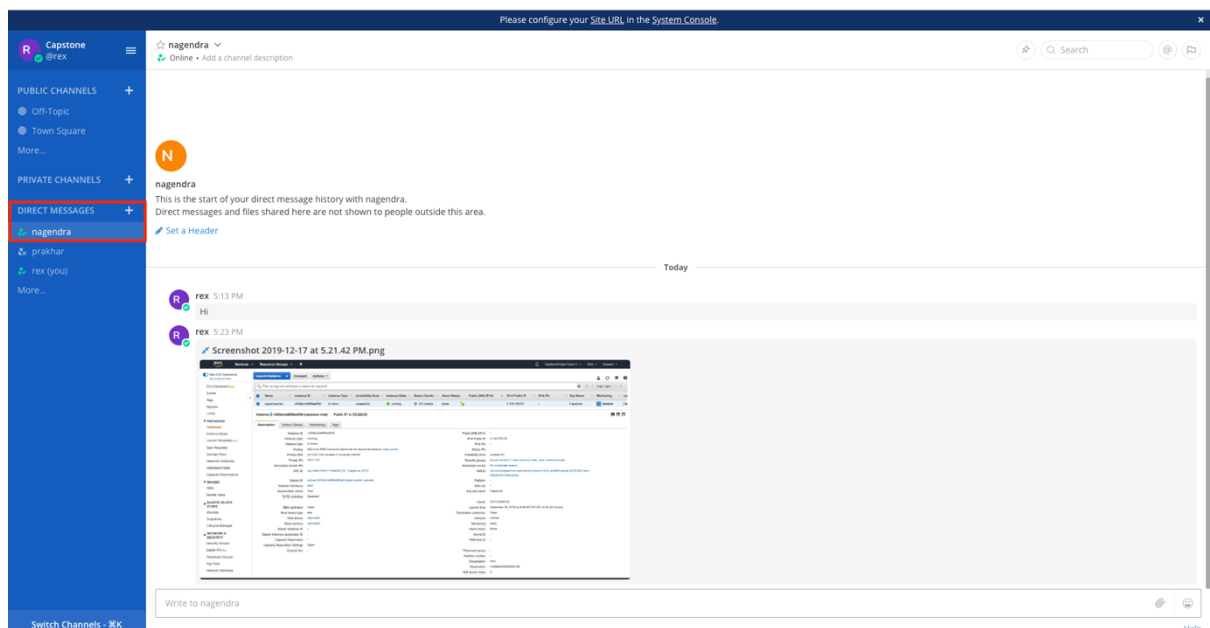
⚠ Usernames have to begin with a lowercase letter and be 3-22 characters long. You can use lowercase letters, numbers, periods, dashes, and underscores.

