

Cloud deployment – MovieApp Case study

RDS

Create aws RDS

The screenshot shows the AWS RDS console for the 'radhika-movie-db' database. The 'Connectivity & security' tab is active. Key details shown include:

- Endpoint & port:** Endpoint: radhika-movie-db.cn88qyk2qcw1.us-east-2.rds.amazonaws.com
- Networking:** Availability Zone: us-east-2a
- Security:** VPC security groups: default (sg-0b56e83af5ffc1f77) (Active)

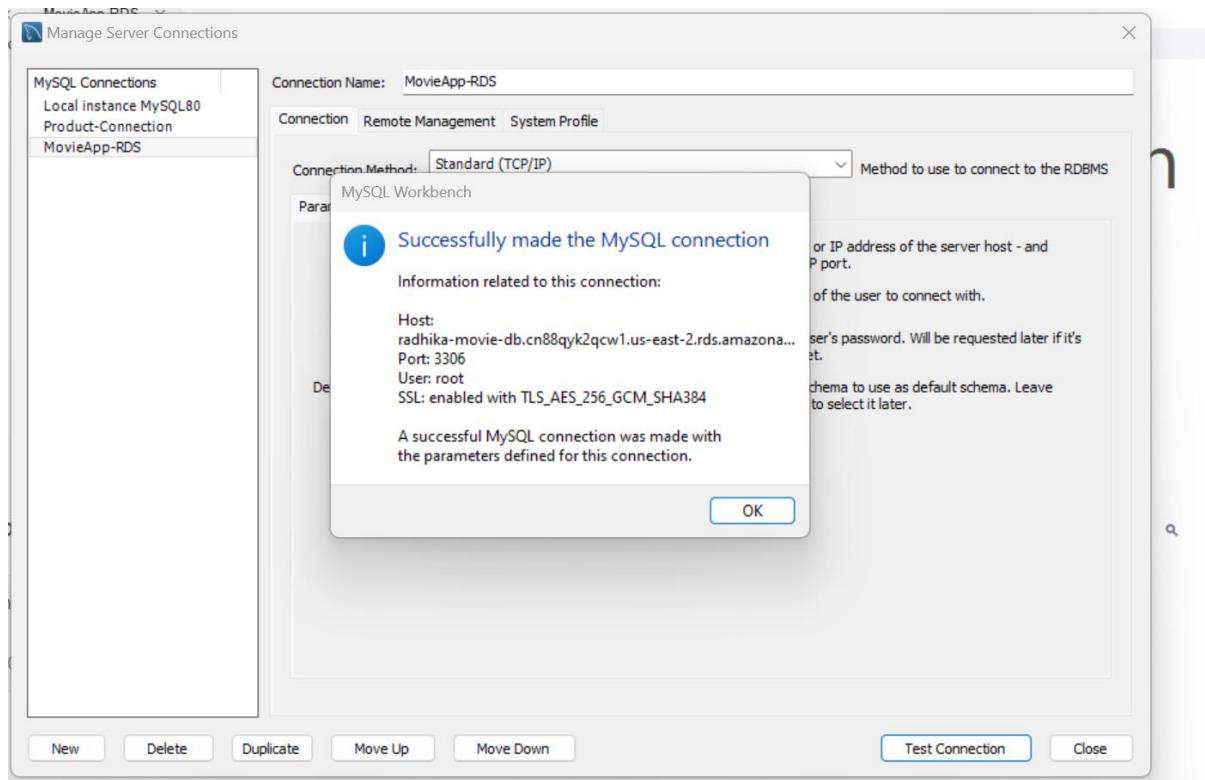
Edit inbound rules in security group

The screenshot shows the AWS Security Groups console for the default security group. The table lists the following inbound rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0a120ef957213424e	All traffic	All	All	Custom	0.0.0.0/0
sgr-098d80278bbe6747a	All traffic	All	All	Custom	sg-0b56e83af5ffc1f77
sgr-02a4d98b6f901e681	MySQL/Aurora	TCP	3306	Custom	0.0.0.0/0
sgr-0185cbea9f1951a75	Custom TCP	TCP	5000	Custom	0.0.0.0/0
sgr-01e7a878399f51d68	HTTP	TCP	80	Custom	0.0.0.0/0

Connect to RDS in MySQL Workbench

Use the endpoint from created RDS as Host and give username & password and test the connection



Add RDS endpoint in application.yml

Use the RDS endpoint in spring application instead of localhost

The screenshot shows the GitHub interface for the 'movieapp-configuration' repository, specifically the 'application.yml' file. The file content is as follows:

```
1 #use this for config server
2 eureka:
3   client:
4     serviceUrl:
5       defaultZone: http://radhika-eureka-ebs-env-1.eba-ueswdptm.us-east-1.elasticbeanstalk.com/eureka
6
7   spring:
8     datasource:
9       driver-class-name: com.mysql.cj.jdbc.Driver
10      username: root
11      url: jdbc:mysql://radhika-movie-db.cn8qyk2qcw1.us-east-2.rds.amazonaws.com:3306/movie_db
12      password: rootroot
13
14     jpa:
15       hibernate:
16         ddl-auto: update
17         show-sql: 'true'
```

Two specific sections of the code are highlighted with red boxes: the 'defaultZone' URL for Eureka and the 'url' for the MySQL datasource.

Create IAM Role

Create 2 IAM roles – EBS and EC2

EBS role

The screenshot shows the AWS IAM 'radhika-ebs-role' configuration page. The 'Summary' tab is selected, displaying basic information: Creation date (February 17, 2024, 23:29 UTC+05:30), Last activity (7 minutes ago), ARN (arn:aws:iam::676266512832:role/radhika-ebs-role), and Maximum session duration (1 hour). An instance profile ARN is also listed. The 'Permissions' tab is active, showing attached policies: AWSElasticBeanstalkCustomPlatformforEC2Role (AWS managed, 6 entities), AWSElasticBeanstalkEnhancedHealth (AWS managed, 76 entities), and AWSElasticBeanstalkManagedUpdatesCustomerRolePolicy (AWS managed, 68 entities). A search bar and filter options are present.

EC2 role

The screenshot shows the AWS IAM 'radhika-ec2-role' configuration page. The 'Summary' tab is selected, displaying basic information: Creation date (February 17, 2024, 20:23 UTC+05:30), Last activity (13 minutes ago), ARN (arn:aws:iam::676266512832:role/radhika-ec2-role), and Maximum session duration (1 hour). An instance profile ARN is also listed. The 'Permissions' tab is active, showing attached policies: AWSElasticBeanstalkMulticontainerDocker (AWS managed, 48 entities) and AWSElasticBeanstalkWebTier (AWS managed, 70 entities). A search bar and filter options are present.

Create ElasticBeanstalk environments for all microservices

Before creating ElasticBeanstalk environment, generate jar files for all spring microservices and update localhost url with ebs domain and upload jar files while creating EBS

Also update server.port to 5000 in application.yml

The screenshot shows a code editor with a dark theme. The file is `application.yml`. A red box highlights the line `import: optional:configserver:http://18.119.76.13:5000`, which is part of the `spring.config` section. The code is as follows:

```
1 server:
2   port: '5000'
3 topic:
4   name: t.user.details
5 management:
6   endpoints:
7     web:
8       exposure:
9         include: "*"
10 spring:
11   config:
12     import: optional:configserver:http://18.119.76.13:5000
13 application:
14   name: user-registration
15 doc:
16   swagger-ui:
17     path: /swagger-ui
18 devtools:
19   restart:
20     enabled: 'true'
```

The screenshot shows a code editor with a dark theme. The file is `MovieClient.java`. A red box highlights the line `@FeignClient(name = "movie-service", url = "http://radhika-movie-service-env.eba-arzcdmy6.us-east-2.elasticbeanstalk.com/movie")`. The code is as follows:

```
1 package com.movie.app.wishlist.WishlistMovieApp.clients;
2
3
4 import com.movie.app.wishlist.WishlistMovieApp.domain.MovieResponse;
5 import org.springframework.cloud.openfeign.FeignClient;
6 import org.springframework.http.ResponseEntity;
7 import org.springframework.web.bind.annotation.GetMapping;
8 import org.springframework.web.bind.annotation.PathVariable;
9
10 @usage + Radhika Kotain *
11 @FeignClient(name = "movie-service", url = "http://radhika-movie-service-env.eba-arzcdmy6.us-east-2.elasticbeanstalk.com/movie")
12 public interface MovieClient {
13   @GetMapping("/getMovieById/{id}")
14   public ResponseEntity<MovieResponse> getByTitle(@PathVariable("id") String id);
15 }
```

Steps to create EBS environment

The screenshot shows the 'Configure environment' step of the AWS Elastic Beanstalk setup. On the left, a sidebar lists steps from 1 to 6. Step 1 is 'Configure environment'. Step 2 is 'Configure service access'. Step 3 is 'optional: Set up networking, database, and tags'. Step 4 is 'optional: Configure instance traffic and scaling'. Step 5 is 'optional: Configure updates, monitoring, and logging'. Step 6 is 'Review'. The main panel is titled 'Configure environment' and contains three sections: 'Environment tier' (selected), 'Application information', and 'Environment information'. In the 'Environment tier' section, 'Web server environment' is selected. In the 'Application information' section, the application name is 'radhika-movie-user'. In the 'Environment information' section, the environment name is 'Radhika-movie-user-env'.

Select platform and version same as in spring pom.xml

The screenshot shows the 'Platform' configuration step. It includes fields for 'Platform type' (set to 'Managed platform'), 'Platform' (set to 'Java'), 'Platform branch' (set to 'Corretto 17 running on 64bit Amazon Linux 2023'), and 'Platform version' (set to '4.2.0 (Recommended)').

Upload jar file

Application code Info

Sample application
 Existing version
Application versions that you have uploaded.
 Upload your code
Upload a source bundle from your computer or copy one from Amazon S3.

Version label
Unique name for this version of your application code.

