

CASE STUDY

Scenario: Streamline Solutions Inc. - Optimizing Software Deployment

Problem Statement: Implementing a Seamless CI/CD Workflow for a Spring Boot Application using Azure Boards, Azure, Git, Maven, Docker, Github, Github Actions

Step 1: Azure VM Configuration

Create a Virtual Machine on Microsoft Azure

The screenshot shows the Microsoft Azure portal interface. At the top, there's a header bar with the URL 'https://portal.azure.com/#home'. Below the header is a blue navigation bar with the text 'Microsoft Azure' and a search bar that says 'Search resources, services, and docs (G+/-)'. On the left side, there's a sidebar with sections for 'Azure services' and 'Resources'. Under 'Azure services', there are icons for 'Create a resource', 'Virtual machines', 'Quickstart Center', 'App Services', 'Storage accounts', 'SQL databases', 'Azure Cosmos DB', 'Kubernetes services', and 'Function App'. Below this is a 'More services' link. Under 'Resources', there's a 'Recent' tab with a single item 'RDBD1' listed as a Resource group, viewed 3 days ago. There's also a 'See all' link. At the bottom of the sidebar, there's a 'Navigate' section with links for 'Subscriptions', 'Resource groups', 'All resources', and 'Dashboard'. The main content area of the portal shows various service icons and navigation links at the bottom.

The screenshot shows the Microsoft Azure home page. On the left, there's a sidebar with 'Azure services' (Create a resource, Virtual machines, More services), 'Resources' (Recent, Favorite, Name: RDBD1, See all), and 'Navigate' (Subscriptions, Resource groups, All resources, Dashboard). The main area is titled 'Virtual machines' with a 'Create' button. It includes a 'Description' section about creating a virtual machine, a 'Free training from Microsoft' section with links to 'Introduction to Azure virtual machines' and 'Create a Windows virtual machine in Azure', and a 'Last Viewed' section for 'Create a Linux virtual machine in Azure'. At the bottom of the main area are 'Useful links' for 'Overview'.

Create a virtual machine with image - Ubuntu Server 22.04

The screenshot shows the 'Create a virtual machine' wizard. Step 1: Set instance details. It asks for a 'Virtual machine name' (JS-server), 'Region' ((US) East US), 'Availability options' (No infrastructure redundancy required), 'Security type' (Trusted launch virtual machines), and 'Image' (Ubuntu Server 22.04 LTS - x64 Gen2). It also shows 'VM architecture' options: Arm64 (radio button) and x64. Navigation buttons at the bottom include 'Review + create', '< Previous', 'Next : Disks >', and 'Give feedback'. The status bar at the bottom shows weather (78°F, Mostly sunny), system icons, and the date/time (9:43 AM, 8/28/2023).

Select Size as B2s and create username and password for your machine

The screenshot shows the 'Create a virtual machine' wizard in the Microsoft Azure portal. The user has selected the 'Standard_B2s - 2 vcpus, 4 GiB memory (₹2,384.94/month)' VM size. Under the 'Administrator account' section, 'Password' is selected as the authentication type, and the username 'xyz' and password '*****' are provided. The 'Inbound port rules' section is collapsed.

The screenshot shows the 'Create a virtual machine' wizard in the Microsoft Azure portal. The user has selected the 'Standard_B2s - 2 vcpus, 4 GiB memory (₹2,384.94/month)' VM size. Under the 'Administrator account' section, 'Password' is selected as the authentication type, and the username 'xyz' and password '*****' are provided. The 'Inbound port rules' section is expanded, showing options for public inbound ports. A note states that all traffic from the internet will be blocked by default. The 'Review + create' button is visible at the bottom.

Click on disks, and select Standard HDD as OS disk type

VM disk encryption
Azure disk storage encryption automatically encrypts your data stored on Azure managed disks (OS and data disks) at rest by default when persisting it to the cloud.

Encryption at host Encryption at host is not registered for the selected subscription.
[Learn more about enabling this feature](#)

OS disk

OS disk type * Standard HDD (locally-redundant storage) The selected VM size supports premium disks. We recommend Premium SSD for high IOPS workloads. Virtual machines with Premium SSD disks qualify for the 99.9% connectivity SLA.

Delete with VM

Key management Platform-managed key

Enable Ultra Disk compatibility Ultra disk is not supported with selected security type.

Data disks for JS-server

Review + create < Previous Next : Networking > Give feedback

78°F Mostly sunny Search

Click on Review and create and then create.

Validation passed

Subscription	RDBD
Resource group	RDBD1
Virtual machine name	JS-server
Region	East US
Availability options	No infrastructure redundancy required
Security type	Trusted launch virtual machines
Enable secure boot	Yes
Enable vTPM	Yes
Integrity monitoring	No
Image	Ubuntu Server 22.04 LTS - Gen2
VM architecture	x64
Size	Standard B2s (2 vcpus, 4 GiB memory)
Authentication type	Password
Username	xyz
Public inbound ports	SSH, HTTP
Azure Spot	No

Create < Previous Next > Download a template for automation Give feedback

78°F Mostly sunny Search

The screenshot shows two windows from the Microsoft Azure portal.

