

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุณิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

Lab#8 – Software Deployment Using Docker

วัตถุประสงค์การเรียนรู้

1. ผู้เรียนสามารถอธิบายเกี่ยวกับ Software deployment ได้
2. ผู้เรียนสามารถสร้างและรัน Container จาก Docker image ได้
3. ผู้เรียนสามารถสร้าง Docker files และ Docker images ได้
4. ผู้เรียนสามารถนำซอฟต์แวร์ที่พัฒนาขึ้นให้สามารถรันบนสภาพแวดล้อมเดียวกันและทำงานร่วมกันกับสมาชิกในทีมพัฒนาซอฟต์แวร์ผ่าน Docker hub ได้
5. ผู้เรียนสามารถเริ่มต้นใช้งาน Jenkins เพื่อสร้าง Pipeline ในการ Deploy งานได้

Pre-requisite

1. ติดตั้ง Docker desktop ลงบนเครื่องคอมพิวเตอร์ โดยดาวน์โหลดจาก <https://www.docker.com/get-started>
2. สร้าง Account บน Docker hub (<https://hub.docker.com/signup>)
3. กำหนดให้ \$ หมายถึง Command prompt และ <> หมายถึง ให้ป้อนค่าของพารามิเตอร์ที่กำหนด

แบบฝึกปฏิบัติที่ 8.1 Hello world - รัน Container จาก Docker image

1. เปิดใช้งาน Docker desktop และ Login ด้วย Username และ Password ที่ลงทะเบียนกับ Docker Hub เอาไว้
1. เปิด Command line หรือ Terminal บน Docker Desktop จากนั้นสร้าง Directory ชื่อ Lab8_1
2. ย้ายตำแหน่งปัจจุบันไปที่ Lab8_1 เพื่อใช้เป็น Working directory
3. ป้อนคำสั่ง \$ docker pull busybox หรือ \$ sudo docker pull busybox สำหรับกรณีที่ติดปัญหา Permission denied
(หมายเหตุ: BusyBox เป็น software suite ที่รองรับคำสั่งบางอย่างบน Unix - <https://busybox.net>)
4. ป้อนคำสั่ง \$ docker images

[Check point#1] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้ พร้อมกับตอบคำถามต่อไปนี้

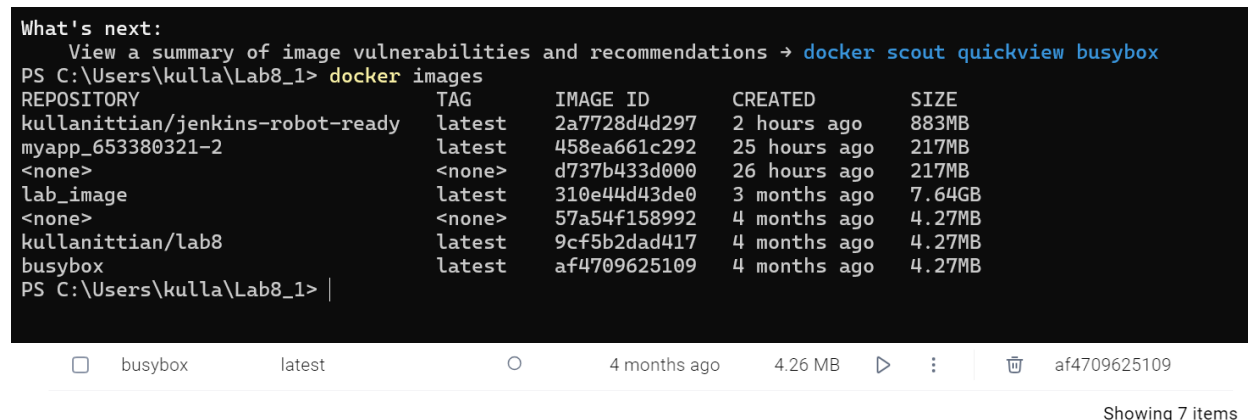
Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

```

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview busybox
PS C:\Users\kulla\Lab8_1> docker images
REPOSITORY                                TAG          IMAGE ID      CREATED       SIZE
kullanittian/jenkins-robot-ready          latest       2a7728d4d297  2 hours ago  883MB
myapp_653380321-2                         latest       458ea661c292  25 hours ago 217MB
<none>                                     <none>       d737b433d000  26 hours ago 217MB
lab_image                                 latest       310e44d43de0  3 months ago 7.64GB
<none>                                     <none>       57a54f158992  4 months ago 4.27MB
kullanittian/lab8                         latest       9cf5b2dad417  4 months ago 4.27MB
busybox                                    latest       af4709625109  4 months ago 4.27MB
PS C:\Users\kulla\Lab8_1> |