Source code origin. Maximum size 500 MB

Local file

Upload application

File name: **User-Service-0.0.1-SNAPSHOT.jar**
File must be less than 500MB max file size

Public S3 URL

Presets Info

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

Single instance (free tier eligible)
 Single instance (using spot instance)

Use ebs and ecs role which we created and create keypair

Configure service access Info

Service access
IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. [Learn more](#)

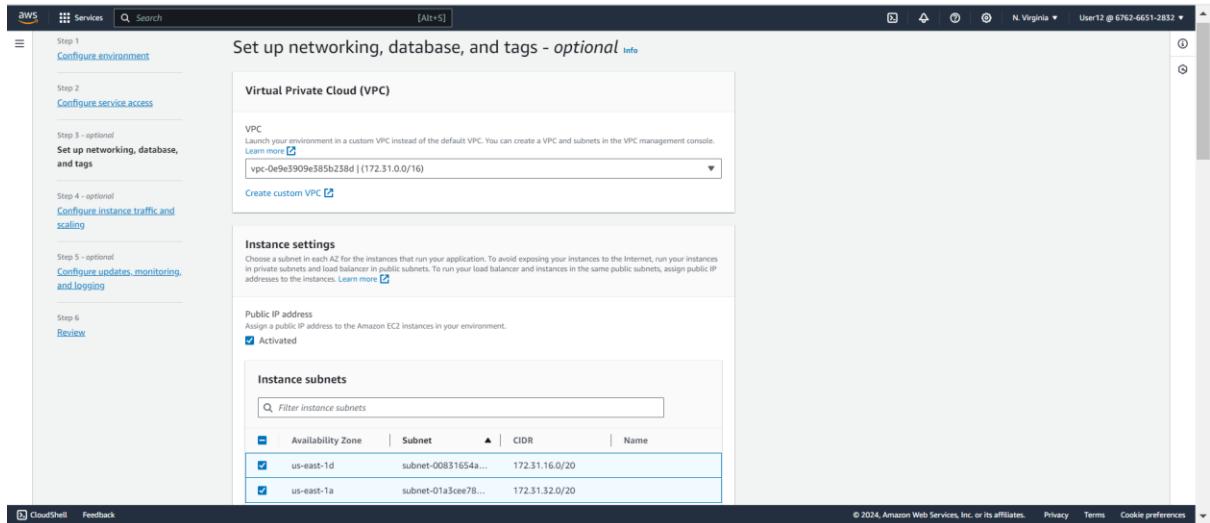
Service role
 Create and use new service role
 Use an existing service role
Existing service role
Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.

EC2 key pair
Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#)

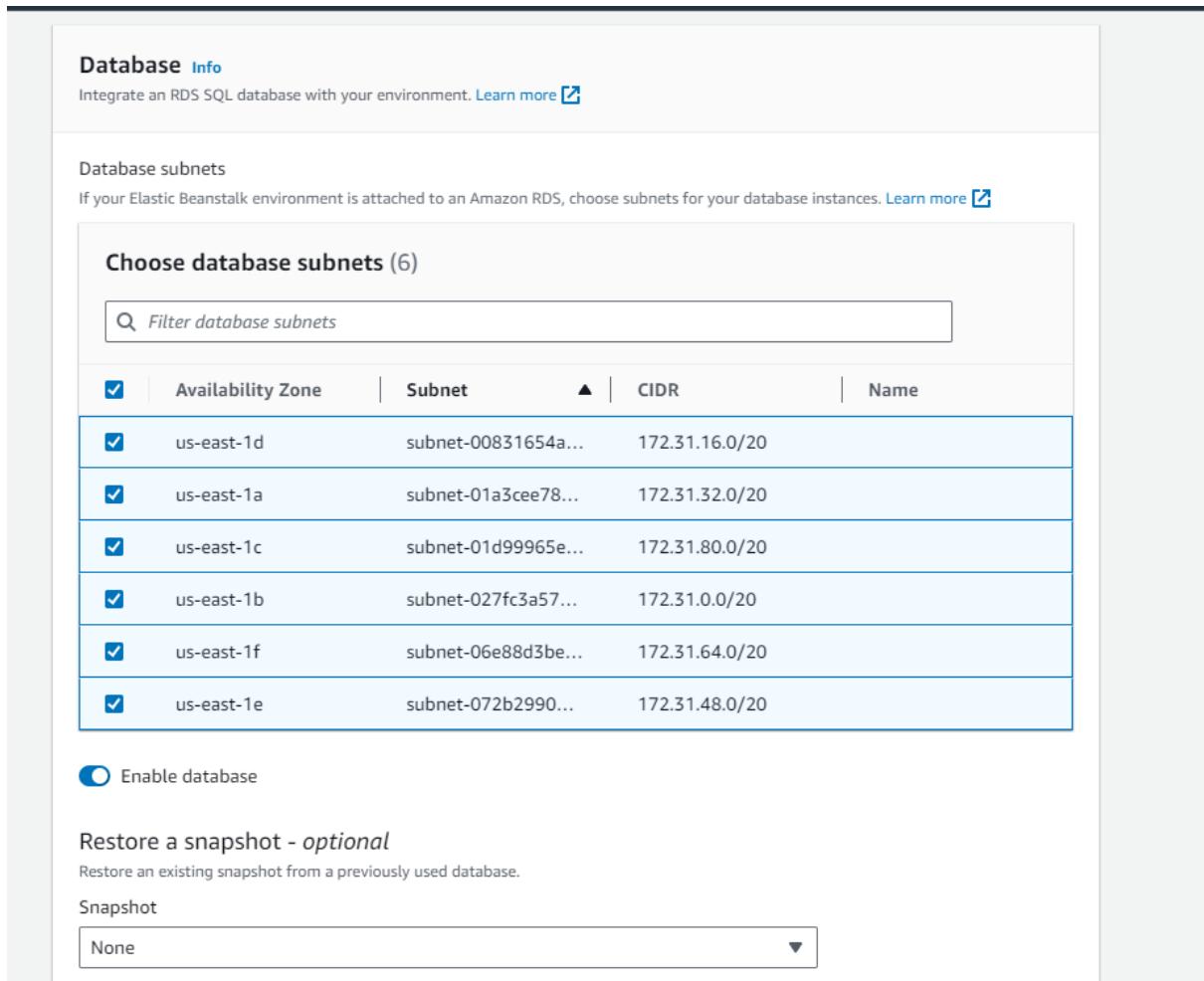
EC2 instance profile
Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

[View permission details](#)

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)



Enable database for microservices which required DB and use RDS which we created



Restore a snapshot - optional

Restore an existing snapshot from a previously used database.

Snapshot

None

Database settings

Choose an engine and instance type for your environment's database.

Engine

mysql

Engine version

8.0.35

Instance class

db.t2.micro

Storage

Choose a number between 5 GB and 1024 GB.

5 GB

Username

root

Password

Availability

Low (one AZ)

root

Password

Availability

Low (one AZ)

Database deletion policy

This policy applies when you decouple a database or terminate the environment coupled to it.

Create snapshot

Elastic Beanstalk saves a snapshot of the database and then deletes it. You can restore a database from a snapshot when you add a DB to an Elastic Beanstalk environment or when you create a new database. You might incur charges for storing database snapshots.

Retain

The decoupled database will remain available and operational external to Elastic Beanstalk.

Delete

Elastic Beanstalk terminates the database. The database will no longer be available.

Tags

Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. [Learn more](#)

No tags associated with the resource.

Add new tag

You can add 50 more tags.

Cancel Skip to review Previous Next

aws Services Search [Alt+S]

Configure instance traffic and scaling - optional [Info](#)

Step 1 [Configure environment](#)

Step 2 [Configure service access](#)

Step 3 - optional [Set up networking, database, and tags](#)

Step 4 - optional [Configure instance traffic and scaling](#)

Step 5 - optional [Configure updates, monitoring, and logging](#)

Step 6 [Review](#)

Instances [Info](#)
Configure the Amazon EC2 instances that run your application.

Root volume (boot device)

Root volume type: (Container default)

Size: 8 GB

IOPS: 100 IOPS

Throughput: 125 MiB/s

Amazon CloudWatch monitoring
The time interval between when metrics are reported from the EC2 instances

Monitoring interval: 5 minute

Select default security group and edit inbound rules

aws Services Search [Alt+S] N. Virginia User12 @ 6762-6651-2832

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

Deactivated

EC2 security groups
Select security groups to control traffic.

EC2 security groups (55)

Group name	Group ID	Name
default	sg-0a4da2f4b52da9a42	

aws Services Search [Alt+S] Ohio User12 @ 6762-6651-2832

Security group rule ID	Type Info	Protocol Info	Port range	Source Info	Description - optional Info
sgr-0120ef957213424e	All traffic	All	All	Custom	<input type="text"/> 0.0.0.0 X
sgr-098d8027bbe6747a	All traffic	All	All	Custom	<input type="text"/> sg-0b56e83af5ffcf1f77 X
sgr-02a4d98bb6f901e681	MySQL/Aurora	TCP	3306	Custom	<input type="text"/> 0.0.0.0 X
sgr-0185cbea9f1951a75	Custom TCP	TCP	5000	Custom	<input type="text"/> 0.0.0.0 X
sgr-01e7a878399f31d68	HTTP	TCP	80	Custom	<input type="text"/> 0.0.0.0 X

[Add rule](#)

The screenshot shows the AWS Elastic Beanstalk configuration interface. On the left, a sidebar lists steps: Step 1 (Configure environment), Step 2 (Configure service access), Step 3 (optional: Set up networking, database, and tags), Step 4 (optional: Configure instance traffic and scaling), Step 5 (optional: Configure updates, monitoring, and logging), and Step 6 (Review). The current step is Step 5. The main panel title is "Configure updates, monitoring, and logging - optional". It contains sections for "Monitoring" (with "Enhanced" selected), "CloudWatch Custom Metrics" for Instance and Environment, and "Health event streaming to CloudWatch Logs" (with Log streaming activated and a retention of 7 days). A note at the bottom says "Lifecycle".

Add SERVER_PORT = 5000 and port should be same in application.yml file

The screenshot shows the "Configure environment properties" step of the AWS Elastic Beanstalk configuration wizard. It includes sections for "Log streaming" (activated with a 7-day retention), "Lifecycle" (keep logs after terminating environment), and "Environment properties". The "Environment properties" section lists four properties: GRADLE_HOME (value /usr/local/gradle), M2 (value /usr/local/apache-maven/bin), M2_HOME (value /usr/local/apache-maven), and SERVER_PORT (value 5000). There is also an "Add environment property" button. At the bottom are "Cancel", "Previous", and "Next" buttons, with "Next" highlighted.