Choose your password

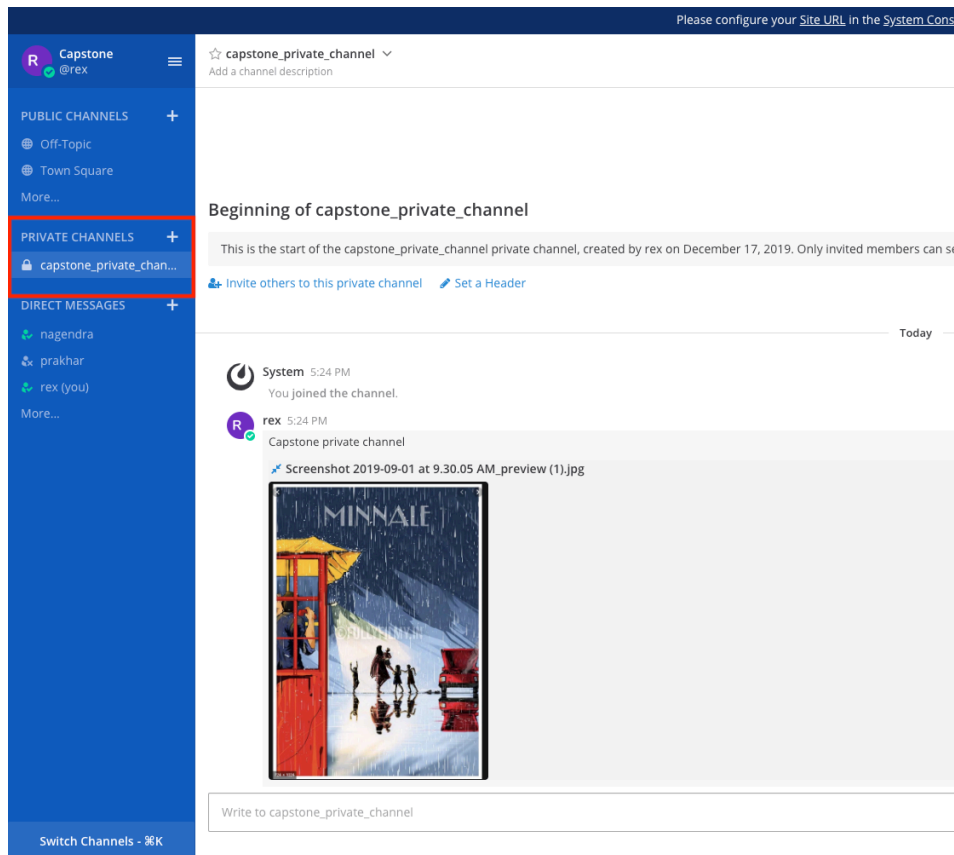
Create Account

By proceeding to create your account and use Mattermost, you agree to our [Terms of Service](#) and [Privacy Policy](#). If you do not agree, you cannot use Mattermost.

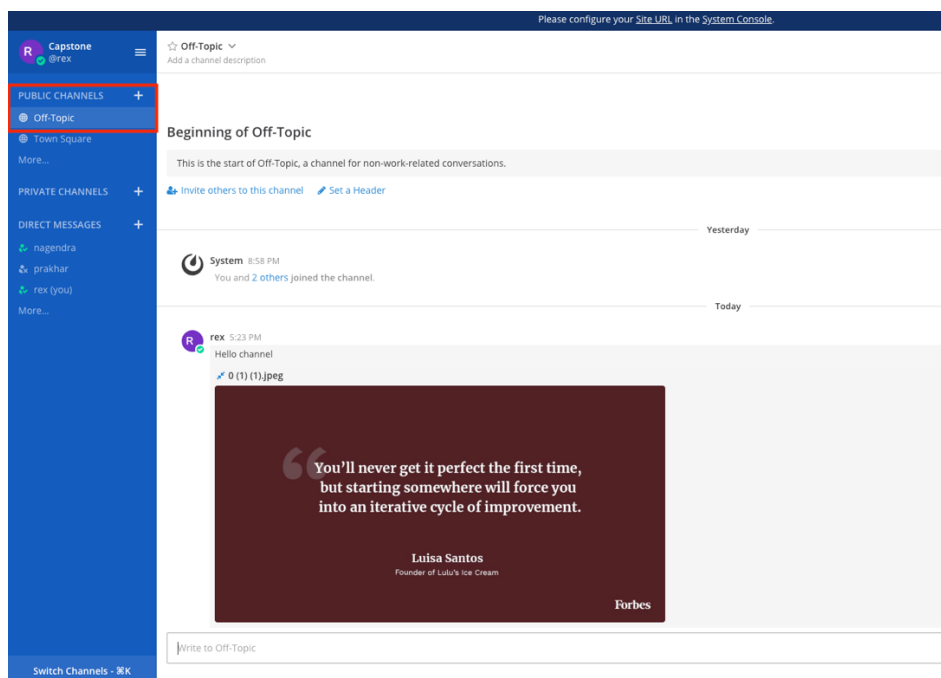
2. Direct Message:



3. Private Channel message – where only the members part of the channel will receive the messages



4. Public Channel message – Broadcasts the message to everyone



Step 4 : Integration with various AWS services

1. S3 Bucket – All the messages and attachment sent will be stored in the custom S3 bucket – **capstone-bucket-team11**

S3 configuration Details :

File Storage

File Storage System:

Amazon S3

Storage system where files and image attachments are saved.

Selecting "Amazon S3" enables fields to enter your Amazon credentials and bucket details.

Selecting "Local File System" enables the field to specify a local file directory.

Local Storage Directory:

/mm/mattermost-data/

Directory to which files and images are written. If blank, defaults to ./data/.

Maximum File Size:

50

Maximum file size for message attachments in megabytes. Caution: Verify server memory can support your setting choice. Large file sizes increase the risk of server crashes and failed uploads due to network interruptions.

Amazon S3 Bucket:

capstone-bucket-team11

Name you selected for your S3 bucket in AWS.

