**CANARY DEPLOYMENT STRATGEY PROJECT DOC**

**# Install AWS CLI**

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"

sudo apt install unzip

unzip awscliv2.zip

sudo ./aws/install

aws configure

-------------------------------------------------------------------------------------

**#Install Terraform**

sudo apt-get update && sudo apt-get install -y gnupg software-properties-common curl

curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg

echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb\_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list

sudo apt-get update && sudo apt-get install terraform -y

terraform -version

-------------------------------------------------------------------------------------------

**# Kubeconfig**

aws eks --region ap-south-1 update-kubeconfig --name devopsshack-cluster

---------------------------------------------------------------------------------

**# Kubectl**

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256"

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

kubectl version --client

------------------------------------------------------------------------------------------

**Install EKSCTL**

curl -sLO "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl\_$(uname -s)\_amd64.tar.gz"

tar -xzf eksctl\_$(uname -s)\_amd64.tar.gz

sudo mv eksctl /usr/local/bin

eksctl version

-------------------------------------------------------------------------------------------

eksctl utils associate-iam-oidc-provider --region ap-south-1 --cluster devopsshack-cluster --approve

-----------------------------------------------------------------------------------------------

eksctl create iamserviceaccount \

--region ap-south-1 \

--name ebs-csi-controller-sa \

--namespace kube-system \

--cluster devopsshack-cluster \

--attach-policy-arn arn:aws:iam::aws:policy/service-role/AmazonEBSCSIDriverPolicy \

--approve \

--override-existing-serviceaccounts

-----------------------------------------------------------------------------------------------

**Install ebs-csi-driver, ingress nginx, cert-manager**

kubectl apply -k "github.com/kubernetes-sigs/aws-ebs-csi-driver/deploy/kubernetes/overlays/stable/ecr/?ref=release-1.11"

kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/main/deploy/static/provider/cloud/deploy.yaml

kubectl apply -f <https://github.com/cert-manager/cert-manager/releases/download/v1.12.0/cert-manager.yaml>

**Install Argo Rollouts in your EKS cluster**

kubectl create namespace argo-rollouts

kubectl apply -n argo-rollouts -f <https://github.com/argoproj/argo-rollouts/releases/latest/download/install.yaml>

kubectl apply -n argo-rollouts -f <https://github.com/argoproj/argo-rollouts/releases/latest/download/dashboard-install.yaml>

***#edit the svc to see the dashboard***

**Install the Plugin kubectl-argo-rollouts**

This gives you the kubectl argo rollouts command.

curl -sLO https://github.com/argoproj/argo-rollouts/releases/latest/download/kubectl-argo-rollouts-linux-amd64

chmod +x kubectl-argo-rollouts-linux-amd64

sudo mv kubectl-argo-rollouts-linux-amd64 /usr/local/bin/kubectl-argo-rollouts

**V1- Project**🡪 <https://github.com/jaiswaladi246/Boardgame.git>

**V2-Project**🡪 IN the above Project got to src/main/resources/templates & remove the index.html file

Create new index.html with below content

<!DOCTYPE html>

<html lang="en" xmlns:th="http://www.thymeleaf.org"

xmlns:sec="https://www.thymeleaf.org/thymeleaf-extras-springsecurity5" data-bs-theme="light">

<head th:replace="fragments/header.html :: html-head(title='Boardgame Home')"></head>

<style>

.canary-banner {

background: linear-gradient(90deg, #00c9ff 0%, #92fe9d 100%);

color: #004d40;

font-weight: bold;

text-align: center;

padding: 12px;

font-size: 1.2rem;

box-shadow: 0 2px 5px rgba(0,0,0,0.1);

}

.game-card {

transition: transform 0.3s, box-shadow 0.3s;

border-radius: 10px;

padding: 20px;

background-color: #f8f9fa;

box-shadow: 0 2px 6px rgba(0,0,0,0.1);

}

.game-card:hover {

transform: scale(1.03);

box-shadow: 0 4px 12px rgba(0,0,0,0.2);

}

.badge-new {

font-size: 0.75rem;

background-color: #ffc107;

color: black;

padding: 3px 8px;

border-radius: 10px;

margin-left: 10px;

}

.top-picks {

background: #fff3cd;

border: 1px solid #ffeeba;

border-radius: 10px;

padding: 20px;

}

.v2-footer {

text-align: center;

font-size: 14px;

color: white;

background: rgba(0, 0, 0, 0.7);

padding: 10px;

position: relative;

bottom: 0;

width: 100%;

}

</style>

<body>

<div class="canary-banner">

🚀 Welcome to <strong>BoardGame v2</strong> – Canary Release with New Features!