Review and submit

AWS Services Search [Alt+S] N. Virginia User12 @ 6702-6651-2832

Step 1 Review Step 1: Configure environment

Configure environment Step 2 Configure service access Step 3 - optional Set up networking, database, and tags Step 4 - optional Configure instance traffic and scaling Step 5 - optional Configure updates, monitoring, and logging Step 6 Review

Step 1: Configure environment

Environment information

Environment tier	Application name
Web server environment	radhika-movie-user
Environment name	Application code
Radhika-movie-user-env	User-Service-0.0.1-SNAPSHOT.jar
Platform	arn:aws:elasticbeanstalk:us-east-1:platform/Corretto 17 running on 64bit Amazon Linux 2023.4.20

Step 2: Configure service access

Service access

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Service role	EC2 key pair	EC2 instance profile
arn:aws:iam::676266512832:role/radhika-user-perm	radhika-user-perm	radhika-ec2-role
hika-eks-role		

AWS Services Search [Alt+S] N. Virginia User12 @ 6702-6651-2832

Managed updates Deployment batch size Deployment batch size type
Activated 100 Percentage
Command timeout Deployment policy Health threshold
600 AllAtOnce Ok
Ignore health check Instance replacement
false false
Platform software
Lifecycle Log streaming Logs retention
false Deactivated 7
Rotate logs Update level X-Ray enabled
Deactivated minor Deactivated
Environment properties

Key	Value
GRADLE_HOME	/usr/local/gradle
M2	/usr/local/apache-maven/bin
M2_HOME	/usr/local/apache-maven
SERVER_PORT	5000

Cancel Previous Submit

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EBS environments for all microservices

The screenshot shows the AWS Elastic Beanstalk console with the URL us-east-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-2#/environments. The page displays a table titled "Environments (21) Info" with 6 matches found for the search term "radhika". The columns include Environment name, Health, Application name, Platform, Domain, Running status, Tier name, Date created, and Last modified. All environments listed are in an "Ok" state. The environments are:

Environment name	Health	Application name	Platform	Domain	Running status	Tier name	Date created	Last modified
Radhika-config-env	Ok	radhika-co...	Corretto 17...	Radhika-config-env.eba-dzqmc...	radhika-co...	WebServer	February 1...	February 18, 2...
Radhika-movie-auth-env	Ok	radhika-mo...	Corretto 17...	Radhika-movie-auth-env.eba-j...	radhika-mo...	WebServer	February 1...	February 18, 2...
Radhika-movie-gateway-env	Ok	radhika-mo...	Corretto 17...	Radhika-movie-gateway-env.e...	radhika-mo...	WebServer	February 1...	February 18, 2...
Radhika-movie-service-env	Ok	radhika-mo...	Corretto 17...	Radhika-movie-service-env.eb...	radhika-mo...	WebServer	February 1...	February 18, 2...
Radhika-movie-user-env	Ok	radhika-mo...	Corretto 17...	Radhika-movie-user-env.eba-g...	radhika-mo...	WebServer	February 1...	February 18, 2...
Radhika-movie-wishlist-env	Ok	radhika-mo...	Corretto 17...	Radhika-movie-wishlist-env.eb...	radhika-mo...	WebServer	February 1...	February 18, 2...

EBS for eureka microservice

The screenshot shows the AWS Elastic Beanstalk console with the URL <https://us-east-2.console.aws.amazon.com/elasticbeanstalk/environments/Radhika-eureka-ebs-env-1>. A green banner at the top indicates "Environment successfully launched". The page displays the "Radhika-eureka-ebs-env-1 Info" page. The "Events" tab is selected, showing 13 events. The "Health" section shows "Ok". The "Platform" section shows "Corretto 17 running on 64bit Amazon Linux 2023/4.2.0" and "Supported" platform state.

All services registered with eureka

The screenshot shows the Eureka service registration interface. At the top, it displays environment details: Environment is 'test', Data center is 'default', Current time is '2024-02-18T10:06:30 +0000', Uptime is '15:48', Lease expiration enabled is 'false', Renews threshold is '10', and Renews (last min) is '10'. A red warning message states: 'EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE.' Below this, the 'DS Replicas' section shows 'localhost' as the host. The main table lists registered applications:

Application	AMIs	Availability Zones	Status
LOGIN-SERVICE	n/a (1)	(1)	UP (1) - ip-172-31-4-62.us-east-2.compute.internal:login-service:5000
MOVIE-CONFIG-SERVER	n/a (1)	(1)	UP (1) - ip-172-31-14-211.us-east-2.compute.internal:movie-config-server:5000
MOVIEAPP-API-GATEWAY	n/a (1)	(1)	UP (1) - ip-172-31-14-230.us-east-2.compute.internal:movieapp-api-gateway:5000
USER-REGISTRATION	n/a (1)	(1)	UP (1) - ip-172-31-3-130.us-east-2.compute.internal:user-registration:5000
WISHLIST-SERVICE	n/a (1)	(1)	UP (1) - ip-172-31-6-101.us-east-2.compute.internal:wishlist-service:5000

EBS for config server

The screenshot shows the AWS Elastic Beanstalk console for the environment 'Radhika-config-env'. The 'Environment overview' section displays the following information:

Health	Environment ID
Ok	e-k8tgguzesn

Domain: Radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com

The 'Platform' section shows:

Platform	Change version
Corretto 17 running on 64bit Amazon Linux 2023/4.2.0	

Running version: radhika-config

Platform state: Supported

Below the overview, there are tabs for Events, Health, Logs, Monitoring, Alarms, Managed updates, and Tags. The Events tab shows 13 events, with a search bar and filters for Time, Type, and Details.

Click on ebs domain to check application is working or not

```

{
  "_links": {
    "self": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator", "templated": false},
    "beans": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/beans", "templated": false},
    "caches-cache": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/caches/{cache}", "templated": true},
    "caches": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/caches", "templated": false},
    "health": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/health", "templated": false},
    "health-path": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/health/{path}", "templated": true},
    "info": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/info", "templated": false},
    "conditions": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/configprops/{prefix}", "templated": false},
    "configprops": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/configprops/{prefix}", "templated": true},
    "env-to-latch": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/env/{Match}", "templated": false},
    "env": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/env", "templated": false},
    "loggers": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/loggers", "templated": false},
    "loggers-name": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/loggers/{name}", "templated": true},
    "headdump": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/headdump", "templated": false},
    "theadump": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/threaddump", "templated": false},
    "metrics-requiredMetricName": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/metrics/{requiredMetricName}", "templated": true},
    "metrics": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/metrics", "templated": false},
    "scheduledtasks": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/scheduledtasks", "templated": false},
    "mappings": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/mappings", "templated": false},
    "refresh": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/refresh", "templated": false},
    "features": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/features", "templated": false},
    "serviceregistry": {"href": "http://radhika-config-env.eba-dzqmrzpt.us-east-2.elasticbeanstalk.com/actuator/serviceregistry", "templated": false}
  }
}

```

EBS for Apigateway service

Environment overview

Health	Environment ID
Ok	e-b3umtaa2ez
Domain	Application name
Radhika-movie-gateway-env.eba-2d2vdjh.us-east-2.elasticbeanstalk.com	radhika-movie-gateway

Platform

Platform	Corretto 17 running on 64bit Amazon Linux 2023/4.2.0
Running version	radhika-movie-gateway
Platform state	Supported

Events (13)

Time	Type	Details
...

```

{
  "_links": {
    "self": {"href": "http://radhika-movie-gateway-env.eba-2d2vdjh.us-east-2.elasticbeanstalk.com/actuator", "templated": false},
    "health": {"href": "http://radhika-movie-gateway-env.eba-2d2vdjh.us-east-2.elasticbeanstalk.com/actuator/health", "templated": false},
    "health-path": {"href": "http://radhika-movie-gateway-env.eba-2d2vdjh.us-east-2.elasticbeanstalk.com/actuator/health/{path}", "templated": true}
  }
}

```

EBS for User-service

The screenshot shows the AWS Elastic Beanstalk console with the environment 'Radhika-movie-user-env' selected. The interface includes sections for 'Environment overview', 'Platform', and 'Events'. The 'Events' section displays 58 events, with a search bar and pagination controls.

Environment overview:

- Health: Ok
- Domain: Radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com
- Environment ID: e-s9yzdqnmvt
- Application name: radhika-movie-user

Platform:

- Platform: Corretto 17 running on 64bit Amazon Linux 2023/4.2.0
- Running version: radhika-movie-user-1
- Platform state: Supported

Events: (58) Info

Filter events by text, property or value

Time Type Details

While creating user ebs add database configuration to it

The screenshot shows the 'Networking and database' configuration page for the environment. It includes sections for 'Network' and 'Database' settings.

Network:

- VPC: vpc-0bcd7da0303f4895a
- Public IP address: true
- Instance subnets: subnet-0ab02c73975736eb4, subnet-04e9b0d5de1a42241

Database:

- Database availability: false
- Has coupled database: true
- Database deletion policy: Snapshot
- Database engine: mysql
- Database engine version: 8.0.35
- Database instance class: db.t2.micro
- Database password: *****
- Database storage: 5
- Database subnets: subnet-0ab02c73975736eb4, subnet-04e9b0d5de1a42241, subnet-0ac7bbe1cf7df7933
- Database username: root

User service swagger

The screenshot shows the Movie App user registration service API documentation on Swagger UI. The title is "Movie App user registration service API". It states, "This API provides endpoints for users to register." Below this, there is a "user-controller" section for the "/movie/register" endpoint. The method is POST, and the URL is /movie/register. The "Parameters" section indicates "No parameters". The "Request body" section shows a schema for a user object with properties: name (string), email (string), and password (string). The "Responses" section shows a 200 OK response with "application/json" content type.