Top Window: This window displays the "CreateVm-canonical.0001-com-ubuntu-server-jammy-2-20230828094259 | Overview" page. It shows a green checkmark indicating the deployment is complete. Deployment details include a name of "CreateVm-canonical.0001-com-u...", a subscription of "RDBD", and a resource group of "RDBD1". The start time was 8/28/2023, 9:48:43 AM. A Correlation ID is also listed. Below this, there are sections for "Deployment details" (Setup auto-shutdown, Monitor VM health, Run a script inside the virtual machine) and "Next steps". Buttons for "Go to resource" and "Create another VM" are present. On the right side, there are promotional links for Cost Management, Microsoft Defender for Cloud, Free Microsoft tutorials, and Work with an expert.

Bottom Window: This window displays the "JS-server - Microsoft Azure" page, specifically the "Virtual machine" view. It shows the VM's status as "Running" in the "East US" location. The operating system is "Linux (ubuntu 22.04)". The public IP address is 172.172.220.114. The virtual network/subnet is "SarvaniMeka-VM-vnet/default". There are tabs for Properties, Monitoring, Capabilities (7), Recommendations, and Tutorials. Under Properties, the virtual machine details are shown: Computer name (JS-server), Operating system (Linux (ubuntu 22.04)), and Image publisher (canonical). Under Networking, the public IP address is 172.172.220.114, and the private IP address is 10.0.0.4. The port 879 is also listed.

**Open ssheasy.com and put public IP address as host to connect.
Also put username and password, and connect.**

JS-server - Microsoft Azure

WebSSH Client

https://ssheasy.com

172.172.220.114 22 Bypass proxy

xyz *****

Private Key Select key file

Connect

Help Us Improve - Please give us feedback

SSH to your machine from anywhere with the SSH client running directly in your browser.

WebSSH is SSH/SFTP client built on golang's SSH and SFTP library, compiled to Web Assembly to allow you to connect to your machine directly from your browser.

No credentials or other plain text information is shared with our servers. Our servers only tunneling the packets from your browser to the machine you connect. They see only the same information that is visible for any router or other computer on the network when you use OpenSSH or other command line client.

Just fill out the connection information and click connect to initiate the SSH session, once the remote hosts answers a port up will show you the fingerprint of the server key and you can decide whether you want to continue with the connection. After successfully connecting you can just type your command into this Xterm.js console or browse/manage your files in the file manager opens below the console.

Source code available on github: <https://github.com/hullarb/ssheasy>

79°F Near record

Find Match Case Find Next Find Previous

JS-server - Microsoft Azure

WebSSH Client

https://ssheasy.com

Connected to xyz@172.172.220.114

```
System information as of Mon Aug 28 04:34:11 UTC 2023
System load: 0.00537109375 Processes: 107
Usage of /: 5.1% of 28.89GB Users logged in: 0
Memory usage: 7% IPv4 address for eth0: 10.0.0.4
Swap usage: 0%
```

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See <https://ubuntu.com/esm> or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Mon Aug 28 04:31:50 2023 from 134.209.166.230
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

xyz@JS-server:~\$

File Manager > home > xyz

.bash_logout... .profile .bashrc .cache .ssh

79°F Near record

Find Match Case Find Next Find Previous

10:03 AM 8/28/2023

10:04 AM 8/28/2023

Connected to xyz@172.172.220.114

```
xyz@JS-server:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://azure.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://azure.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://azure.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://azure.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://azure.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://azure.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [894 kB]
Get:12 http://azure.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [214 kB]
Get:13 http://azure.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [15.6 kB]
Get:14 http://azure.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [714 kB]
Get:15 http://azure.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [114 kB]
Get:16 http://azure.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [973 kB]
Get:17 http://azure.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [212 kB]
Get:18 http://azure.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [21.7 kB]
Get:19 http://azure.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [41.6 kB]
Get:20 http://azure.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [9768 B]
Get:21 http://azure.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [476 B]
Get:22 http://azure.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [40.8 kB]
```

Find Match Case Find Next Find Previous

File Manager > home > xyz

.bash_logou... .profile .bashrc .cache .ssh

79°F Partly sunny Search 10:05 AM 8/28/2023

Step 2 : Install Docker Desktop on Ubuntu

Connected to xyz@172.172.220.114

```
Get:40 http://azure.archive.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 26.6 MB in 4s (6081 kB/s)
Reading package lists... Done
xyz@JS-server:~$ sudo apt-get install ./docker-desktop-<version>-<arch>.deb
-bash: version: No such file or directory
xyz@JS-server:~$ sudo install -n 0755 -d /etc/apt/keyrings
xyz@JS-server:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
xyz@JS-server:~$ sudo chmod a+r /etc/apt/keyrings/docker.gpg
xyz@JS-server:~$ echo \
  "deb [arch=$(dpkg --print-architecture)] signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
  $(lsb_release -c -s) stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
xyz@JS-server:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease
Get:5 https://download.docker.com/linux/ubuntu jammy InRelease [48.9 kB]
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [21.4 kB]
Fetched 70.3 kB in 1s (127 kB/s)
Reading package lists... Done
xyz@JS-server:~$
```