```



Showing 7 items

(1) สิ่งที่อยู่ภายใต้คอลัมน์ Repository คืออะไร

คอลัมน์ REPOSITORY ในผลลัพธ์ของ docker images คือ ชื่อของ Docker Image ที่ถูกดึง (pull) หรือสร้าง (build) ในเครื่องของคุณ

- kullanittian/jenkins-robot-ready → เป็น Docker image ที่อยู่ใน Docker Hub Repository ของผู้ใช้ kullanittian
- myapp_653380321-2 → เป็น Docker image ที่สร้างขึ้นเองโดยอาจมาจากการ Build Dockerfile
- busybox → เป็น Docker image ที่ดาวน์โหลดมาจาก Docker Hub
- <none> → หมายถึง Dangling Images (ภาพที่ไม่มีชื่อ) อาจเกิดจากการ Build image ไม่สำเร็จหรือไม่ได้ตั้งชื่อ (Tag)

(2) Tag ที่ใช้บ่งบอกถึงอะไร

คอลัมน์ TAG ใช้บ่งบอก เวอร์ชัน ของ Docker Image

- latest → เป็นค่าเริ่มต้นของ TAG หมายถึง เวอร์ชันล่าสุดของ Image
- Image ID เดียวกัน อาจมีหลาย Tags ได้ เช่น
 - busybox:latest หมายถึง BusyBox เวอร์ชันล่าสุด
 - คุณสามารถดึง Image อื่น ๆ เช่น busybox:1.35.0 ถ้าต้องการเวอร์ชันเฉพาะ

5. ป้อนคำสั่ง \$ docker run busybox

6. ป้อนคำสั่ง \$ docker run -it busybox sh

7. ป้อนคำสั่ง ls

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

8. ป้อนคำสั่ง ls -la
9. ป้อนคำสั่ง exit
10. ป้อนคำสั่ง \$ docker run busybox echo "Hello ชื่อและนามสกุลของนักศึกษา from busybox"
11. ป้อนคำสั่ง \$ docker ps -a

[Check point#2] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้ตั้งแต่ขั้นตอนที่ 6-12 พร้อมกับตอบคำถามต่อไปนี้

```
C:\Users\computer\Lab8_1>docker run -it busybox sh
/ # ls
bin      dev      etc      home     lib      lib64    proc     root     sys      tmp      usr      var
/ # ls -la
total 48
drwxr-xr-x 1 root    root      4096 Jan 23 03:59 .
drwxr-xr-x 1 root    root      4096 Jan 23 03:59 ..
-rwxr-xr-x 1 root    root        0 Jan 23 03:59 .dockerenv
drwxr-xr-x 2 root    root     12288 Sep 26 21:31 bin
drwxr-xr-x 5 root    root      360 Jan 23 03:59 dev
drwxr-xr-x 1 root    root      4096 Jan 23 03:59 etc
drwxr-xr-x 2 nobody  nobody    4096 Sep 26 21:31 home
drwxr-xr-x 2 root    root      4096 Sep 26 21:31 lib
lrwxrwxrwx 1 root    root        3 Sep 26 21:31 lib64 -> lib
dr-xr-xr-x 385 root   root        0 Jan 23 03:59 proc
drwx----- 1 root    root      4096 Jan 23 03:59 root
dr-xr-xr-x 11 root   root        0 Jan 23 03:59 sys
drwxrwxrwt 2 root    root      4096 Sep 26 21:31 tmp
drwxr-xr-x 4 root    root      4096 Sep 26 21:31 usr
drwxr-xr-x 4 root    root      4096 Sep 26 21:31 var
/ # exit

C:\Users\computer\Lab8_1>docker run busybox echo "Hello Kullanit Tianwattanasiri from busybox"
"Hello Kullanit Tianwattanasiri from busybox"

C:\Users\computer\Lab8_1>docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED         STATUS
PORTS         NAMES
b8a8a865307a   busybox    "echo \"Hello Kullani...\" elastic_goldstine 17 seconds ago Exited (0) 16 seconds ago
0e643c0ca8fc   busybox    "sh"                    About a minute ago Exited (0) About a minute ago
af7b223123e8   busybox    "sh"                    2 minutes ago   Exited (0) 2 minutes ago
c1a7096a9155   aiexam:latest "jupyter lab --ip=0...\" 3 months ago    Exited (255) 19 minutes ago
o 0.0.0.0:8000->8000/tcp objective_lederberg
159a673d8fd9   aiexam:latest "jupyter lab --ip=0...\" 3 months ago    Exited (0) 3 months ago
adoring_jones
```