The screenshot shows the actuator endpoint documentation on Swagger UI. The URL is /actuator. The response body is a JSON object containing various links to actuator endpoints, such as beans, health, info, conditions, configprops, prefix, env, toMatch, loggers, loggers-name, loggers-(name), heapdump, threaddump, metrics-requiredMetricName, scheduledtasks, mappings, refresh, features, and serviceregistry.

```
{"links":{ "self":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator", "templated": false}, "beans":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/beans", "templated": false}, "caches-cache":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/caches", "templated": true}, "caches":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/caches", "templated": false}, "health":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/health", "templated": false}, "info":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/info", "templated": false}, "conditions":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/conditions", "templated": false}, "configprops":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/configprops", "templated": false}, "configprops-prefix":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/configprops/{prefix}", "templated": true}, "env":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/env", "templated": false}, "env-toMatch":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/env/{toMatch}", "templated": true}, "loggers":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/loggers", "templated": false}, "loggers-name":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/loggers/{name}", "templated": true}, "heapdump":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/heapdump", "templated": false}, "threaddump":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/threaddump", "templated": false}, "metrics-requiredMetricName":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/metrics/{requiredMetricName}", "templated": true}, "metrics":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/metrics", "templated": false}, "scheduledtasks":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/scheduledtasks", "templated": false}, "mappings":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/mappings", "templated": false}, "refresh":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/refresh", "templated": false}, "features":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/features", "templated": false}, "serviceregistry":{ "href": "http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/actuator/serviceregistry", "templated": false}}}}
```

User service tested in postman using ebs domain

The screenshot shows a Postman collection named "User-Registration / add-user". A red box highlights the URL field containing the endpoint: `http://radhika-movie-user-env.eba-guuu3guk.us-east-2.elasticbeanstalk.com/movie/register`. The request method is set to POST. The body is defined as JSON, containing the following data:

```
1 {
2   "userName": "arjun",
3   "email": "arjun@gmail",
4   "fullname": "arjun",
5   "password": "pass"
6 }
```

The response status is 409 Conflict, with the message: "User Already Exists with id : arjun", "Message": "User already exists".

EBS for Authentication service

The screenshot shows the AWS Elastic Beanstalk environment Radhika-movie-auth-env. The environment overview section shows the Health status as "Ok" and the Platform as "Corretto 17 running on 64bit Amazon Linux 2023/4.2.0". The Events section shows 27 events, with the first few being system-related messages.

The screenshot shows a browser window displaying the Actuator endpoint for the Radhika-movie-auth-env. The URL is `radhika-movie-auth-env.eba-jvgmnpjp.us-east-2.elasticbeanstalk.com/actuator`. The page lists various links and metrics, including the "links" section which contains a self-link to the actuator and a link to the health endpoint.

```
{"_links": {"self": {"href": "http://radhika-movie-auth-env.eba-jvgmnpjp.us-east-2.elasticbeanstalk.com/actuator", "templated": false}, "health": {"href": "http://radhika-movie-auth-env.eba-jvgmnpjp.us-east-2.elasticbeanstalk.com/actuator/health", "templated": false}, "health-path": {"href": "http://radhika-movie-auth-env.eba-jvgmnpjp.us-east-2.elasticbeanstalk.com/actuator/health/{path}", "templated": true}}}
```

Authentication in postman

Checking login

The screenshot shows a Postman collection named "User-Registration / login". A POST request is made to <http://radhika-movie-auth-env.eba-jvgmnpjp.us-east-2.elasticbeanstalk.com/movie/login>. The Body tab contains the following JSON:

```
1 {
2   .... "username" : "radhika",
3   .... "password" : "P@ssw0rd"
4 }
```

The response status is 200 OK, time: 1539 ms, size: 489 B. The response body is:

```
1 {
2   "message": "Login Successful",
3   "token": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJyYWRoawthIiwidXNlck5hbWUiOiJyYWRoawthIiwiakF0IjoxNzA4MjUxNDU3LCJleHAiOjE3MDgzMzc4NTd9.PUmDacLAbwz5uIS95wSeERSn5-6LN7z_s18K-5WXSaSnBNGaow10_JVhAS87yVA3ql4RcpfEk52pxn9soN2I7w"
```

Validating token

The screenshot shows a Postman collection named "User-Registration / validate". A GET request is made to <http://radhika-movie-auth-env.eba-jvgmnpjp.us-east-2.elasticbeanstalk.com/movie/validate>. The Authorization tab is selected, showing "Bearer ...". A note says: "Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about [variables](#)." The Token field contains a long JWT token.

The response status is 200 OK, time: 811 ms, size: 297 B. The response body is:

```
1 Valid token AuthResponse(username=null, isValid=true)
```

EBS for Movie service

Environment overview

Health: Ok

Environment ID: e-6yg2xxzm6c

Domain: Radhika-movie-service-env.eba-arzcdmy6.us-east-2.elasticbeanstalk.com

Application name: radhika-movie-service

Platform: Corretto 17 running on 64bit Amazon Linux 2023/4.2.0

Running version: radhika-movie-service

Platform state: Supported

Events (13) Info

Events (13) Info

Filter events by text, property or value

Time | Type | Details

```
[{"rank":1,"title":"The Shawshank Redemption","description":"Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.","image":"https://m.media-amazon.com/images/M/MVSBMDFKTYC0rGETZmHnB002D1zLWFmTET0OM1zRlYhMwMwFmXxEyXkFqcGdeQxVvYMTMxDk20TU@._V1_QL75_UX380_CR0,1,380,562_.jpg","big_image":"https://m.media-amazon.com/images/M/MVSBMDFKTYC0rGETZmHnC002D1zLWFmTET0OM1zRlYhMwMwFmXxEyXkFqcGdeQxVvYMTMxDk20TU@._V1_QL75_UX380_CR0,1,380,562_.jpg","genre":["Drama"],"thumbnail":"https://m.media-amazon.com/images/M/MVSBMDFKTYC0rGETZmHnC002D1zLWFmTET0OM1zRlYhMwMwFmXxEyXkFqcGdeQxVvYMTMxDk20TU@._V1_UV67_CR0,0,45,67_AL_.jpg","rating":"9.3","id":"top1","year":1994,"imdb_id":"tt0111161","imdb_link":"https://www.imdb.com/title/tt0111161"}, {"rank":2,"title":"The Godfather","description":"The aging patriarch of an organized crime dynasty in postwar New York City transfers control of his clandestine empire to his reluctant youngest son.","image":"https://m.media-amazon.com/images/M/MVSBM20MyWjYxhmltYTAWli0@ITXWJmWYtyzLzL0Dy3ZTk30TF1XxEyXkFqcGdeQxVvNzkwhjQ5Nz@._V1_QL75_UV562_CR8,0,380,562_.jpg","big_image":"https://m.media-amazon.com/images/M/MVSBM20MyWjYxhmltYTAWli0@ITXWJmWYtyzLzL0Dy3ZTk30TF1XxEyXkFqcGdeQxVvNzkwhjQ5Nz@._V1_QL75_UV562_CR8,0,380,562_.jpg","genre":["Crime","Drama"],"thumbnail":"https://m.media-amazon.com/images/M/MVSBM20MyWjYxhmltYTAWli0@ITXWJmWYtyzLzL0Dy3ZTk30TF1XxEyXkFqcGdeQxVvNzkwhjQ5Nz@._V1_UV67_CR1,0,45,67_AL_.jpg","rating":"9.2","id":"top2","year":1972,"imdb_id":"tt0068646","imdb_link":"https://www.imdb.com/title/tt0068646"}, {"rank":3,"title":"The Dark Knight","description":"When the menace known as the Joker wreaks havoc and chaos on the people of Gotham, Batman must accept one of the greatest psychological and physical tests of his ability to fight injustice.","image":"https://m.media-amazon.com/images/M/MVSBMTHmNThwODh0Mf5Bm158anBnxKfzTcw0DA@MTK2h@@._V1_QL75_UX380_CR0,0,380,562_.jpg","big_image":"https://m.media-amazon.com/images/M/MVSBMTHmNThwODh0Mf5Bm158anBnxKfzTcw0DA@MTK2h@@._V1_QL75_UX380_CR0,0,380,562_.jpg","genre":["Action","Crime","Drama"],"thumbnail":"https://m.media-amazon.com/images/M/MVSBMTHmNThwODh0Mf5Bm158anBnxKfzTcw0DA@MTK2h@@._V1_UK61_CR0,0,67,98_AL_.jpg","rating":"9.0","id":"top3","year":2008,"imdb_id":"tt0468569","imdb_link":"https://www.imdb.com/title/tt0468569"}, {"rank":4,"title":"The Godfather Part II","description":"The early life and career of Vito Corleone in 1920s New York City is portrayed, while his son, Michael, expands and tightens his grip on the family crime syndicate.","image":"https://m.media-amazon.com/images/M/MVSBMwMwMlgQz21Ttv21NC0002zilWjyM0ctDk2ZDQ2YjRjMwQ0XxEyXkFqcGdeQxVvNzkwhjQ5Nz@._V1_QL75_UV562_CR7,0,380,562_.jpg","big_image":"https://m.media-amazon.com/images/M/MVSBMwMwMlgQz21Ttv21NC0002zilWjyM0ctDk2ZDQ2YjRjMwQ0XxEyXkFqcGdeQxVvNzkwhjQ5Nz@._V1_QL75_UV562_CR7,0,380,562_.jpg","genre":["Crime","Drama"],"thumbnail":"https://m.media-amazon.com/images/M/MVSBMwMwMlgQz21Ttv21NC0002zilWjyM0ctDk2ZDQ2YjRjMwQ0XxEyXkFqcGdeQxVvNzkwhjQ5Nz@._V1_UY67_CR1,0,45,67_AL_.jpg","rating":"9.0","id":"top4","year":1974,"imdb_id":"tt0071562","imdb_link":"https://www.imdb.com/title/tt0071562"}, {"rank":5,"title":"12 Angry Men","description":"The jury in a New York City murder trial is frustrated by a single member whose skeptical caution forces them to more carefully consider the evidence before jumping to a hasty verdict.","image":"https://m.media-amazon.com/images/M/MVSBMwU4Nf2fjhNzYtVhKc00n2zQLTg0@JAtYTJ1MfHNGUxZDFmXxEyXkFqcGdeQxVvNjciNTYyjig@._V1_QL75_UX380_CR0,11,380,562_.jpg","big_image":"https://m.media-amazon.com/images/M/MVSBMwU4Nf2fjhNzYtVhKc00n2zQLTg0@JAtYTJ1MfHNGUxZDFmXxEyXkFqcGdeQxVvNjciNTYyjig@._V1_QL75_UX380_CR0,11,380,562_.jpg","genre":["Crime","Drama"],"thumbnail":"https://m.media-amazon.com/images/M/MVSBMwU4Nf2fjhNzYtVhKc00n2zQLTg0@JAtYTJ1MfHNGUxZDFmXxEyXkFqcGdeQxVvNjciNTYyjig@._V1_UY67_CR1,0,45,67_AL_.jpg","rating":"9.0","id":"top5","year":1957,"imdb_id":"tt0050083","imdb_link":"https://www.imdb.com/title/tt0050083"}, {"rank":6,"title":"Schindler's List","description":"In German-occupied Poland during World War II, industrialist Oskar Schindler gradually becomes concerned for his Jewish work-force after witnessing their persecution by the Nazis.","image":"https://m.media-amazon.com/images/M/MVSBNDE4OTHxItcthmRhY0@NE2Lg3YzItyTK3M2luOTUSNjg4XxEyXkFqcGdeQxVvNjU0eTO0OT@._V1_QL75_UX380_CR0,4,380,562_.jpg","big_image":"https://m.media-amazon.com/images/M/MVSBNDE4OTHxItcthmRhY0@NE2Lg3YzItyTK3M2luOTUSNjg4XxEyXkFqcGdeQxVvNjU0eTO0OT@._V1_QL75_UX380_CR0,4,380,562_.jpg","genre":["Biography","Drama","History"],"thumbnail":"https://m.media-amazon.com/images/M/MVSBNDE4OTHxItcthmRhY0@NE2Lg3YzItyTK3M2luOTUSNjg4XxEyXkFqcGdeQxVvNjU0eTO0OT@._V1_UX45_CR0,0,45,67_AL_.jpg","rating":"9.0","id":"top6","year":1993,"imdb_id":"tt0108052","imdb_link":"https://www.imdb.com/title/tt0108052"}, {"rank":7,"title":"The Lord of the Rings: The Return of the King","description":"Gandalf and Aragorn lead
```