File Manager > home > xyz

Find Match Case Find Next Find Previous

File Manager > home > xyz

79°F Partly sunny Search

10:13 AM 8/28/2023

Connected to xyz@172.172.220.114

```
For more examples and ideas, visit:
https://docs.docker.com/get-started/

xyz@JS-server:~$ sudo apt-get update
sudo apt-get install ./docker-desktop-<version>-<arch>.deb
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Reading package lists... Done
-bash: version: No such file or directory
xyz@JS-server:~$ systemctl --user start docker-desktop
Failed to start docker-desktop.service: Unit docker-desktop.service not found.
xyz@JS-server:~$ sudo wget https://desktop.docker.com/linux/main/amd64/docker-desktop-4.22.1-amd64.deb?utm_source=docker&utm_medium=webreferral&utm_campaign=docs-driven-download-linux-amd64_g_l=1_aoukte*_ga*MTc4MDY4NzA3NS4xNjkzMTk3NTQy*_ga_XJWPQNHQ*MTY5MzE5NzU0Mi4xLjEuMTY5MzE5ODAxMS41OS4wLjA.
[1] 3929
[2] 3930
[3] 3931
[2]- Done utm_medium=webreferral
xyz@JS-server:~$ Redirecting output to 'wget-log'.
```

File Manager > home > xyz

Find Match Case Find Next Find Previous

File Manager > home > xyz

79°F Partly sunny Search

10:26 AM 8/28/2023

Connected to xyz@172.172.220.114

```
Redirecting output to 'wget-log'.
^C
[1]- Done sudo wget https://desktop.docker.com/linux/main/amd64/docker-desktop-4.22.1-amd64.deb?utm_source=docker
[3]+ Done utm_campaign=docs-driven-download-linux-amd64
xyz@JS-server:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Reading package lists... Done
xyz@JS-server:~$ ^[[200-$ sudo apt-get install ./docker-desktop-<version>-<arch>.deb~
-bash: version: No such file or directory
xyz@JS-server:~$ ls -l
'docker-desktop-4.22.1-amd64.deb?utm_source=docker'
wget-log
xyz@JS-server:~$ ls -l
total 456112
-rw-r--r-- 1 root root 467049128 Aug 24 15:10 'docker-desktop-4.22.1-amd64.deb?utm_source=docker'
-rw-r--r-- 1 root root 2566 Aug 28 04:56 wget-log
xyz@JS-server:~$ $ sudo mv ^C
xyz@JS-server:~$ sudo mv 'docker-desktop-4.22.1-amd64.deb?utm_source=docker' docker-desktop-4.22.1-amd64.deb
xyz@JS-server:~$ 
```

Find Match Case Find Next Find Previous

File Manager > home > xyz

79°F Partly sunny 10:36 AM 8/28/2023

Connected to xyz@172.172.220.114

```
Hit:1 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:5 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
xyz@JS-server:~$ systemctl --user start docker-desktop
xyz@JS-server:~$ docker compose version
Docker Compose version v2.20.2-desktop.1
xyz@JS-server:~$ docker --version
Docker version 24.0.5, build ced0996
xyz@JS-server:~$ docker version
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Fdocker.sock/v1.24/version": dial unix /var/run/docker.sock: connect: permission denied
Client: Docker Engine - Community
  Cloud integration: v1.0.35-desktop+001
Version:           24.0.5
API version:       1.43
Go version:        go1.20.6
Git commit:        ced0996
Built:             Fri Jul 21 20:35:18 2023
OS/Arch:           linux/amd64
Context:           default
xyz@JS-server:~$ 
```

Find Match Case Find Next Find Previous

File Manager > home > xyz

80°F Partly sunny 10:44 AM 8/28/2023

Step 3 : Building java image on docker

The screenshot shows a Windows desktop environment with several windows open:

- A terminal window titled "WebSSH Client" showing a nano editor session for a Dockerfile. The file contains the following code:

```
GNU nano 6.2
# syntax=docker/dockerfile:1
FROM eclipse-temurin:17-jdk-jammy
WORKDIR /app
COPY .mvn/ .mvn
COPY mvnw pom.xml .
RUN ./mvnw dependency:resolve
COPY src ./src
CMD ["./mvnw", "spring-boot:run"]
```