</div>

<nav th:replace="fragments/links :: nav.navigation-links"></nav>

<main class="container py-4">

<div class="text-center mb-5">

<h1 class="display-5 fw-bold">🎲 Explore the Latest Boardgames</h1>

<p class="lead text-muted">Discover new experiences added in v2</p>

</div>

<div id="error" class="error pt-3 text-center">

<div th:if="${userAddedMsg}">

<p th:text="${userAddedMsg}" class="alert alert-info"></p>

</div>

</div>

<div th:unless="${boardgames.empty}" class="row g-4">

<div class="col-md-4" th:each="boardgame : ${boardgames}">

<div class="game-card h-100 text-center">

<h4 th:text="${boardgame.name}">Game Name</h4>

<span class="badge-new" th:if="${boardgame.name.contains('X')}">NEW</span>

<a th:href="@{|/${boardgame.id}|}" class="btn btn-outline-primary mt-3">View Details</a>

</div>

</div>

</div>

<div th:if="${boardgames.empty}" class="text-center">

<p class="text-muted">Currently there is no boardgame database available.</p>

</div>

<div sec:authorize="!hasRole('ROLE\_USER')" class="text-center mt-4">

<p class="lead">For more services, login <a href="#" th:href="@{/login}">Here</a></p>

<p class="lead">To join the service, <a href="#" th:href="@{/newUser}">Click</a> here</p>

</div>

<div sec:authorize="hasRole('ROLE\_USER') || hasRole('ROLE\_MANAGER')" class="text-center mt-4">

<p class="lead">Click <a href="#" th:href="@{/secured/addBoardGame}">Here</a> to add a boardgame!</p>

<form th:action="@{/logout}" method="post">

<input type="submit" value="Logout" class="btn btn-danger mt-2">

</form>

</div>

<!-- Top Picks Section -->

<section class="top-picks mt-5">

<h3 class="text-center mb-3">🔥 Top Picks This Week</h3>

<div class="d-flex justify-content-around flex-wrap">

<div>🎯 <strong>Codenames</strong></div>

<div>🧠 <strong>Azul</strong></div>

<div>🛡️ <strong>Carcassonne</strong></div>

<div>🌍 <strong>Ticket to Ride</strong></div>

</div>

</section>

</main>

<div th:include="fragments/links :: div.bottom-links"></div>

<div th:replace="fragments/footer.html :: page-footer"></div>

<div class="v2-footer">

⚡ You are viewing version 2.0 (Canary)

</div>

</body>

</html>

**Project YAMLs**

rollout.yaml

apiVersion: argoproj.io/v1alpha1

kind: Rollout

metadata:

name: boardgame

namespace: webapps # Optional: remove if using default namespace

spec:

replicas: 4

revisionHistoryLimit: 2

selector:

matchLabels:

app: boardgame

template:

metadata:

labels:

app: boardgame

spec:

containers:

- name: boardgame

image: adijaiswal/boardgame:v1

ports:

- containerPort: 8080

strategy:

canary:

maxSurge: 1

maxUnavailable: 0

steps:

- setWeight: 20

- pause: { duration: 1m }

- setWeight: 50

- pause: { duration: 1m }

- setWeight: 100

svc.yaml

apiVersion: v1

kind: Service

metadata:

name: boardgame-svc

spec:

selector:

app: boardgame

ports:

- port: 80

targetPort: 8080

type: LoadBalancer

**Start Canary Rollout**

kubectl-argo-rollouts set image boardgame boardgame=adijaiswal/boardgame:v2 -n webapps