Postman testing

getTop100 movies

The screenshot shows the Postman interface with a successful response for the 'getTop100' endpoint. The response body is a JSON object representing the top movie in the list:

```
1 {
2   "rank": 1,
3   "title": "The Shawshank Redemption",
4   "description": "Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.",
5   "image": "https://m.media-amazon.com/images/M/MV5BMDFkYTc0MGEtZmNhMC00ZDizLWFmNTEtODM1ZmR1YWMwMWFiXkEyXkFqcGdeQXVyMTMxODk2OTU@.V1_QL75_UX380_CR0,1,380,562_.jpg",
6   "big_image": "https://m.media-amazon.com/images/M/MV5BMDFkYTc0MGEtZmNhMC00ZDizLWFmNTEtODM1ZmR1YWMwMWFiXkEyXkFqcGdeQXVyMTMxODk2OTU@",
7   "genre": [
8     "Drama"
9   ],
10  "thumbnail": "https://m.media-amazon.com/images/M/MV5BMDFkYTc0MGEtZmNhMC00ZDizLWFmNTEtODM1ZmR1YWMwMWFiXkEyXkFqcGdeQXVyMTMxODk2OTU@.V1_UY67_CR0,0,45,67_AL_.jpg",
11  "rating": "9.3",
12  "id": "top1"
```

getMovieById

The screenshot shows the Postman interface with a successful response for the 'getMovieById' endpoint. The response body is a JSON object representing the movie with rank 10:

```
1 {
2   "rank": 10,
3   "title": "The Good, the Bad and the Ugly",
4   "description": "A bounty hunting scam joins two men in an uneasy alliance against a third in a race to find a fortune in gold buried in a remote cemetery.",
5   "image": "https://m.media-amazon.com/images/M/MV5BNjIyMkZGItM2NhYy00Mj1mLTk5NmQtNjg1NmM20DU40TMwXkEyXkFqcGdeQXVyMjUzOTY1NTc@.V1_QL75_UX380_CR0,4,380,562_.jpg",
6   "big_image": "https://m.media-amazon.com/images/M/MV5BNjIyMkZGItM2NhYy00Mj1mLTk5NmQtNjg1NmM20DU40TMwXkEyXkFqcGdeQXVyMjUzOTY1NTc@.V1_QL75_UX380_CR0,4,380,562_.jpg",
7   "genre": [
8     "Adventure",
9     "Western"
10  ]
```

EBS for Wishlist service

The screenshot shows the AWS Elastic Beanstalk console for the environment "Radhika-movie-wishlist-env". The "Events" tab is active, displaying 17 events. The first event is a deployment log:

```
2023-09-12T10:30:00+00:00 [Deployment] INFO Application version v1 is deployed to environment Radhika-movie-wishlist-env.
```

The screenshot shows a web browser displaying the actuator endpoint for the Wishlist service. The URL is `http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/actuator`. The page lists several links under the "_links" section:

```
{"_links": {"self": {"href": "http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/actuator"}, "templated": false}, "health": {"href": "http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/actuator/health"}, "templated": false}, "health-path": {"href": "http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/actuator/health/{path}"}, "templated": true}}
```

Postman testing

Add to wishlist

The screenshot shows a Postman collection named "wishlist / AddmovieToWishlist". A GET request is made to the endpoint `http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/movie/wishlist/add/radhika/top20`. The response status is 201 Created, and the response body is a JSON object:

```
1 {  
2     "Status": "CREATED",  
3     "Response": {  
4         "id": 3,  
5         "username": "radhika",  
6         "email": null,  
7         "movieId": "top20"  
8     },  
9     "Message": "movie added to watchlist"  
10 }
```

Get all movie list based on username

The screenshot shows a Postman request for 'wishlist / getall'. The URL is <http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/movie/wishlist/all>. The request method is GET, and the response status is 200 OK with a time of 322 ms and a size of 1.84 KB. The response body is a JSON object representing a movie, with lines numbered 12 through 22. The JSON includes fields like rating, id, year, imdbid, imdb_link, rank, title, description, and image.

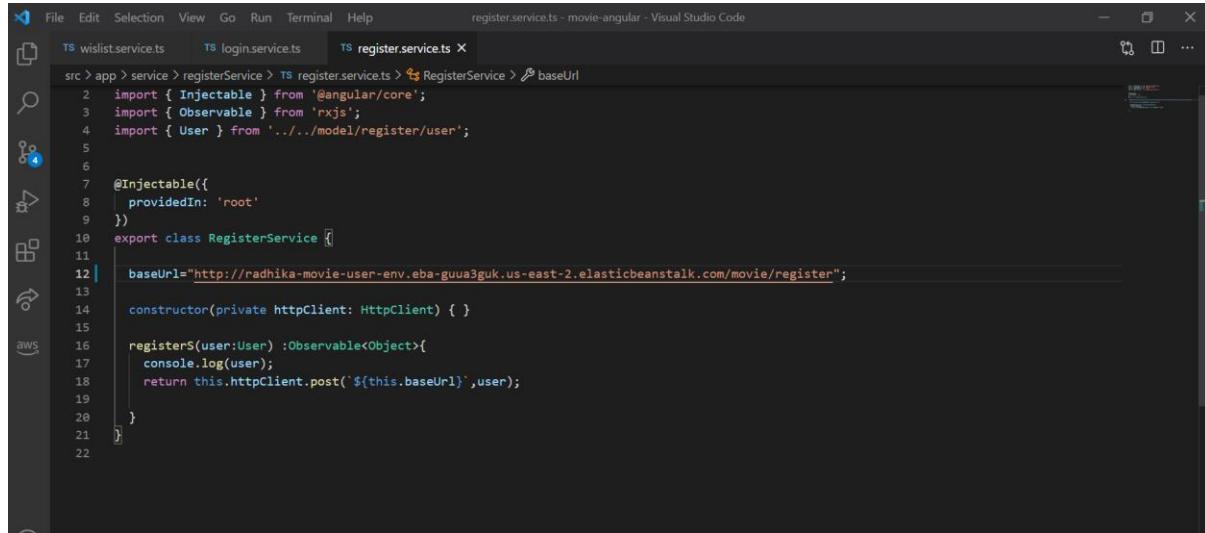
```
12     "rating": "8.6",
13     "id": "top20",
14     "year": 1975,
15     "imdbid": "tt0073486",
16     "imdb_link": "https://www.imdb.com/title/tt0073486"
17 },
18 {
19     "rank": 1,
20     "title": "The Shawshank Redemption",
21     "description": "Two imprisoned men bond over a number of years, finding solace and eventual
redemption through acts of common decency.",
22     "image": "https://m.media-amazon.com/images/M/
MV5BMDFkYTc0MGEtZmNhMC00ZDItLWFmNTEtODM1ZmR1YWMwMWFnXkEyXkFqcGdeQXVyMTMxODk2OTU@..
```

Delete movie from wishlist by using username and movie id to be deleted

The screenshot shows a Postman request for 'wishlist / delete'. The URL is <http://radhika-movie-wishlist-env.eba-r9fq5rpk.us-east-2.elasticbeanstalk.com/movie/wishlist/delete/radhika/top1>. The request method is DELETE, and the response status is 200 OK with a time of 366 ms and a size of 269 B. The response body is a simple message: '1 top1 removed from wishlist'.