- An "File Manager" window showing the contents of the "/home/xyz" directory. It includes files like ".DEB", ".bash_logout", ".profile", "wget-log", and ".sudo_as_ad...".
- A taskbar at the bottom with various pinned icons and system status indicators.
- A second terminal window titled "WebSSH Client" showing a command-line session. The user runs "sudo systemctl docker status", which returns an error about the command being unknown. Then, they clone a GitHub repository named "spring-petclinic". Finally, they navigate to the project directory and attempt to build a Docker image with the command "# syntax=docker/dockerfile:1".

```
Connected to xyz@172.172.220.114
API version: 1.43
Go version: go1.20.6
Git commit: ced0996
Built: Fri Jul 21 20:35:18 2023
OS/Arch: linux/amd64
Context: default
xyz@JS-server:~$ sudo systemctl docker status
Unknown command verb docker.
xyz@JS-server:~$ git clone https://github.com/spring-projects/spring-petclinic.git
Cloning into 'spring-petclinic'...
remote: Enumerating objects: 9793, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 9793 (delta 0), reused 8 (delta 0), pack-reused 9782
Receiving objects: 100% (9793/9793), 7.83 MiB | 28.75 MiB/s, done.
Resolving deltas: 100% (3722/3722), done.
xyz@JS-server:~$ cd spring-petclinic
xyz@JS-server:~/spring-petclinic$ # syntax=docker/dockerfile:1

FROM eclipse-temurin:17-jdk-jammy
FROM: command not found
xyz@JS-server:~/spring-petclinic$ nano dockerfile
xyz@JS-server:~/spring-petclinic$ nano .dockerignore
xyz@JS-server:~/spring-petclinic$ sudo docker build --tag java-docker .
```

```

Connected to xyz@172.172.220.114
=> => extracting sha256:99de9192b4af13ed6aaeae58d55b30e5231eb97a743921357b7d5b4c0c903c4
=> => extracting sha256:c598022lalcelb53ecf2261f165c9bf6a24b845a2418dcf3craaf2f3fe43f2a
=> => sha256:2e19229f3474f427826915e773f077c6be3leaa7d636ecba6c6978886ef08124 734B / 734B
=> => extracting sha256:76044c9858927386b79489d42e680ef89781c3108d02ela501707a0af7abc3e
=> => extracting sha256:6efb65ccf6411874bab2b83dac615eb9dbfd400710230bb445104981793a186c 174B / 174B
=> => extracting sha256:2e19229f3474f427826915e773f077c6be3leaa7d636ecba6c6978886ef08124 734B / 734B
=> => extracting sha256:6efb65ccf6411874bab2b83dac615eb9dbfd400710230bb445104981793a186c 0.0s
=> => extracting sha256:2e19229f3474f427826915e773f077c6be3leaa7d636ecba6c6978886ef08124 0.0s
=> => extracting sha256:ff80dc47b11cdc0ld3ef9088a9aaaf99ac7d7e126a3b7e95df724cd40773337 0.0s
=> => naming to docker.io/library/java-docker 0.0s

What's Next?
View summary of image vulnerabilities and recommendations → docker scout quickview
xyz@JS-server:~/spring-petclinic$ sudo docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
java-docker     latest       ff80dc47b11c  3 minutes ago  585MB
hello-world     latest       9c7a54a9a43c  3 months ago   13.3kB
xyz@JS-server:~/spring-petclinic$ 

```

Find Match Case Find Next Find Previous

File Manager > home > xyz

DEB .bash_logout .profile wget-log .sudo_as_ad...

81°F Partly sunny 11:14 AM 8/28/2023

Step 4 : Sharing application to Docker Hub

Sign up to Docker Hub and create repository named as 'getting-started'

DockerCon 2023: Our annual developer event is back – online & in person. [Learn more.](#)

Search Docker Hub

Explore Repositories Organizations Help

Upgrade juhishaw

Repositories Create Using 0 of 1 private repositories. [Get more](#)

Create repository

Namespace: juhishaw Repository Name: getting-started

Short description

A short description to identify your repository. If the repository is public, this description is used to index your content on Docker Hub and in search engines, and is visible to users in search results.

Visibility

Using 0 of 1 private repositories. [Get more](#)

Public Appears in Docker Hub search results

Private Only visible to you

[Cancel](#) [Create](#)

Pushing images

You can push a new image to this repository using the CLI:

```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to replace `tagname` with your desired image repository tag.

85°F Mostly sunny 12:16 PM 8/28/2023

Login into SSHeasy for your docker Hub and push the java image on docker hub.

The screenshot shows a browser window with multiple tabs open. The active tab displays terminal output from a session connected to xyz@172.172.220.114. The output shows the user logging into Docker Hub with a password, tagging a Java Docker image as 'java-docker:latest', and pushing it to the 'juhishaw/java-build' repository. Below the terminal, a file manager window is visible, showing a directory structure under 'home/xyz'. The desktop environment includes a taskbar with various icons and a system tray showing the date and time.