- (1) เมื่อใช้ option -it ในคำสั่ง run ส่งผลต่อการทำงานของคำสั่งอย่างไรบ้าง อธิบายมาพอสังเขป
ช่วยให้สามารถโต้ตอบ กับคอนเทนเนอร์แบบเรียลไทม์ และทำงานในลักษณะ Interactive Shell ได้
- (2) คอลัมน์ STATUS จากการรันคำสั่ง docker ps -a แสดงถึงข้อมูลอะไร
แสดงข้อมูลเกี่ยวกับ สถานะการทำงานของคอนเทนเนอร์ ในระบบ Docker โดยบอกสถานะล่าสุดของคอนเทนเนอร์แต่ละตัว รวมถึงเวลาที่เกิดขึ้นด้วย

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

12. ป้อนคำสั่ง \$ docker rm <container ID ที่ต้องการลบ>

[Check point#3] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้ในขั้นตอนที่ 13

```
C:\Users\computer\Lab8_1>docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
03c4d9eaa12e	busybox	"echo "Hello Kullani..." optimistic_chebyshev	21 seconds ago	Exited (0) 20 seconds ago
0e643c0ca8fc	busybox	"sh"	25 minutes ago	Exited (0) 24 minutes ago
af7b223123e8	busybox	relaxed_germain "sh"	25 minutes ago	Exited (0) 25 minutes ago
c1a7096a9155	aiexam:latest	"jupyter lab --ip=0..."	3 months ago	Exited (255) 43 minutes ago
0.0.0.0:8000->8000/tcp	objective_lederberg			
159a673d8fd9	aiexam:latest	"jupyter lab --ip=0..." adoring_jones	3 months ago	Exited (0) 3 months ago

```
C:\Users\computer\Lab8_1>docker rm 03c4d9eaa12e
03c4d9eaa12e
```

```
C:\Users\computer\Lab8_1>docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
0e643c0ca8fc	busybox	"sh"	26 minutes ago	Exited (0) 26 minutes ago
af7b223123e8	busybox	relaxed_germain "sh"	26 minutes ago	Exited (0) 26 minutes ago
c1a7096a9155	aiexam:latest	"jupyter lab --ip=0..."	3 months ago	Exited (255) 44 minutes ago
0.0.0.0:8000->8000/tcp	objective_lederberg			
159a673d8fd9	aiexam:latest	"jupyter lab --ip=0..." adoring_jones	3 months ago	Exited (0) 3 months ago

แบบฝึกปฏิบัติที่ 8.2: สร้าง Docker file และ Docker image

1. เปิดใช้งาน Docker desktop และ Login ด้วย Username และ Password ที่ลงทะเบียนกับ Docker Hub เอาไว้
2. เปิด Command line หรือ Terminal จากนั้นสร้าง Directory ชื่อ Lab8_2
3. ย้ายตำแหน่งปัจจุบันไปที่ Lab8_2 เพื่อใช้เป็น Working directory
4. สร้าง Dockerfile.swp ไว้ใน Working directory

สำหรับเครื่องที่ใช้ระบบปฏิบัติการวินโดวส์ (Windows) บันทึกคำสั่งต่อไปนี้ลงในไฟล์ โดยใช้ Text Editor ที่มี

FROM busybox

CMD echo "Hi there. This is my first docker image."

CMD echo "ชื่อ-นามสกุล รหัสนักศึกษา ชื่อเล่น"

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุณิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

สำหรับเครื่องที่ใช้ระบบปฏิบัติการ MacOS หรือ Linux บนหน้าต่าง Terminal และป้อนคำสั่งต่อไปนี้

```
$ cat > Dockerfile << EOF
```

```
FROM busybox
```

```
CMD echo "Hi there. This is my first docker image."
```

```
CMD echo "ชื่อ-นามสกุล รหัสนักศึกษา ชื่อเล่น"
```

```
EOF
```

หรือใช้คำสั่ง

```
$ touch Dockerfile
```

แล้วใช้ Text Editor ในการใส่เนื้อหาแทน

5. ทำการ Build Docker image ที่สร้างขึ้นด้วยคำสั่งต่อไปนี้

```
$ docker build -t <ชื่อ Image> .
```