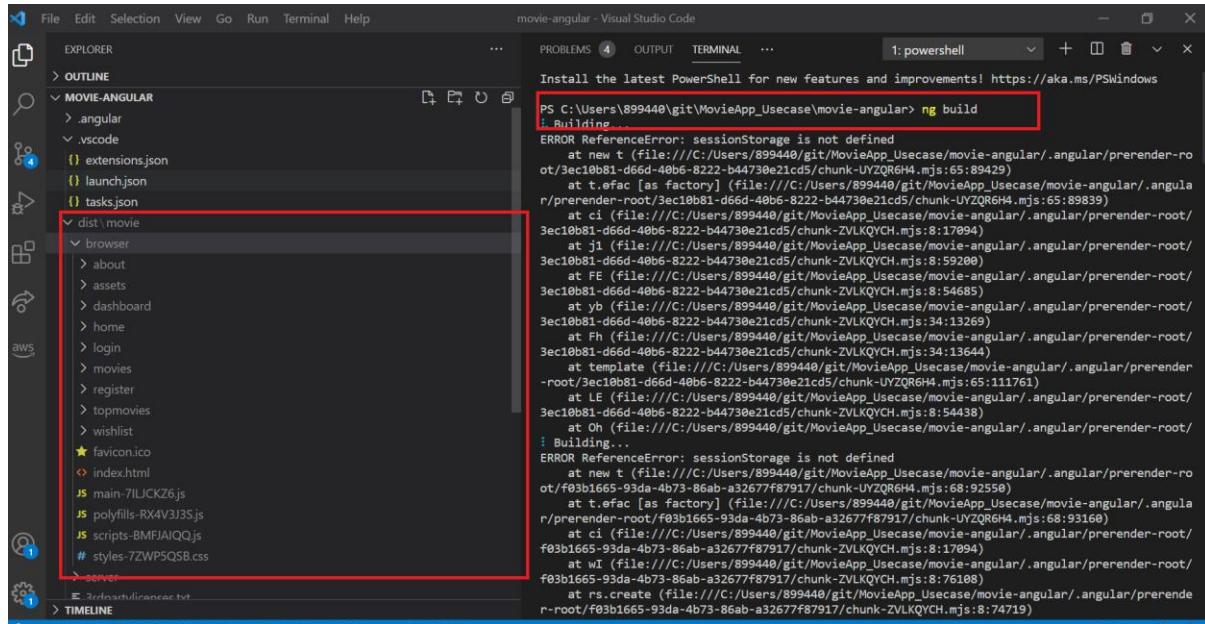
Angular

Update localhost urls with ebs



```
src > app > service > registerService > register.service.ts - RegisterService > baseU...  
1 import { Injectable } from '@angular/core';  
2 import { Observable } from 'rxjs';  
3 import { User } from '../../../../../model/register/user';  
4  
5 @Injectable({  
6   providedIn: 'root'  
7 })  
8 export class RegisterService {  
9  
10   baseUrl="http://radhika-movie-user-env.eba-guua3guk.us-east-2.elasticbeanstalk.com/movie/register";  
11  
12   constructor(private httpClient: HttpClient) { }  
13  
14   registerS(user:User) :Observable<Object>{  
15     console.log(user);  
16     return this.httpClient.post(` ${this.baseUrl} `,user);  
17   }  
18 }  
19  
20  
21 }  
22
```

Give ng build to generate dist folder



```
File Edit Selection View Go Run Terminal Help movie-angular - Visual Studio Code  
PROBLEMS 4 OUTPUT TERMINAL ... 1: powershell v + ☰ ×  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
PS C:\Users\899440\git\MovieApp_Usecase\movie-angular> ng build  
Building...  
ERROR ReferenceError: sessionStorage is not defined  
  at new t (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:65:89429)  
  at t.eefar [as factory] (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:65:89839)  
  at ci (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:65:17094)  
  at j1 (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:59208)  
  at FE (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:54685)  
  at ci (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:17094)  
  at t.eefar [as factory] (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:13269)  
  at ci (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:13644)  
  at template (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:65:11761)  
  at LE (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:54438)  
  at OH (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:17094)  
  at t.eefar [as factory] (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/3ec10b81-d66d-40b6-8222-b44730e21cd5/chunk-ZVLQYCH.mjs:8:92558)  
  at ci (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/f03b1665-93da-4b73-86ab-a32677f87917/chunk-ZVLQYCH.mjs:68:93160)  
  at t.eefar [as factory] (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/f03b1665-93da-4b73-86ab-a32677f87917/chunk-ZVLQYCH.mjs:68:17094)  
  at ci (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/f03b1665-93da-4b73-86ab-a32677f87917/chunk-ZVLQYCH.mjs:8:76108)  
  at rs.create (file:///C:/Users/899440/git/MovieApp_Usecase/movie-angular/.angular/prerender-root/f03b1665-93da-4b73-86ab-a32677f87917/chunk-ZVLQYCH.mjs:8:74719)
```

Create s3 bucket and upload files from dist/browser folder

The screenshot shows the AWS S3 console with the bucket 'radhika-movie-angular' selected. The 'Objects' tab is active, displaying 15 items. The objects listed are:

Name	Type	Last modified	Size	Storage class
about/	Folder	-	-	-
assets/	Folder	-	-	-
dashboard/	Folder	-	-	-
favicon.ico	ico	February 18, 2024, 18:20:15 (UTC+05:30)	14.7 KB	Standard
home/	Folder	-	-	-
index.html	html	February 18, 2024, 18:20:15 (UTC+05:30)	45.6 KB	Standard
login/	Folder	-	-	-

In permissions tab edit bucket policy

The screenshot shows the AWS S3 console with the bucket 'radhika-movie-angular' selected. The 'Permissions' tab is active. Under 'Block public access', there is a note about granting access through access control lists (ACLs), bucket policies, access point policies, or all. A 'Block all public access' button is present. Below it, a note says 'Individual Block Public Access settings for this bucket'. In the 'Bucket policy' section, there is a JSON editor with the following policy:

```
{ "Version": "2012-10-17", "Statement": [ { "Sid": "String1708216889112", "Effect": "Allow", "Principal": "*", "Action": "S3:GetObject", "Resource": "arn:aws:s3:::radhika-movie-angular/*" } ] }
```

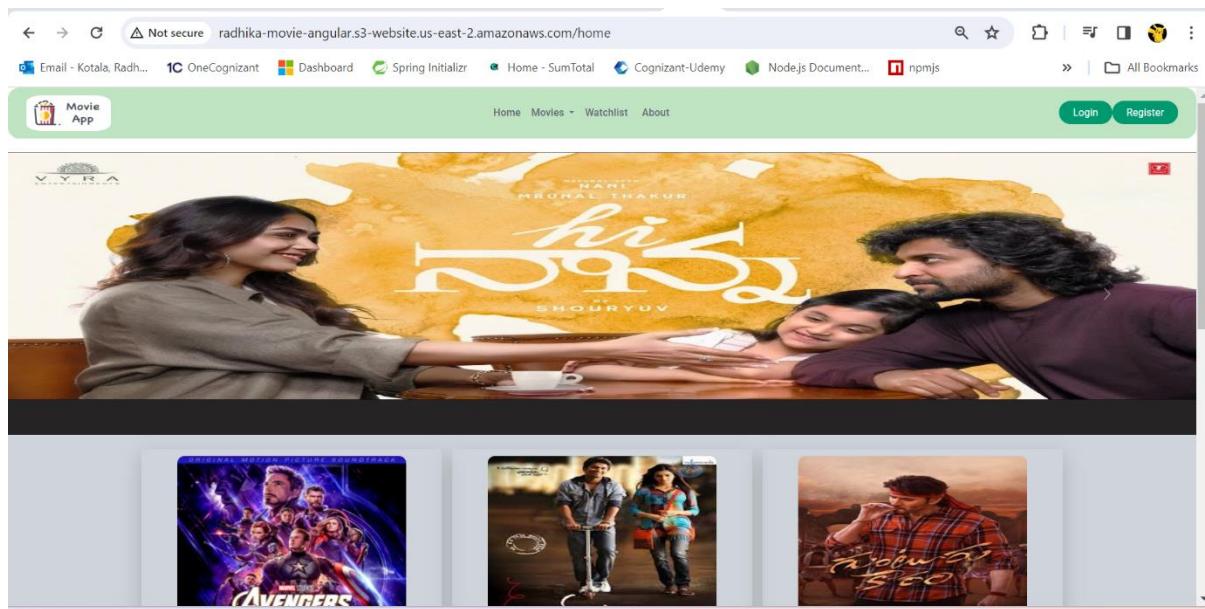
In properties tab enable static website hosting

The screenshot shows the 'Edit static website hosting' configuration page for an S3 bucket named 'radhika-movie-angular'. The 'Static website hosting' section is enabled, and the 'Host a static website' option is selected. The 'Index document' field contains 'index.html'. A note about making content publicly readable is visible. The 'Redirection rules' section is currently empty.

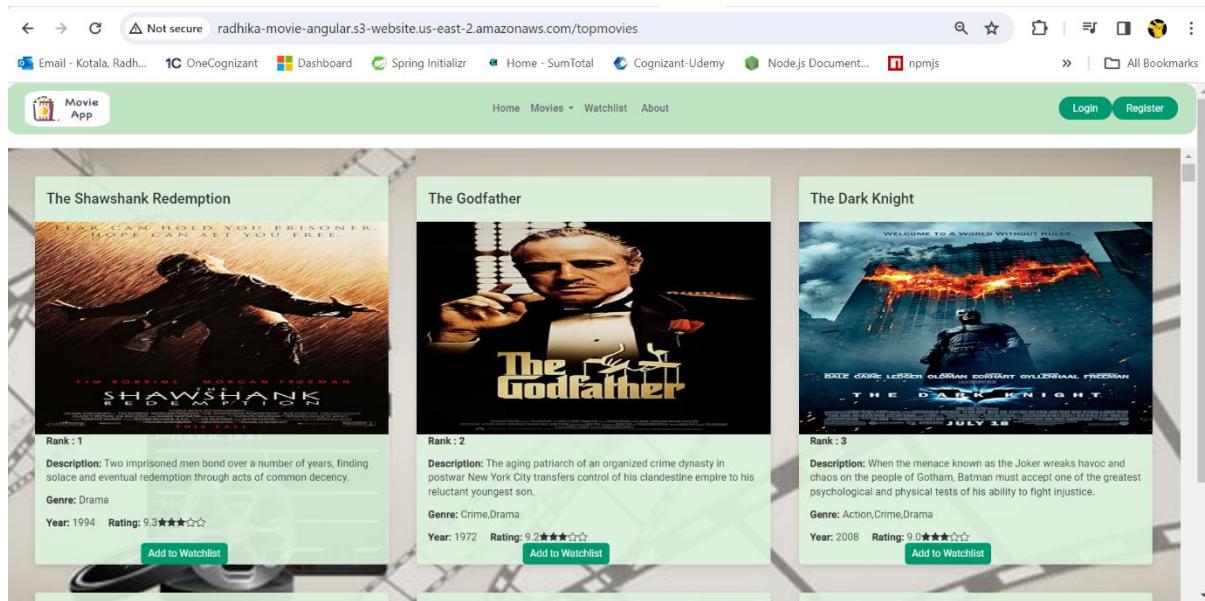
Open static website hosting url

The screenshot shows the 'Static website hosting' section of the bucket properties page. It displays the endpoint URL 'http://radhika-movie-angular.s3-website.us-east-2.amazonaws.com', which is highlighted with a red box.