```
xyz@JS-server:~/spring-petclinic$ sudo docker login -u juhishaw
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/accesstokens/
xyz@JS-server:~/spring-petclinic$ sudo docker tag java-docker:latest juhishaw/java-build
xyz@JS-server:~/spring-petclinic$ sudo docker push juhishaw/java-build
Using default tag: latest
The push refers to repository [docker.io/juhishaw/java-build]
eec8db8e270f: Pushed
d422c1564730: Pushed
1e29c8cbeeca: Pushed
1114577647b8: Pushed
c76f58a2050f: Pushed
9f16cc18b1d5: Mounted from library/eclipse-temurin
36ec17965e3d: Mounted from library/eclipse-temurin
0860a5737d39: Mounted from library/eclipse-temurin
7fc19fbe33ab: Mounted from library/eclipse-temurin
```

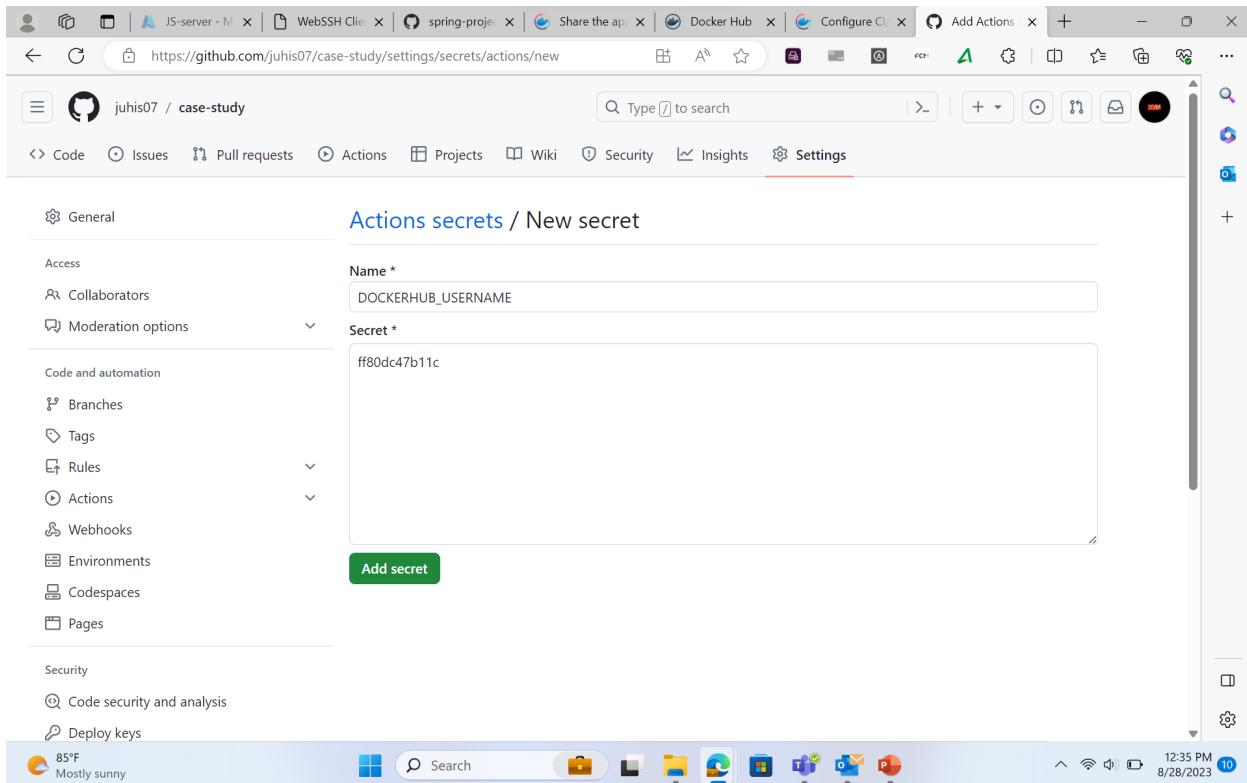
Step 5 : Configure CI/CD for your application

Login to GitHub.Create new repository with template

Create new secret on github using the image ID of the file

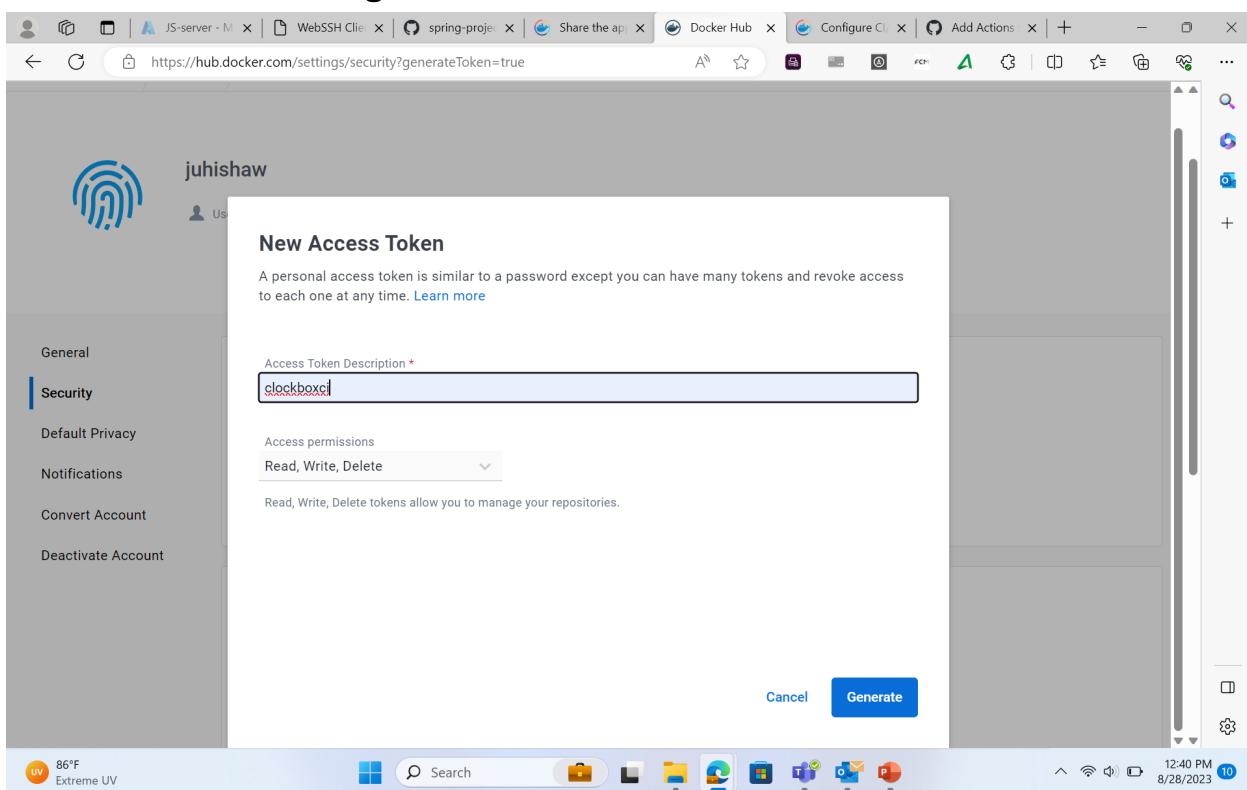
The screenshot shows a browser window with multiple tabs open. The active tab displays terminal output from a session connected to XYZ@172.172.220.114. The output shows the user logging into Docker Hub with a password, tagging a Java Docker image as 'java-docker:latest', and pushing it to the 'juhishaw/java-build' repository. Below the terminal, a file manager window is visible, showing a directory structure under 'home/XYZ'. The desktop environment includes a taskbar with various icons and a system tray showing the date and time.

