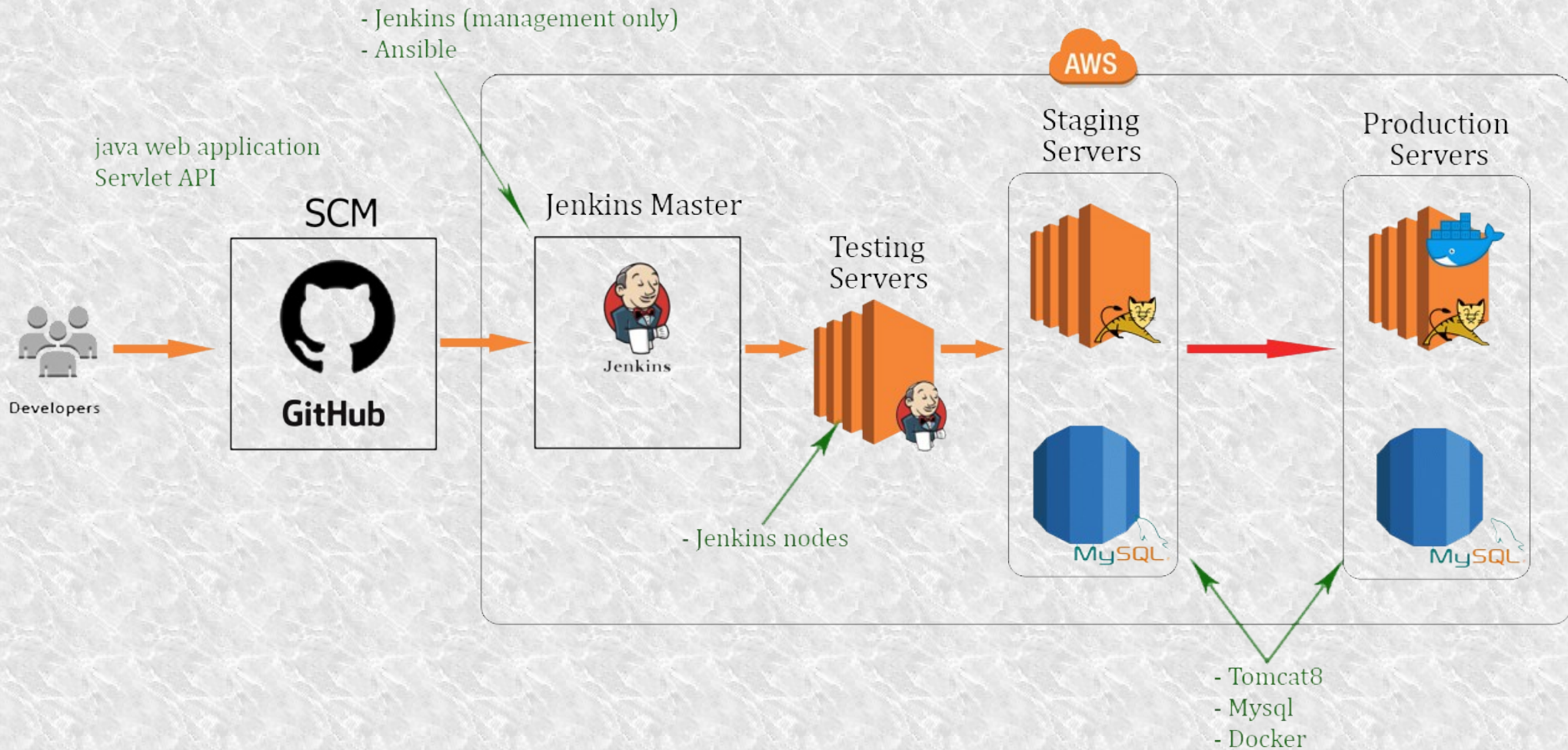


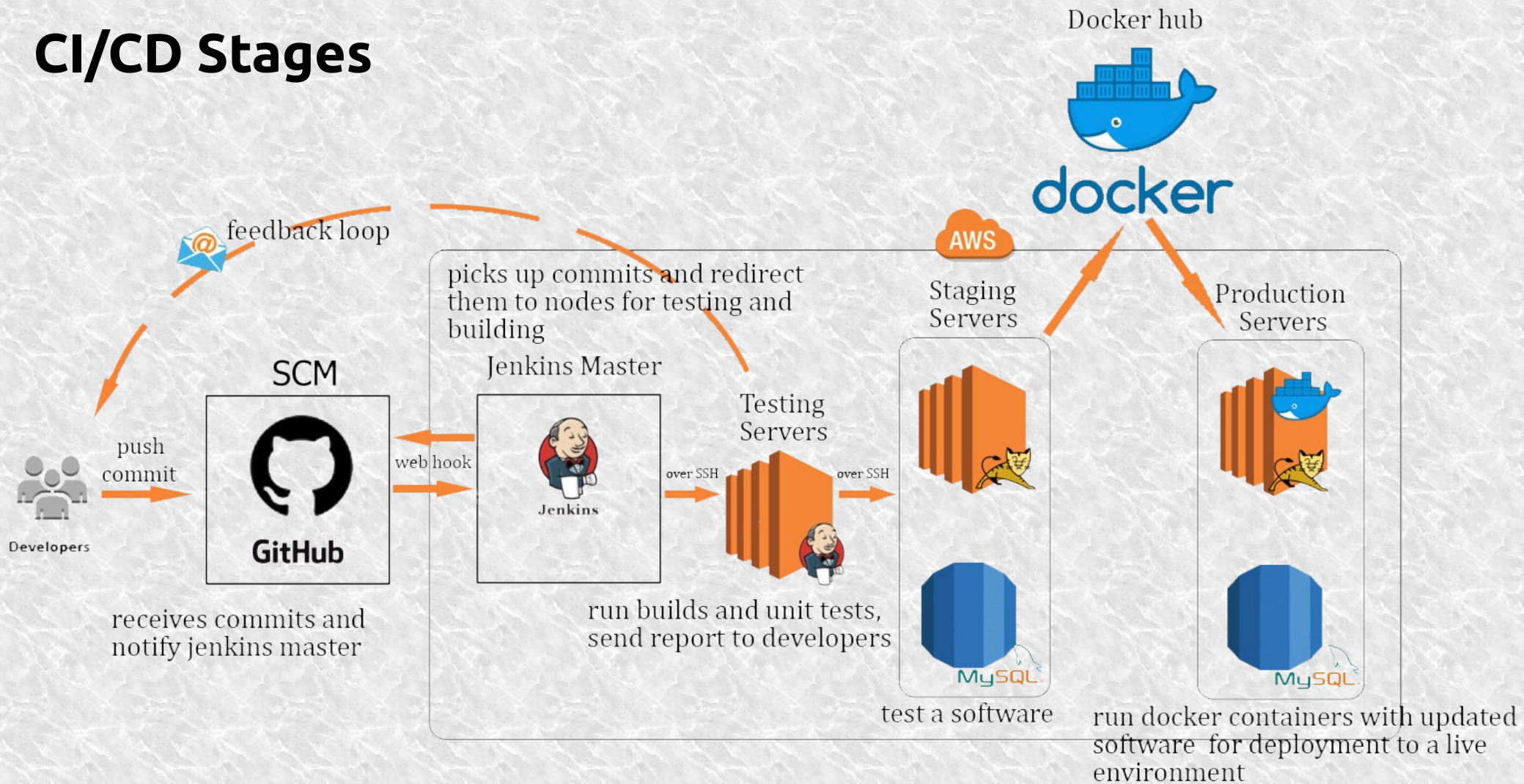
Project Task

1. Create book store management web application
2. Apply MySQL database to accessing and managing data
3. Focus on performance optimization