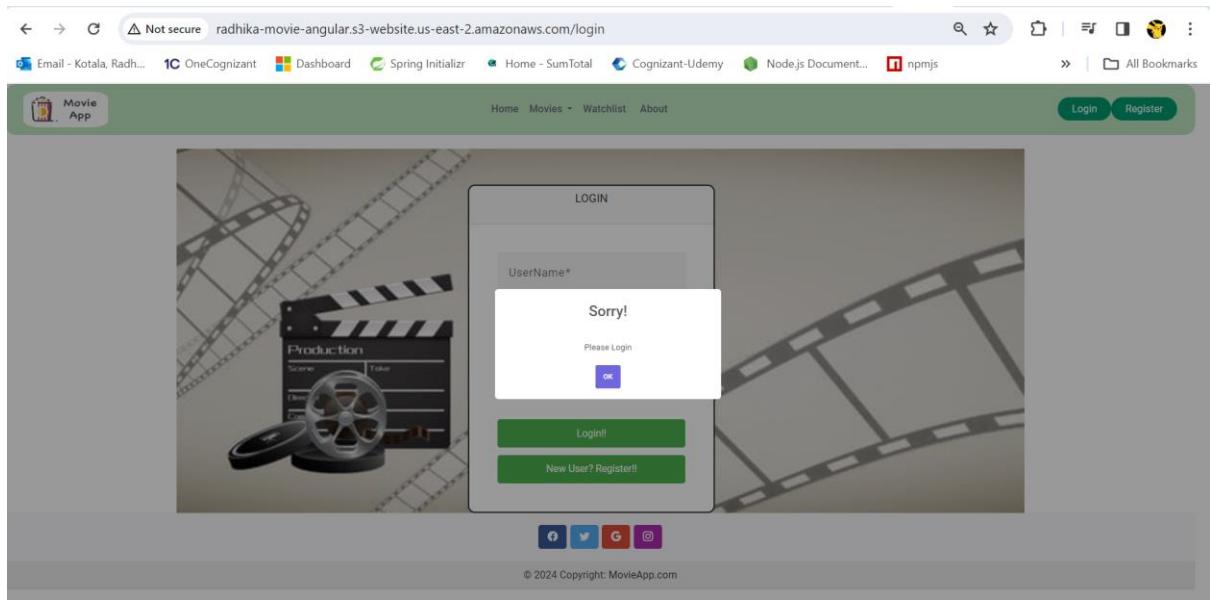
Home page of MovieAPP



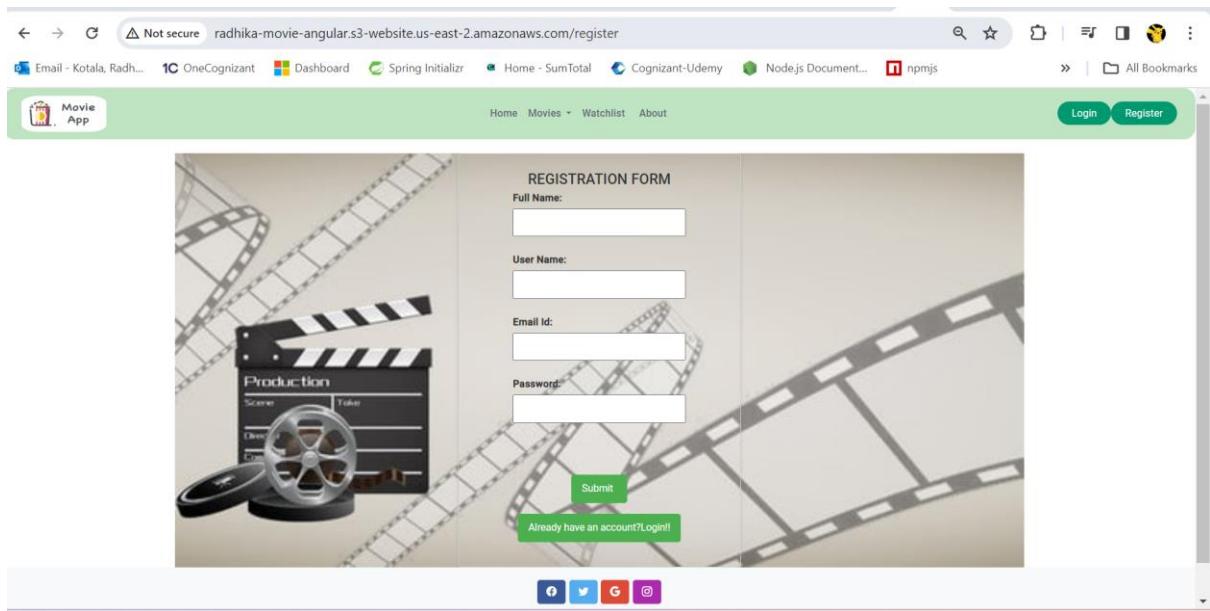
User can watch top100 movie details without login



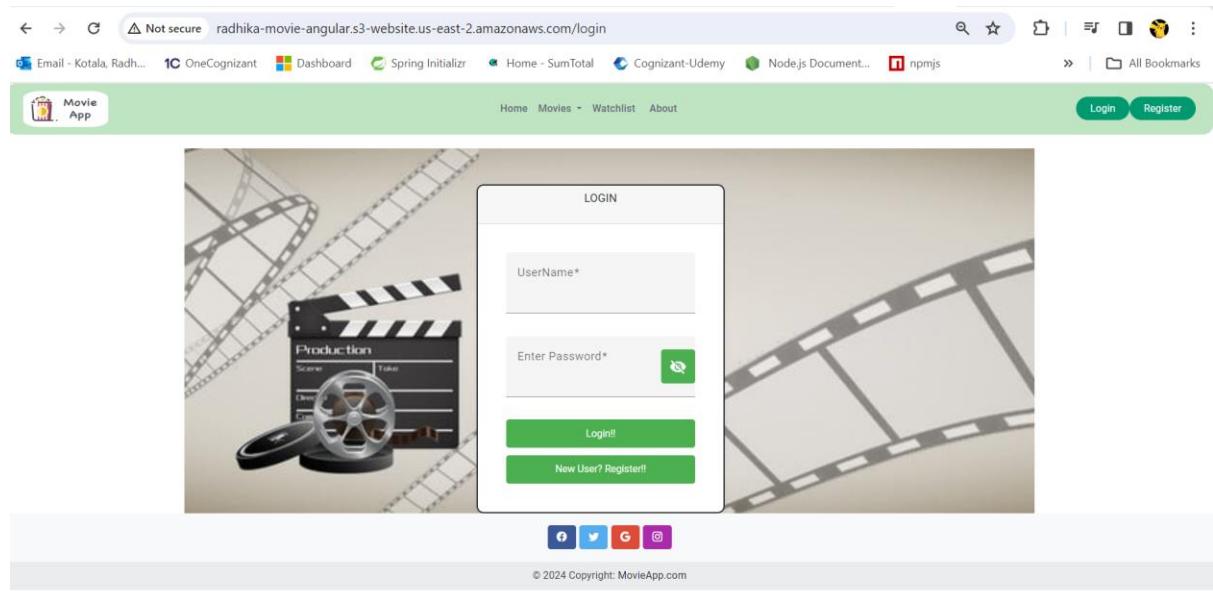
But user cannot add movies to wishlist before login and it redirects to login page