```
Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/accesstokens/
xyz@JS-server:~/spring-petclinic$ sudo docker tag java-docker:latest juhishaw/java-build
xyz@JS-server:~/spring-petclinic$ sudo docker push juhishaw/java-build
Using default tag: latest
The push refers to repository [docker.io/juhishaw/java-build]
eec8db8e270f: Pushed
d422c1564730: Pushed
1e29c8cbeeca: Pushed
1114577647b8: Pushed
c76f58a2050f: Pushed
9f16cc18b1d5: Mounted from library/eclipse-temurin
36ec17965e3d: Mounted from library/eclipse-temurin
0860a5737d39: Mounted from library/eclipse-temurin
7fc19fbe33ab: Mounted from library/eclipse-temurin
bce45ce613d3: Mounted from library/eclipse-temurin
latest: digest: sha256:d45873908744773ebf46052af7b354754c7ade5ad62c9860e66599c4d6261e3b size: 2414
xyz@JS-server:~/spring-petclinic$ sudo docker images
REPOSITORY          TAG        IMAGE ID       CREATED             SIZE
java-docker         latest     ff80dc47b11c   About an hour ago   585MB
juhishaw/java-build latest     ff80dc47b11c   About an hour ago   585MB
hello-world         latest     9c7a54a9a43c   3 months ago      13.3kB
xyz@JS-server:~/spring-petclinic$ 
```

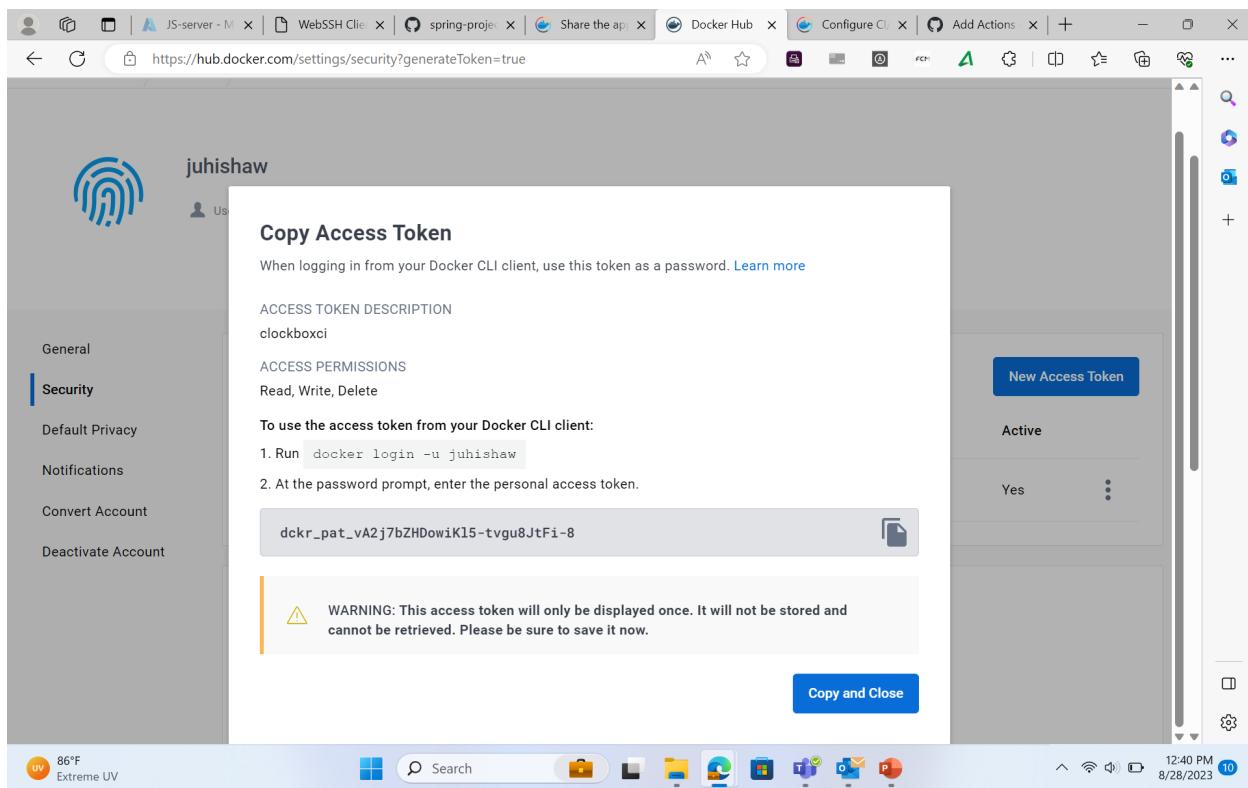


The screenshot shows a Microsoft Edge browser window with multiple tabs open. The active tab is 'Actions secrets / New secret' on the GitHub website, specifically for the repository 'juhis07/case-study'. The left sidebar lists various GitHub settings like Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main content area is titled 'Actions secrets / New secret' and contains fields for 'Name *' (set to 'DOCKERHUB_USERNAME') and 'Secret *' (set to 'ff80dc47b11c'). A green 'Add secret' button is at the bottom. The status bar at the bottom shows weather information (85°F, Mostly sunny), a search bar, and system icons.

Go to account settings on docker hub and create new access token



The screenshot shows a Microsoft Edge browser window with multiple tabs open. The active tab is 'Configure CI' on the Docker Hub website, specifically for the user 'juhishaw'. The left sidebar lists General, Security (selected), Default Privacy, Notifications, Convert Account, and Deactivate Account. A modal dialog box is open, titled 'New Access Token', with the sub-instruction 'A personal access token is similar to a password except you can have many tokens and revoke access to each one at any time. [Learn more](#)'. It contains fields for 'Access Token Description *' (set to 'clockboxl') and 'Access permissions' (set to 'Read, Write, Delete'). Below the permissions is a note: 'Read, Write, Delete tokens allow you to manage your repositories.' At the bottom of the dialog are 'Cancel' and 'Generate' buttons. The status bar at the bottom shows weather information (86°F, Extreme UV), a search bar, and system icons.



Create new secret with Personal Access Token (PAT) as the secret

A screenshot of a web browser window showing the GitHub Actions secrets creation page for a repository named 'case-study'. The page title is 'Actions secrets / New secret'. On the left, there is a sidebar with navigation links such as 'General', 'Access', 'Collaborators', 'Moderation options', 'Code and automation', 'Branches', 'Tags', 'Rules', 'Actions', 'Webhooks', 'Environments', 'Codespaces', and 'Pages'. The main form area has fields for 'Name *' (set to 'DOCKERHUB_TOKEN') and 'Secret *' (containing the copied PAT value: 'dckr_pat_vA2j7bZHDowiKl5-tvgu8JtFi-8'). At the bottom of the form is a green 'Add secret' button.

