# Containerization using Docker

## Project

**Problem Statement**

You are working with a web development agency that highly relies on Drupal as their base

framework for developing web applications for their clients. So far, you have been deploying

Drupal manually across all the servers but now the firm wants to have the process streamlined and automated.

**Objectives:**

• Download your company’s website files from the given link

• Write a docker file that will make your company’s website work out of the box with a web server (Tip -You can use httpd / apache image and build on top of it)

• Make sure that you use volumes to store the actual data of the website outside of the container

• Push the docker image to your docker hub account so that it can be pulled later

• Create a swarm cluster

• Deploy your firm’s website on the swarm cluster and expose port 8083 to access the website.

Also, ensure that the volumes are configured properly so that the source of the

files are the same for all the containers of the service

Application Link: https://github.com/edurekacontent/dockerContent

**Answer: Containerization using Docker (Drupal Website Deployment)**

> Platform: AWS EC2 (Ubuntu)

> Base Image: httpd:latest

> Website Folder: case-study-app

> Docker Hub Username: henryvijayraj2468

> Exposed Port: 8083

> Cluster Mode: Docker Swarm (with 3 replicas)

> Data: Shared Docker Volume for persistence

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

STEP 1: Launch EC2 & Connect

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

1. Launch an Ubuntu instance

2. t2.medium recommended.

3. Security Group: Jenkins(All-TCP, http, https, ssh, 80, 8083))

4. ppk key

5. Conf Storage: 25, GP3

In Putty,

==========

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

STEP 2: Install Docker

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

sudo apt update &&

sudo apt install -y docker.io &&

sudo systemctl start docker &&

sudo systemctl enable docker &&

sudo usermod -aG docker $USER &&

newgrp docker

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

STEP 3: Clone the Project Repo

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

git clone https://github.com/edurekacontent/dockerContent &&

cd dockerContent

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

STEP 4: Rename Website Folder

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

mv "Case-study app" case-study-app

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

STEP 5: Create Dockerfile

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

vi Dockerfile

paste this:

-----------

FROM httpd:latest

COPY case-study-app/ /usr/local/apache2/htdocs/

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

STEP 6: Login to Docker Hub

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

step 1: login docker hub and copy the username

step 2: generate personal access token(for login password

(((eg: mytoken11

Read & Write

docker login -u henryvijayraj

dckr\_pat\_FAlQpudQAgPDvmH4fxxxx\_-xxxx)))

docker login -u <docker-username-paste here>

password: <docker-personal-access-token>

(login successes)

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

**STEP 7: Build Docker Image**

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

docker build -t <docker-username-paste here>/drupal-site .

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

**STEP 8: Push Image to Docker Hub**

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

docker push <docker-username-paste here>/drupal-site

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

**STEP 9: Initialize Docker Swarm**

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

docker swarm init

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

**STEP 10: Create Docker Volume**

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

docker volume create shared\_web\_data

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

**STEP 11: Deploy Docker Service**

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

docker service create \

--name web-service \

--publish 8083:80 \

--replicas 3 \

--mount type=volume,source=shared\_web\_data,target=/usr/local/apache2/htdocs \

<docker-username-paste here>/drupal-site

**Check status:**

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

docker service ls

**Check containers:**

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

docker ps

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

**STEP 12: Access Website**

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

Open in browser:

http://<your-ec2-public-ip>:8083

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

**You’ve Successfully Deployed Your Website Using:**

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

- Docker

- Dockerfile

- Docker Hub

- Docker Volume

- Docker Swarm (Multi-replica)

- Apache HTTP Server (httpd)