CI/CD Pipeline



CI/CD Stages



Servers Configuration with Ansible Playbooks

1. Install Ansible on Jenkins Master server
2. Run Ansible Playbook for Jenkins Master to configure required software on it
3. Setup required software on all servers by running Ansible Playbooks



```
--
- name: Install Jenkins on localhost
  hosts: localhost
  connection: local
  become: yes

  tasks:
    - name: Install Jenkins apt repository key
      apt_key: url=https://pkg.jenkins.io/debian-stable/jenkins.io.key state=present

    - name: Configure the Jenkins repository on localhost
      apt_repository: repo='deb https://pkg.jenkins.io/debian-stable binary/' state=present

    - name: Install Java (openjdk-8-jdk)
      apt: name=openjdk-8-jdk state=latest

    - name: Install Jenkins
      apt: name=jenkins state=latest update_cache=yes

    - name: Run Jenkins' service and put it to autorn
      service: name=jenkins state=started enabled=yes

    - name: check if initialAdminPassword file exists
      stat: path=/var/lib/jenkins/secrets/initialAdminPassword
      register: initialAdminPassword_status
```

```
ubuntu@inst1:~/ansible/playbooks$ cat playbook_install_docker.yml
--
- name: Install Docker engine
  hosts: TEST_SERVERS,STAGING_SERVERS,PROD_SERVERS
  become: yes

  tasks:
    # ===== for Ubuntu only =====
    - name: Install transport-https for Ubuntu
      apt: name=apt-transport-https state=latest

    - name: Install Docker apt repository key
      apt_key: url=https://download.docker.com/linux/ubuntu/gpg state=present

    - name: Configure the Docker repository on localhost
      apt_repository: repo='deb https://download.docker.com/linux/ubuntu bionic stable' state=present

    - name: Update apt and install docker-ce
      apt: update_cache=yes name=docker-ce state=latest

    - name: Add user to docker group
      user:
        name: 'ubuntu'
        groups: docker
        append: yes

    - name: Unconditionally reboot the machine with all defaults
      reboot:
```

```
ubuntu@inst1:~/ansible/playbooks$ cat playbook_install_tomcat.yml
--
- name: Install Tomcat server
  hosts: STAGING_SERVERS,PROD_SERVERS
  become: yes

  tasks:
    - name: Check LINUX Version
      debug: var=ansible_os_family

    - block:
        # ===== Block for Ubuntu =====
        - name: Install Java for Ubuntu
          apt: name=default-jre state=latest

        - name: Install Tomcat for ubuntu
          apt: name=tomcat8 state=latest

        - name: Start Tomcat service for ubuntu
          service: name=tomcat8 state=started enabled=yes

        when: ansible_os_family == "Debian"

    - block:
        # ===== Block for RedHat =====
        - name: Install tomcat for RedHat
          yum: name=tomcat state=latest

        - name: Start tomcat service for RedHat
          service: name=tomcat state=started enabled=yes

        when: ansible_os_family == "RedHat"
```

Jenkins Configuration



Jenkins

1. Setup plugins for Jenkins
 2. Add required credentials for GitHub, Nodes etc.
 3. Create and configure the pipeline
- Builds run on jenkins nodes only !

Credentials				
T	P	Store	Domain	ID
		Jenkins	(global)	a876f452-8cd3-46c1-aec6-efe222f5168a
		Jenkins	(global)	inst2_private_key
		Jenkins	(global)	dockerhub
		Jenkins	(global)	Prod1

Icon: [S](#) [M](#) [L](#)

Publish over SSH	
Jenkins SSH Key	
Passphrase
Path to key	
Key	-----BEGIN RSA PRIVATE KEY----- MIIEogIBAAKCAQEajzMPsRnCLjP7LOvmcy ID39PgZMG7DBER/j5nES+TYQuH1gBlwwP BJs1uJUQreOAAjgH63aYuo09fagPTwtjUH/8 GNTzArc5MpaNcPC7BichqXUVjIDi3Gityb4C 9kYnPX5e36WcX1N1DX/oxlVZPIxtJWRzTf

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap S
	master	Linux (amd64)	In sync	4.39 GB	
	Prod1	Linux (amd64)	In sync	4.67 GB	
	Srv2	Linux (amd64)	In sync	4.24 GB	
Data obtained		28 min	28 min	28 min	

Building And Unit Testing

- Unit testing - JUnit
- Maven builds war archive
- Feedback Loops

The Maven logo, featuring the word "Maven" in a black, sans-serif font with a small trademark symbol. The letter 'a' is stylized with a colorful, multi-colored underline.

Testing Environment

- Publishing release on Staging servers
- Tests
- Sending Docker image with new version of release to Docker Hub

The JUnit logo, featuring the letters "J" and "U" in green and "nit" in red, all in a bold, sans-serif font.

Deploying To Production

- Run Docker container with the desired version of software on production servers



Jobs Implementation

All +					
S	W	Name ↓	Last Success	Last Failure	Last Duration
		BookStore	19 hr - #15	N/A	20 sec

☒ Restrict where this project can be run

Label Expression:

Label TEST_SERVERS is serviced by 1 node. Permissions or other restrictions provided by plugins may prevent this job from running on those nodes.

Advanced...

Source Code Management

☐ None

☒ Git

Repositories

Repository URL:

Credentials: Add

Advanced...

Build Triggers

- ☐ Trigger builds remotely (e.g., using scripts)
- ☐ Build after other projects are built
- ☐ Build periodically
- ☒ GitHub hook trigger for GITScm polling

Build

Invoke top-level Maven targets

Maven Version:

Goals:

POM:

Properties:

Post-build Actions

Editable Email Notification

Disable Extended Email Publisher

Failure - Any

Send To:

Add

Advanced...

Success

Send To:

Add

Advanced...

Send build artifacts over SSH

SSH Publishers

SSH Server

Name:

Advanced...

Transfers

Transfer Set

Source files:

Remove prefix:

Remote directory:

Exec command:

All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)

Advanced...