Amazon S3 Region:

us-east-2

AWS region you selected when creating your S3 bucket. If no region is set, Mattermost attempts to get the appropriate region from AWS, or sets it to 'us-east-1' if none found.

Amazon S3 Access Key ID:

AKIAJPPJI4NGSSQFIA

(Optional) Only required if you do not want to authenticate to S3 using an [IAM role](#). Enter the Access Key ID provided by your Amazon EC2 administrator.

Amazon S3 Endpoint:

s3.us-east-2.amazonaws.com

Hostname of your S3 Compatible Storage provider. Defaults to "s3.amazonaws.com".

Amazon S3 Secret Access Key:

Files in S3 bucket :

aws

Services

Resource Groups

CapstoneProjectTeam11

Global

Support

Amazon S3

capstone-bucket-team11

capstone-bucket-team11

Overview

Properties

Permissions

Management

Access points

Q

Type a prefix and press Enter to search. Press ESC to clear.

Upload

Create folder

Download

Actions

US East (Ohio)

Viewing 1 to 6

Name	Last modified	Size	Storage class
20191215	--	--	--
20191216	--	--	--
20191217	--	--	--
mm	--	--	--
users	--	--	--
0 (1).jpeg	Dec 15, 2019 11:33:59 PM GMT+0530	32.7 KB	Standard

Viewing 1 to 6

File Details S3 Bucket :

The screenshot shows the AWS S3 console interface. The breadcrumb navigation path is: Amazon S3 > capstone-bucket-team11 > 20191217 > teams > noteam > channels > 19r1dyqubuf9jkaj74iksutx9h > users > 7mdpia1pcbnt9p14ywnbxf500 > oh5zpo3cqtmb3cbfysp7ga67ch > 0 (1) (1).jpeg. The file name '0 (1) (1).jpeg' is displayed with a 'Latest version' dropdown. Below the name are tabs for Overview, Properties, Permissions, and Select from. Action buttons include Open, Download, Download as, Make public, and Copy path. The file's metadata is listed: Owner (512c10b9a5dd0f347e62ec1933494e4d70d5a8a61e2df31a01530bf57ec71669), Last modified (Dec 17, 2019 5:24:06 PM GMT+0530), Etag (a3c93e186e0b6fd77b5df29e8db1a1), Storage class (Standard), Server-side encryption (None), Size (32.7 KB), Key (20191217/teams/tearn/channels/19r1dyqubuf9jkaj74iksutx9h/users/7mdpia1pcbnt9p14ywnbxf500/oh5zpo3cqtmb3cbfysp7ga67ch/0 (1) (1).jpeg), and Object URL (https://capstone-bucket-team11.s3.us-east-2.amazonaws.com/20191217/teams/tearn/channels/19r1dyqubuf9jkaj74iksutx9h/users/7mdpia1pcbnt9p14ywnbxf500/oh5zpo3cqtmb3cbfysp7ga67ch/0(1)(1).jpeg).

2. RDS – MySQL DB

DB Name : capstonesql

The screenshot shows the AWS RDS console 'Databases' page. A notification banner at the top states: 'Update your Amazon RDS SSL/TLS certificates before March 5, 2020'. Below the banner, the 'Databases' section includes a search bar and a table of databases. The table has columns: DB identifier, Role, Engine, Region & AZ, Size, Status, CPU, Current activity, and Maintenance. One database is listed: 'capstonesql' with Role 'Instance', Engine 'MySQL Community', Region & AZ 'us-east-2c', Size 'db.t2.micro', Status 'Available', CPU '1.48%', Current activity '2 Connections', and Maintenance 'next window'. Action buttons at the top right include 'Group resources', 'Modify', 'Actions', 'Restore from S3', and 'Create database'.

DB Connection Details :

The screenshot shows a database connection configuration window with tabs for 'Standard', 'Socket', and 'SSH'. The 'Standard' tab is selected. The fields are: Name (capstone), Host (capstonesql.czynjkiyr4f5.us-east-2.rds), Username (root), Password (masked with dots), Database (capstonesql), and Port (3306). There is a checkbox for 'Connect using SSL' which is unchecked. A 'Connect' button is at the bottom right.