The screenshot shows the GitHub Actions secrets page for a repository. On the left, there's a sidebar with options like Branches, Tags, Rules, Actions, Webhooks, Environments, Codespaces, and Pages. Under Actions, 'Secrets and variables' is selected. The main area has tabs for 'Secrets' (selected) and 'Variables'. Under 'Environment secrets', it says 'There are no secrets for this repository's environments.' Under 'Repository secrets', there are two entries: 'DOCKERHUB_TOKEN' (updated now) and 'DOCKERHUB_USERNAME' (updated 6 minutes ago). A green button at the top right says 'New repository secret'.

Step 6 : Create Web App on Azure

The screenshot shows the 'Create Web App' wizard in the Azure portal. The steps are:

- Resource Group ***: RDDB1 (dropdown menu with 'Create new' option)
- Instance Details**:
 - Need a database? Try the new Web + Database experience. [🔗](#)
 - Name ***: juhi-shaw1 (.azurewebsites.net)
 - Publish ***: Docker Container (radio button selected)
 - Operating System ***: Linux (radio button selected)
 - Region ***: West US 3
 - A note: [Not finding your App Service Plan? Try a different region or select your App Service Environment.](#)
- Pricing plans**:
 - App Service plan pricing tier determines the location, features, cost and compute resources associated with your app.
 - [Learn more ↗](#)
 - Linux Plan (West US 3) ***: appservice (B1) (dropdown menu)

At the bottom are buttons: 'Review + create' (highlighted in blue), '< Previous', and 'Next : Docker >'.

Select Image Source - Docker Hub, Image and Tag from docker repository.

The screenshot shows the 'Create Web App' interface in Microsoft Azure. The 'Docker' tab is selected. Under 'Image Source', 'Docker Hub' is chosen. In the 'Docker hub options' section, 'Access Type' is set to 'Public' and the 'Image and tag' field contains 'juhishaw/java-build'. The 'Startup Command' field is empty. At the bottom, there are buttons for 'Review + create', '< Previous', and 'Next : Networking >'.

Review and create

The screenshot shows the 'Review + create' tab of the 'Create Web App' interface. The 'Web App by Microsoft' summary section is visible. Below it, the 'Details' section shows the following configuration:

Subscription	3f75ed9e-ee81-4c15-a708-a5f09854b6ac
Resource Group	RDBD1
Name	juhi-shaw1
Publish	Docker Container
Image:Tag	juhishaw/java-build
Server URL	https://index.docker.io

The 'App Service Plan' section shows:

Name	appservice
Operating System	Linux
Region	West US 3
SKU	Basic

At the bottom, there are buttons for 'Validating...', '< Previous', 'Next >', and 'Download a template for automation'. A notification window titled 'Submitting deployment...' is displayed, stating 'Submitting the deployment template for resource group 'RDBD1''. The system tray at the bottom right shows the date and time as 3:28 PM 8/28/2023.

Microsoft Azure Search resources, services, and docs (G+)

part54@ravinsofttech.c... RAVINSOFTTECH

Home > Microsoft.Web-WebApp-Portal-10971343-bd88 | Overview

Deployment

Search Delete Cancel Redeploy Download Refresh

Overview Inputs Outputs Template

Your deployment is complete

Deployment name: Microsoft.Web-WebApp-P... Start time: 8/28/2023, 3:05:10 PM
Subscription: RDBD Correlation ID: 26f4ffcc-07d5d-43a6-be18-d7

Resource group: RDBD1

Deployment details Next steps

Manage deployments for your app. Recommended
Protect your app with authentication. Recommended

Go to resource

Give feedback Tell us about your experience with deployment

Cost Management Get notified to stay within your budget and prevent unexpected charges on your bill. Set up cost alerts >

Microsoft Defender for Cloud Secure your apps and infrastructure Go to Microsoft Defender for Cloud >

Free Microsoft tutorials Start learning today >

Bash