Build other projects

Projects to build:

- ☒ Trigger only if build is stable
- ☐ Trigger even if the build is unstable

Console Output

```
Started by user Student S T
Running as SYSTEM
Building remotely on Srv2 (TEST_SERVERS) in workspace /home/ubuntu/jenkins_slave/workspace/BookStore
using credential a876f452-8cd3-46c1-aec6-efe222f5168a
> git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url git@github.com:serjr777/applet_deploy.git # timeout=10
Fetching upstream changes from git@github.com:serjr777/applet_deploy.git
> git --version # timeout=10
using GIT_SSH to set credentials
> git fetch --tags --progress -- git@github.com:serjr777/applet_deploy.git +refs/heads/*:refs/remotes/origin/* #
timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 6d87ae67bb36eef36f45014c043f51abe2db3293 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f 6d87ae67bb36eef36f45014c043f51abe2db3293 # timeout=10
Commit message: "Merge branch 'master' of github.com:serjr777/applet_deploy"
> git rev-list --no-walk 6d87ae67bb36eef36f45014c043f51abe2db3293 # timeout=10
No emails were triggered.
[BookStore] $ /home/ubuntu/jenkins_slave/tools/hudson.tasks.Maven_MavenInstallation/Maven/bin/mvn -f /home/ubuntu
/jenkins_slave/workspace/BookStore/pom.xml clean install
[INFO] Scanning for projects...
[INFO]
[INFO] -----< Bookstore:Bookstore >-----
[INFO] Building App Maven Webapp 1.0-SNAPSHOT
```

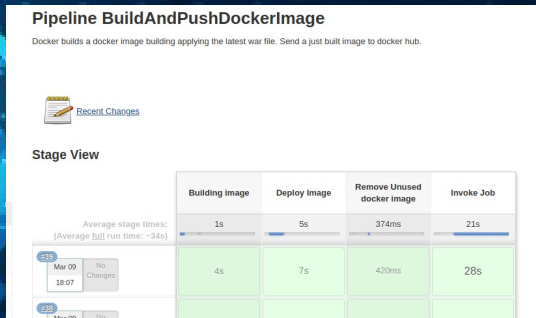
```
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 6.070 s
[INFO] Finished at: 2020-03-09T16:07:12Z
[INFO] -----
```

```
SSH: Connecting from host [inst2]
SSH: Connecting with configuration [Inst3] ...
SSH: EXEC: STDOUT/STDERR from command [sudo systemctl restart tomcat8] ...
SSH: EXEC: completed after 5,209 ms
SSH: Disconnecting configuration [Inst3] ...
SSH: Transferred 1 file(s)
Email was triggered for: Success
Sending email for trigger: Success
Sending email to: serjr777@gmail.com
Triggering a new build of BuildAndPushDockerImage
Finished: SUCCESS
```

```
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running ua.bookstore.BookTest
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.111 s - in ua.bookstore.BookTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- maven-war-plugin:3.2.2:war (default-war) @ Bookstore ---
[INFO] Packaging webapp
[INFO] Assembling webapp [Bookstore] in [/home/ubuntu/jenkins_slave/workspace/BookStore/target/bookstore]
[INFO] Processing war project
```

Jobs Implementation

Jobs Implementation



General Build Triggers Advanced Project Options Pipeline

Description Node builds a docker image applying the latest war file. Sends the just built image to docker hub. Invokes parameterized job

General

```
pipeline {
  environment {
    registry = "serj777/bookstore"
    registryCredential = 'dockerhub'
    dockerImage = ''
  }
  agent { label 'TEST_SERVERS' }
  stages {
    stage('Building image') {
      steps {
        script {
          dockerImage = docker.build(registry + ":%BUILD_NUMBER", "/home/ubuntu/jenkins_slave/workspace")
        }
      }
    }

    stage('Deploy Image') {
      steps {
        script {
          docker.withRegistry( '', registryCredential ) {
            dockerImage.push()
          }
        }
      }
    }

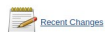
    stage('Remove Unused docker image') {
      steps {
        sh "docker rmi $registry:$BUILD_NUMBER"
      }
    }

    stage('Invoke Job') {
      steps {
        build job: 'DeployToProduction',
              parameters: [
                string(name: 'bs_version', value: String.valueOf("${currentBuild.previousBuild.getNumber()}"))
              ]
      }
    }
  }
}
```

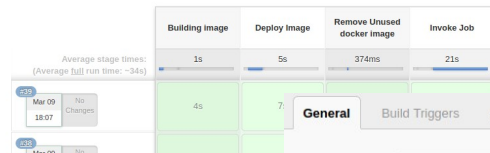
Jobs Implementation

Pipeline BuildAndPushDockerImage

Docker builds a docker image building applying the latest war file. Send a just built image to docker hub.