Table Details:

[illegible]

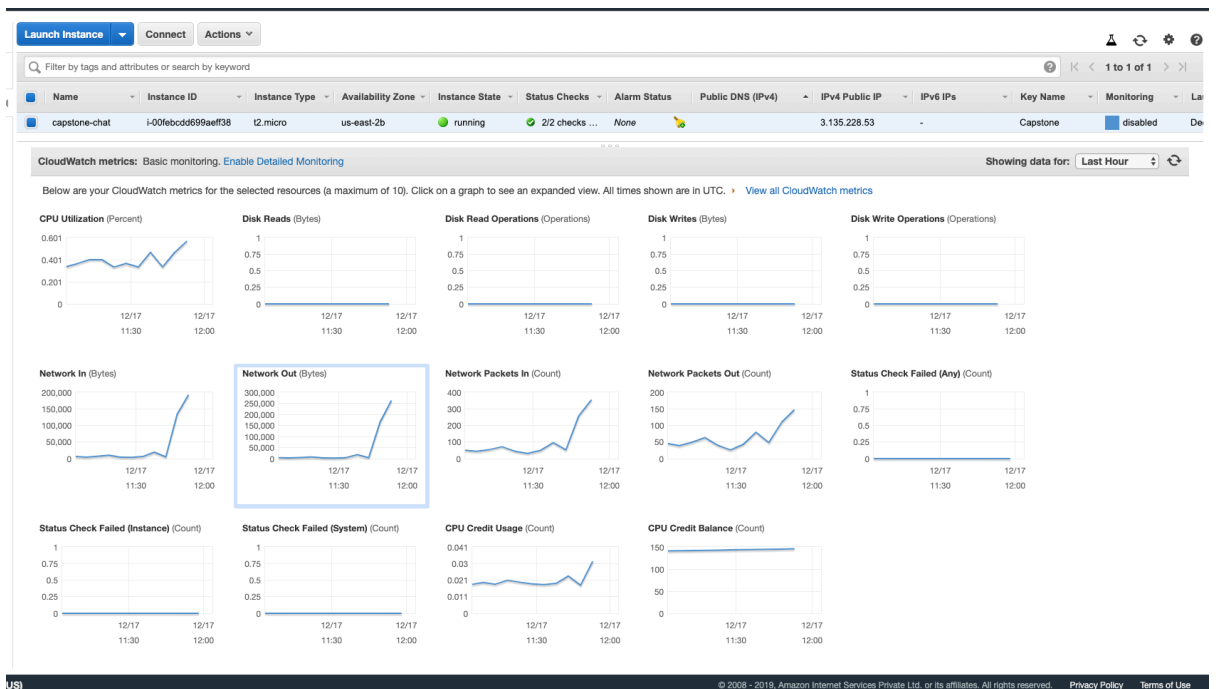
Configuration Details :

```

},
"JmxMinVersion": "1.10",
"SqlSettings": {
  "DriverName": "mysql",
  "DriverClass": "com.mysql.jdbc.Driver",
  "DataSourceReplicas": [],
  "DataSourceSearchReplicas": [],
  "MaxIdleConns": 20,
  "ConnMaxLifetimeMilliseconds": 3600000,
  "MaxOpenConns": 300,
  "Trace": false,
  "AtRestEncryptKey": "",
  "QueryTimeout": 30
}
}

```

3. Cloud watch :



4. Route53 :

The screenshot shows the AWS Route53 console. The main table displays the following record sets:

Name	Type	Value	Evaluate Target Health	Health Check ID	TTL	Region	Weight	Geolocation	Multivalue Answer
capstoneproject11.com	NS	ns-234.awdns-29.com, ns-1005.awdns-61.net, ns-1530.awdns-63.org, ns-1706.awdns-21.co.uk	-	-	172800				
capstoneproject11.com	SOA	ns-234.awdns-29.com. awdns-hostmaster.amazon	-	-	900				
www.capstoneproject11.com	A	3.135.228.53	-	-	300				

The right sidebar shows the 'Edit Record Set' form for the selected NS record. The 'Name' is 'capstoneproject11.com', the 'Type' is 'NS - Name server', and the 'TTL (Seconds)' is 172800. The 'Value' field contains the same four name servers as the table.

5. Code commit :

The screenshot shows the AWS CodeCommit console. The main view displays the 'Slack_code_repo' repository. The repository is empty, with a list of files: 'Docker_app_code', 'Docker_file', and 'entrypoint.sh'. The right sidebar shows the 'Add file' button.