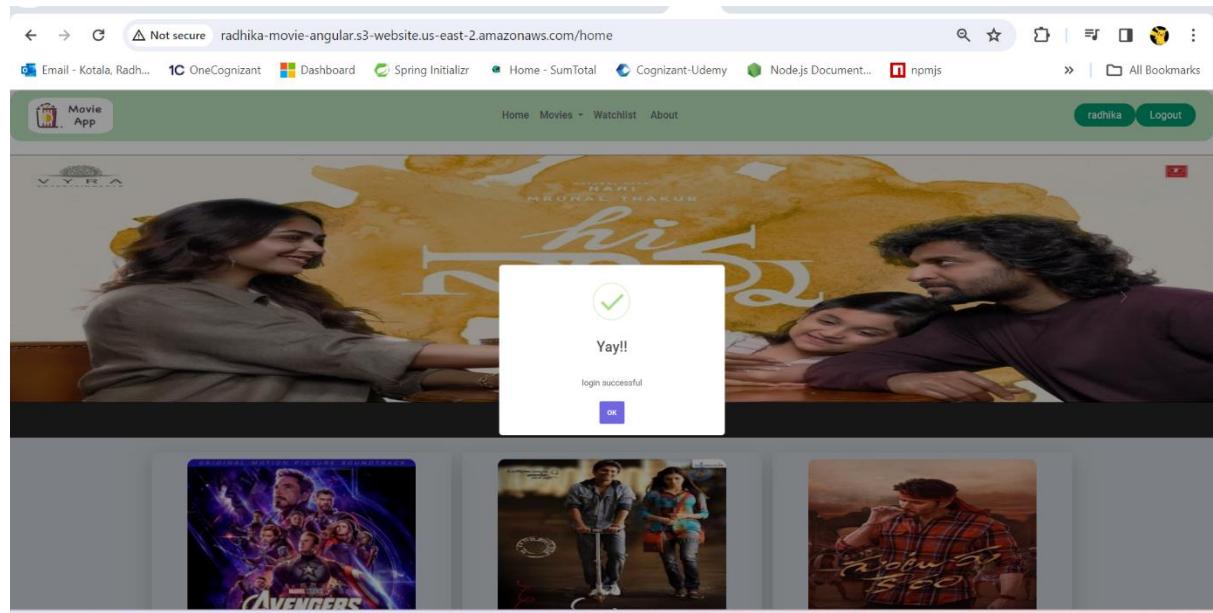
User has to register to movie app



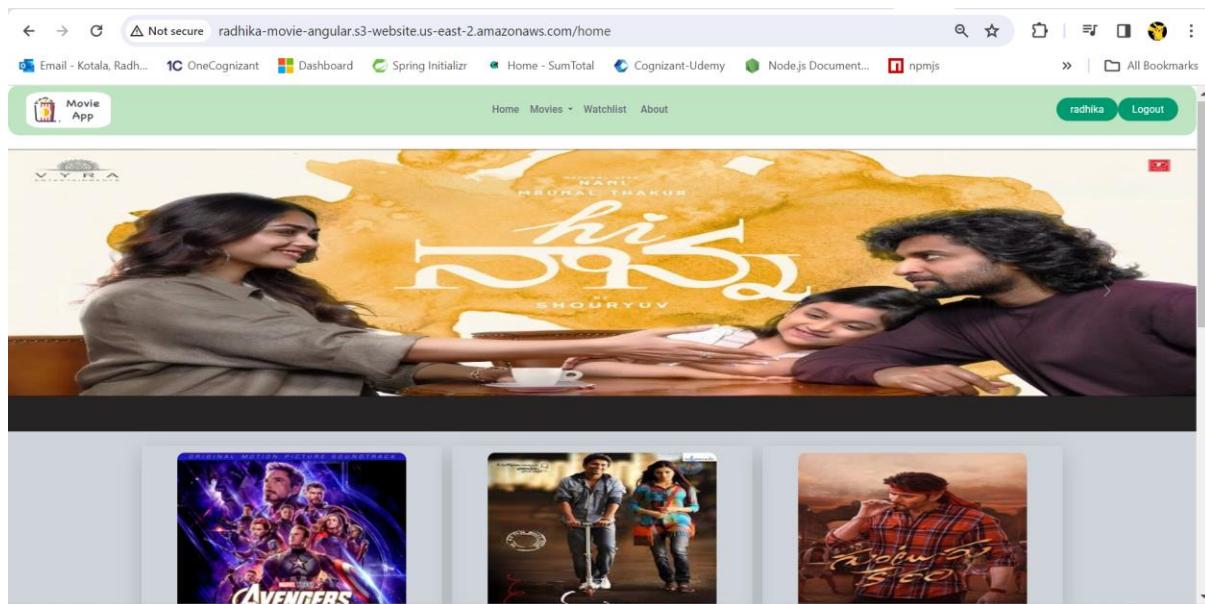
After registering user it redirects to login page



Once user logged in successfully it redirects to home page

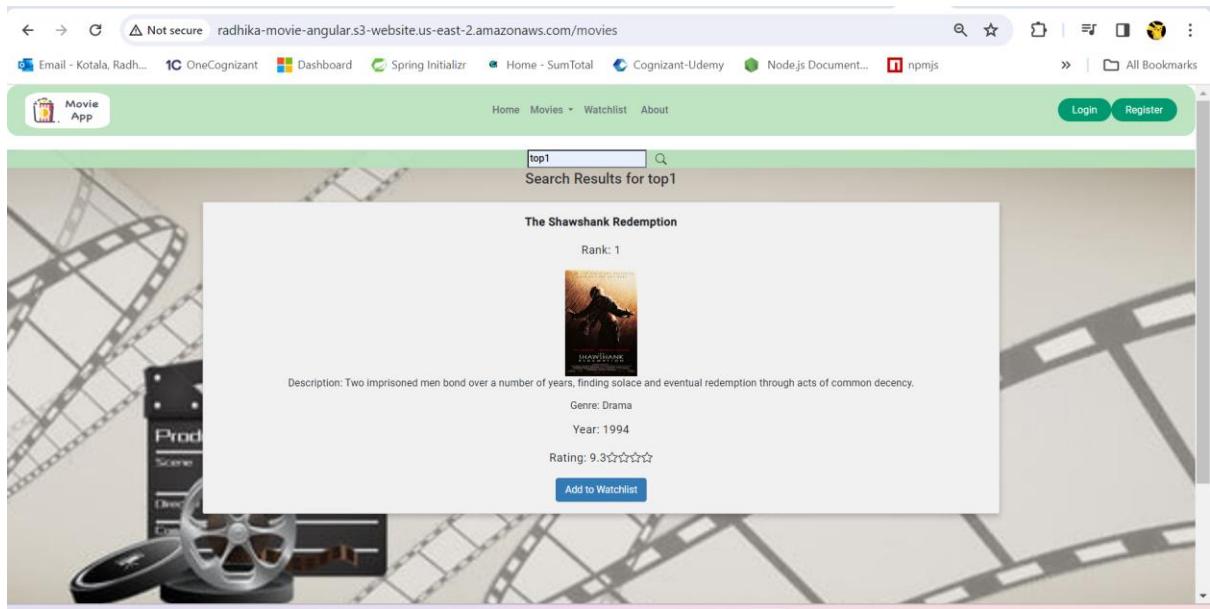


Home page after user logged in and logged in username displays

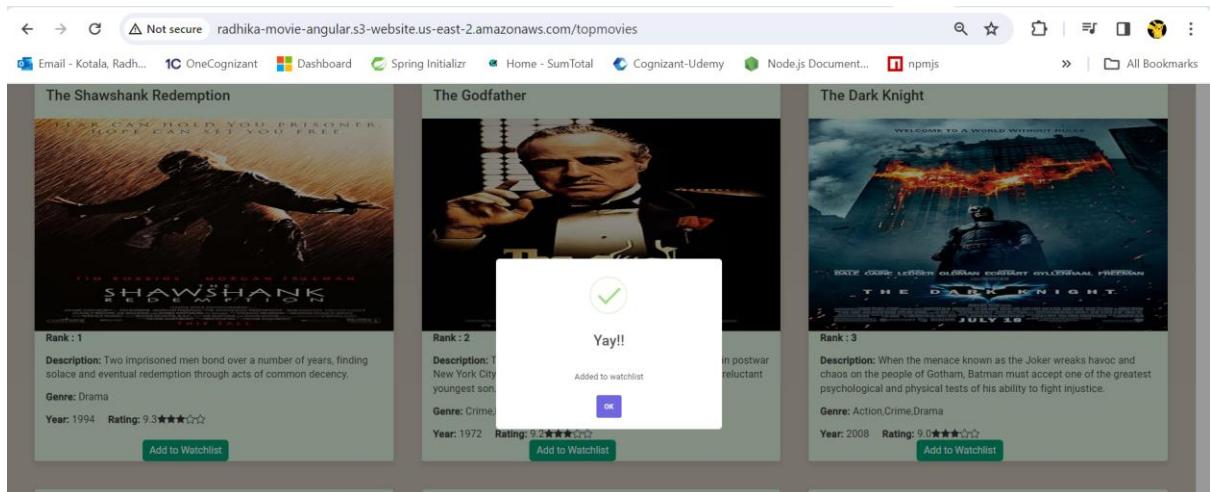


User can search with movie rank

Example – top1, top5, top11



After adding movie to watchlist it



User cannot added movie which is already in watchlist

← → ⌂ Not secure radhika-movie-angular.s3-website.us-east-2.amazonaws.com/topmovies

Email - Kotala, Radh... 1C OneCognizant Dashboard Spring Initializr Home - SumTotal Cognizant-Udemy Node.js Document... npmjs All Bookmarks

The Shawshank Redemption

Rank : 1

Description: Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Genre: Drama

Year: 1994 Rating: 9.3★★★★☆

[Add to Watchlist](#)

The Godfather

Rank : 2

Description: The aging patriarch of an organized crime dynasty in postwar New York City transfers control of his clandestine empire to his reluctant youngest son.

Genre: Crime,Drama

Year: 1972 Rating: 9.2★★★★☆

[Add to Watchlist](#)

The Dark Knight

Rank : 3

Description: When the menace known as the Joker wreaks havoc and chaos on the people of Gotham, Batman must accept one of the greatest psychological and physical tests of his ability to fight injustice.

Genre: Action,Crime,Drama

Year: 2008 Rating: 9.0★★★★☆

[Add to Watchlist](#)

All watchlisted movies

← → ⌂ Not secure radhika-movie-angular.s3-website.us-east-2.amazonaws.com/watchlist

Email - Kotala, Radh... 1C OneCognizant Dashboard Spring Initializr Home - SumTotal Cognizant-Udemy Node.js Document... npmjs All Bookmarks

Movie App

Home Movies Watchlist About radhika Logout

The Godfather

Rank : 1

Description: The aging patriarch of an organized crime dynasty in postwar New York City transfers control of his clandestine empire to his reluctant youngest son.

Genre: Crime,Drama

Year: 1972 Rating: 9.2★★★★☆

[Remove from Watchlist](#)

The Shawshank Redemption

Rank : 2

Description: Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Genre: Drama

Year: 1994 Rating: 9.3★★★★☆

[Remove from Watchlist](#)

12 Angry Men

Rank : 5

Description: The jury in a New York City murder trial is frustrated by a single member whose skeptical caution forces them to more carefully consider the evidence before jumping to a hasty verdict.

Genre: Crime,Drama

Year: 1957 Rating: 9.0★★★★☆

[Remove from Watchlist](#)

User can remove movies from watchlist

← → ⌂ Not secure radhika-movie-angular.s3-website.us-east-2.amazonaws.com/wishlist

Email - Kotala, Radh... OneCognizant Dashboard Spring Initializr Home - SumTotal Cognizant-Udemy Node.js Document... npmjs » All Bookmarks radhika Logout

Movie App

Home Movies Watchlist About

The Godfather

Description: The aging patriarch of an organized crime dynasty in postwar New York City transfers control of his clandestine empire to his reluctant youngest son.

Genre: Crime,Drama

Year: 1972 Rating: 9.2★★★★☆ Remove from Watchlist

The Shawshank Redemption

Description: Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.

Genre: Drama

Year: 1994 Rating: 9.3★★★★☆ Remove from Watchlist

12 Angry Men

Description: The jury in a New York City murder trial is frustrated by a single member whose skeptical caution forces them to more carefully consider the evidence before jumping to a hasty verdict.

Genre: Crime,Drama

Year: 1957 Rating: 9.0★★★★☆ Remove from Watchlist

← → ⌂ Not secure radhika-movie-angular.s3-website.us-east-2.amazonaws.com/wishlist

Email - Kotala, Radh... OneCognizant Dashboard Spring Initializr Home Movies Watchlist About

Movie App

One Flew Over the Cuckoo's Nest

Description: In the Fall of 1963, a Korean War veteran and criminal pleads insanity and is admitted to a mental institution, where he rallies up the scared patients against the tyrannical nurse.

Genre: Drama

Year: 1975 Rating: 8.6★★★★☆ Remove from Watchlist