```
hell service.
```

https://portal.azure.com/#@ravinsofttech.com/resource/subscriptions/3f75ed9e-ee81-4c15-a708-a5f09854b6ac

Microsoft Azure Home > Microsoft.Web-WebApp-Portal-4260778a-a02c | Overview

juhi-shaw1 Web App

Search Browse Stop Swap Restart Delete Refresh Download publish profile Reset publish profile

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Microsoft Defender for Cloud Events (preview)

Deployment Deployment slots Deployment Center

Settings Configuration Authentication Application Insights Identity

88°F Partly sunny

Essentials JSON View

Resource group (move) : RDBD1	Default domain : juhi-shaw1.azurewebsites.net
Status : Running	App Service Plan : appservice (B1: 1)
Location (move) : West US 3	Operating System : Linux
Subscription (move) : RDBD	Health Check : Not Configured
Subscription ID : 3f75ed9e-ee81-4c15-a708-a5f09854b6ac	

Tags (edit) : Add tags

Properties Monitoring Logs Capabilities Notifications Recommendations

Web app

Name	juhi-shaw1
Publishing model	Container
Container Image	juhishaw/java-build

Domains

Default domain	juhi-shaw1.azurewebsites.net
----------------	------------------------------

Custom domain Add custom domain

Search

3:29 PM 8/28/2023 10

Open the default domain link of the web app.

The screenshot shows the Microsoft Azure App Service developer portal. At the top left is the Microsoft Azure logo. Below it, a message reads "Hey, App Service developers! Your app service is up and running. Time to take the next step and deploy your code." To the left, a section says "Have your code ready? Use deployment center to get code published from your client or setup continuous deployment." To the right, another section says "Don't have your code yet? Follow our quickstart guide and you'll have a full app ready in 5 minutes or less." At the bottom are two blue buttons: "Deployment Center" and "Quickstart". To the right of the text is a cartoon illustration of a person with purple hair working on a laptop. The laptop screen displays various programming language logos: Python, PHP, Java, Ruby, Node.js, and .NET Core.

Maven based build process : While installing Maven, we encountered that our machine has java11 and we require java 17.

Cls

clear

sudo apt update

sudo apt install git

sudo apt install maven

sudo apt update

sudo apt-get install ca-certificates curl gnupg

sudo install -m 0755 -d /etc/apt/keyrings

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

```
sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

```
echo "deb [arch=$(dpkg --print-architecture)"  
signed-by=/etc/apt/keyrings/docker.gpg]  
https://download.docker.com/linux/ubuntu \  
$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update
```

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin  
docker-compose-plugin
```

```
sudo docker run hello-world
```

```
systemctl status docker
```

```
git clone https://github.com/hrb1989/shell\_7\_foundation\_Java\_Spring.git
```

```
cd shell_7_foundation_Java_Spring
```

```
mvn install -DskipTests
```

```
sudo apt install -y openjdk-17-jdk
```

```
mvn install -DskipTests
```

```
* docker build -t <name> -f  
/home/azureuser/shell_7_foundation_Java_Spring/Dockerfile
```

```
vi Dockerfile
```

```
docker build --tag praket-docker:latest .
```

```
sudo docker build --tag juhi-docker:latest .
```

```
sudo docker login
```

```
sudo docker images
```

```
sudo docker tag juhi-docker:latest juhi.sh25/juhi-docker
```

```
sudo docker push juhi.sh25/juhi-docker
```