Stage View



General Build Triggers Advanced Project Options Pipeline

Description Node builds a docker image with latest war file. Sends the just built image to docker hub. Invokes parameterized job

```
GNU nano 2.9.3 Dockerfile
# we are extending everything from tomcat:8.0 image ...
FROM tomcat:8.0

MAINTAINER sergey
# COPY path-to-your-application-war path-to-webapps-in-docker-tomcat
COPY BookStore/target/bookstore.war /usr/local/tomcat/webapps/
```

```
pipeline {
  environment {
    registry = "serjir77/bookstore"
    registryCredential = "dockerhub"
    dockerImage = ""
  }
  agent { label 'TEST_SERVERS' }
  stages {
    stage('Building image') {
      steps {
        script {
          dockerImage = docker.build(registry + ":%BUILD_NUMBER", "/home/ubuntu/jenkins_slave/workspace")
        }
      }
    }
    stage('Deploy Image') {
      steps {
        script {
          docker.withRegistry( '', registryCredential ) {
            dockerImage.push()
          }
        }
      }
    }
    stage('Remove Unused docker image') {
      steps {
        sh "docker rmi $registry:$BUILD_NUMBER"
      }
    }
    stage('Invoke Job') {
      steps {
        build job: 'DeployToProduction',
        parameters: [
          string(name: 'bs_version', value: String.valueOf("${currentBuild.previousBuild.getNumber()}"))
        ]
      }
    }
  }
}
```

Jobs Implementation



DeployToProduction

20 hr - #3623 hr - #2919

Icon: [S](#) [M](#) [L](#)

[Legend](#) [Atom feed for all](#) [Atom feed for failures](#)

☒ This project is parameterized

String Parameter

Name

Default Value

Description

Pipeline

Definition

```
1 pipeline {  
2   agent { label 'PROD_SERVERS' }  
3   stages {  
4     stage('Remove Unused docker image') {  
5       steps(  
6         sh 'sudo /home/ubuntu/deploy.sh ${params.bs_version}'  
7       )  
8     }  
9   }  
10 }
```

```
GNU nano 2.9.3 deploy.sh  
  
#!/bin/bash  
  
#read -p "Enter version of bookstore image in dockerhub:" image  
  
docker login --username serjrr777 --password 7a9274c1-60e8-49e0-89c6-91ba4f19fdd8  
  
docker stop bookstore  
docker rm bookstore  
  
docker image prune -a -f #remove unusable images  
  
docker run -d -it --name=bookstore -p 80:8080 serjrr777/bookstore:$1
```

Books Management

[Add New Book](#) [List All Books](#)

List of Books

ID	Title	Author	Price	Actions
2	The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win	Gene Kim, Kevin Behr , George Spafford	17.0	Edit Delete
3	Effective Java - 3rd Edition	Joshua Bloch	35.0	Edit Delete
4	What Is DevOps?	Mike Loukides	102.0	Edit Delete
5	Building a DevOps Culture	Mandi Walls	15.0	Edit Delete
6	Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale	Ryn Daniels	16.0	Edit Delete
	The DevOps Handbook: How to			

Pros and Cons

```
mirror_mod = modifier_ob.modifiers.new(  
    # Add mirror object to mirror_ob  
    mirror_mod.mirror_object = mirror_ob  
    operation = "MIRROR_X":  
    mirror_mod.use_x = True  
    mirror_mod.use_y = False  
    mirror_mod.use_z = False  
    operation = "MIRROR_Y":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = True  
    mirror_mod.use_z = False  
    operation = "MIRROR_Z":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = False  
    mirror_mod.use_z = True
```

```
Selection at the end -add back the deselected  
mirror_ob.select= 1  
modifier_ob.select=1  
key.context.scene.objects.active = modifier_ob  
print "selected" + str(modifier_ob) # modifier  
mirror_ob.select = 0  
key = key.context.selected_objects[0]  
key.data.objects[key.name].select = 1  
print("please select exactly two objects, we")
```

OPERATOR CLASSES -----

```
class MirrorOperator(bpy.types.Operator):  
    """Add a mirror to the selected object"""  
    bl_idname = "object.mirror_mirror_x"  
    bl_label = "Mirror X"
```

```
    def execute(self, context):  
        context.active_object is not None
```

Questions

The background of the slide is a dark blue field filled with a complex, glowing pattern. It features a network of interconnected lines and nodes, resembling a circuit board or a data flow diagram. Overlaid on this network are numerous binary digits (0s and 1s) in a lighter blue color, creating a digital or technological aesthetic.



Thank you !