The screenshot shows the AWS CodeCommit console. The main view displays the 'Slack_code_repo' repository. The 'Docker_file' file is selected, and its content is displayed. The file content is as follows:

```
1 # Copyright (C) 2016-present Mattermost, Inc. All Rights Reserved.
2 # See License.txt for license information.
3 FROM mysql:5.7
4
5 RUN apt-get update && apt-get install -y ca-certificates
6
7 #
8 # Configure SQL
9 #
10
11 ENV MYSQL_ROOT_PASSWORD=mostest
12 ENV MYSQL_USER=mmuser
13 ENV MYSQL_PASSWORD=mostest
14 ENV MYSQL_DATABASE=mattermost_test
15
16 #
17 # Configure Mattermost
18 #
19 WORKDIR /mm
20
21 # Copy over files
22 ADD https://releases.mattermost.com/5.18.0-rc1/mattermost-team-5.18.0-rc1-linux-amd64.tar.gz .
23 RUN tar -zxvf ./mattermost-team-5.18.0-rc1-linux-amd64.tar.gz
24 ADD config_docker.json ./mattermost/config/config_docker.json
25 ADD docker-entry.sh .
26
27 RUN chmod +x ./docker-entry.sh
28 ENTRYPOINT ./docker-entry.sh
29
30 # Mattermost environment variables
31 ENV PATH="/mm/mattermost/bin:${PATH}"
32
33 # Create default storage directory
34 RUN mkdir ./mattermost-data
35 VOLUME /mm/mattermost-data
36
37 # Ports
38 EXPOSE 8065
```

6. Simple Notification System and Lambda

The screenshot shows the Amazon SNS console for the topic 'capst-notification-service'. The left sidebar contains 'Dashboard', 'Topics', and 'Subscriptions'. The main content area has tabs for 'Details', 'Subscriptions', 'Access policy', 'Delivery retry policy (HTTP/S)', 'Delivery status logging', 'Encryption', and 'Tags'. The 'Details' tab shows the topic's name, ARN, display name, and owner. The 'Subscriptions' tab shows a table with one subscription.

ID	Endpoint	Status	Protocol
771f44c9-be0b-42f3-98de-afba8ad52410	arn:aws:lambda:us-east-2:121772939170:function:capst-Lambda	Confirmed	LAMBDA

The screenshot shows the Amazon SNS console for a specific subscription: 'Subscription: 771f44c9-be0b-42f3-98de-afba8ad52410'. The left sidebar is the same. The main content area has tabs for 'Subscription filter policy' and 'Redrive policy (dead-letter queue)'. The 'Subscription filter policy' tab shows the subscription's ARN, endpoint, topic, status, and protocol.

Subscription filter policy
This policy filters the messages that a subscriber receives. [Info](#)

No filter policy configured for this subscription.
To apply a filter policy, edit this subscription.

[Edit](#)

7. Amazon Container Services:

aws

Services

Resource Groups

CapstoneProjectTeam11

Ohio

Support

Amazon Container Services

Amazon ECS

Task definitions

Amazon EKS

Amazon ECR

Repositories

ECR

Repositories

Repositories (1)

Find repositories

View push commands

Delete

Edit

Create repository

1

Repository name

URI

Created at

Tag immutability

Scan on push

capstone-team11

121772939170.dkr.ecr.us-east-2.amazonaws.com/capstone-team11

12/18/19, 01:04:40 PM

Disabled

Disabled

aws

Services

Resource Groups

CapstoneProjectTeam11

Ohio

Support

Amazon ECS

Clusters

Task Definitions

Account Settings

Amazon EKS

Clusters

Amazon ECR

Repositories

AWS Marketplace

Discover software

Subscriptions

Clusters

default

Service: sample-app-service

Service : sample-app-service

Cluster default

Status ACTIVE

Task definition first-run-task-definition:1

Service type REPLICIA

Launch type FARGATE

Platform version LATEST(1.3.0)

Service role AWSServiceRoleForECS

Details

Tasks

Events

Auto Scaling

Deployments

Metrics

Tags

Logs

Load Balancing

Load Balancer Name

Container Name

Container Port

No load balancers

Network Access

Allowed VPC vpc-0ea22bb84513f1571

Allowed subnets subnet-0811f7770a8e3153a,subnet-01f7b669013e2ed21

Security groups* sg-01c5df1d7df43a79a

Auto-assign public IP ENABLED