Group 2's Documentation
Sai, Maisha, Xavier, Andrew, Bishajit, Mohammed

Useful Links
https://hub.docker.com/r/jenkins/jenkins
https://dev.to/kamalhossain/how-to-run-docker-containers-in-aws-ec2-3bh0

Process
Created a EC2 with a ubuntu ami on aws.
Ssh into public ec2
sudo apt-get update
Sudo apt-get upgrade

https://dev.to/kamalhossain/how-to-run-docker-containers-in-aws-ec2-3bh0
Install docker onto public ec2:
sudo apt-get install \
  apt-transport-https \
  ca-certificates \
  curl \
  gnupg-agent \
  Software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
apt-cache madison docker-ce
sudo apt install docker.io
sudo apt install docker-compose
sudo usermod -a -G docker ubuntu (so you don't have to use sudo afterwards)
Exit and ssh back in

Create jenkinsfile and dockerfile on a github repo
Paste the code from the pdf into the jenkinsfile

```
pipeline {
agent { label "your label"}
environment{
DOCKERHUB_CREDENTIALS = credentials('your
username-dockerhub')
}
stages {
stage ('Build') {
steps {
```

```
sh ‘’'docker build
[docker-username]/[enter-new-name-here]-demo .’’’
}
}
stage ('Login') {
steps {
sh ‘’'echo $DOCKERHUB_CREDENTIALS_PSW | docker login -u
$DOCKERHUB_CREDENTIALS_USR --password-stdin’’’
}}
stage ('Push') {
steps {
sh ‘’'docker push
[docker-username]/[enter-new-name-here]-demo:latest’’’
}
}
}
}

docker pull jenkins/jenkins
```

Create ECS Cluster

```
Create ECR
Sudo apt install awscli
Aws configure
Push up the image to the repository
```

Copy the URI from the image that was pushed up
Create a Task definition
Paste in the URI and fill in the required/necessary fields (Port mapping to 8080)
Run the task in your cluster
Edit security group to allow port 8080
Get jenkins password and install suggested plugins
Install docker pipeline plugin and aws ec2 plugin

Create a Jenkinsfile and Dockerfile on your github

Paste the following into your dockerfile...

```
FROM openjdk:11

COPY demo-0.0.1-SNAPSHOT.jar .

EXPOSE 8080

CMD java -jar demo-0.0.1-SNAPSHOT.jar
```

Drop the jar file into your repository
Generate token from dockerhub
Follow the steps provided by dockerhub when you generate your token..
Add the token as a password to your jenkins global credentials
Create a multi branch pipeline on your jenkins
Edit the Jenkins file because the old file will not work because it is not telling Docker to do anything

This is my edited jenkins file

29 lines (24 sloc) | 705 Bytes

```
pipeline {
    agent { label "agent1"
          }

environment {
    DOCKERHUB_CREDENTIALS = credentials("bishajit-dockerhub")
}
    stages {
          stage ('Build') {
              steps {
                  sh 'docker build -t bishajit/jenkins .'
                }
              }

             stage ('Login') {
                   steps {
                     sh 'echo $DOCKERHUB_CREDENTIALS_PSW | docker login -u $DOCKERHUB_CREDENTIALS_USR --password-stdin'
                    }
                  }

              stage ('Push') {
                   steps {
                     sh 'docker push bishajit/jenkins:latest'
          }
        }
    }
}


```

Update your Jenkins plugins!!!
Build the pipeline and it should be successful