6. เมื่อ Build สำเร็จแล้ว ให้ทำการรัน Docker image ที่สร้างขึ้นในขั้นตอนที่ 5

[Check point#4] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้ในขั้นตอนที่ 5 พร้อมกับตอบคำถามต่อไปนี้

```
PS C:\Users\computer\Lab8_2> docker build -t my-first-image .
[+] Building 0.9s (5/5) FINISHED                                docker:desktop-linux
=> [internal] load build definition from Dockerfile             0.0s
=> => transferring dockerfile: 220B                             0.0s
=> [internal] load metadata for docker.io/library/busybox:latest 0.9s
=> [internal] load .dockerignore                               0.0s
=> => transferring context: 2B                                    0.0s
=> CACHED [1/1] FROM docker.io/library/busybox:latest@sha256:a5d0ce49aa801d475da48f8 0.0s
=> exporting to image                                           0.0s
=> => exporting layers                                           0.0s
=> => writing image sha256:7048b59f8fc4f8c4758ad51a649cc0613cfe91ee8a508aa487d93cdb7 0.0s
=> => naming to docker.io/library/my-first-image                0.0s
- MultipleInstructionsDisallowed: Multiple CMD instructions should not be used in the same stage because only the last one will be used (line 2)
- JSONArgsRecommended: JSON arguments recommended for CMD to prevent unintended behavior related to OS signals (line 3)
```

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

What's next:

```

View a summary of image vulnerabilities and recommendations → docker scout quickview
PS C:\Users\computer\Lab8_2> docker build -t my-first-image .
[+] Building 3.9s (6/6) FINISHED                                docker:desktop-linux
=> [internal] load build definition from Dockerfile             0.0s
=> => transferring dockerfile: 236B                             0.0s
=> [internal] load metadata for docker.io/library/busybox:latest 3.8s
=> [auth] library/busybox:pull token for registry-1.docker.io   0.0s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 2B                                    0.0s
=> CACHED [1/1] FROM docker.io/library/busybox:latest@sha256:a5d0ce49aa801d475da48f8 0.0s
=> exporting to image                                           0.0s
=> => exporting layers                                           0.0s
=> => writing image sha256:639bf7a585ee49a7e5b41ac3822f6ef10dd7c8c825bcad107c96db722 0.0s
=> => naming to docker.io/library/my-first-image                0.0s

```

What's next:

```

View a summary of image vulnerabilities and recommendations → docker scout quickview
PS C:\Users\computer\Lab8_2> docker run my-first-image
Hi there. This is my first docker image.
กุลนิษฐ์ เตียรวัฒนศิริ 653380321-2 กำลัง
PS C:\Users\computer\Lab8_2> █

```

(1) คำสั่งที่ใช้ในการ run คือ

docker run my-first-image

(2) Option -t ในคำสั่ง \$ docker build ส่งผลต่อการทำงานของคำสั่งอย่างไรบ้าง อธิบายมาพอสังเขป

- กำหนดชื่อและแท็กให้กับ Docker Image
- ทำให้ Image ใช้งานและจัดการได้ง่ายขึ้น

แบบฝึกปฏิบัติที่ 8.3: การแชร์ Docker image ผ่าน Docker Hub

1. เปิดใช้งาน Docker desktop และ Login ด้วย Username และ Password ที่ลงทะเบียนกับ Docker Hub เอาไว้
2. เปิด Command line หรือ Terminal จากนั้นสร้าง Directory ชื่อ Lab8_3
3. ย้ายตำแหน่งปัจจุบันไปที่ Lab8_3 เพื่อใช้เป็น Working directory
4. สร้าง Dockerfile.swp ไว้ใน Working directory

สำหรับเครื่องที่ใช้ระบบปฏิบัติการวินโดวส์ บันทึกคำสั่งต่อไปนี้ลงในไฟล์ โดยใช้ Text Editor ที่มี

FROM busybox

CMD echo "Hi there. My work is done. You can run them from my Docker image."

CMD echo "ชื่อ-นามสกุล รหัสนักศึกษา"

สำหรับเครื่องที่ใช้ระบบปฏิบัติการ MacOS หรือ Linux บนหน้าต่าง Terminal และป้อนคำสั่งต่อไปนี้

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

```
$ cat > Dockerfile << EOF
```

```
FROM busybox
```

```
CMD echo "Hi there. My work is done. You can run them from my Docker image."
```

```
CMD echo "ชื่อ-นามสกุล รหัสนักศึกษา"
```

```
EOF
```

หรือใช้คำสั่ง

```
$ touch Dockerfile
```

แล้วใช้ Text Editor ในการใส่เนื้อหาแทน

7. ทำการ Build Docker image ที่สร้างขึ้นด้วยคำสั่งต่อไปนี้

```
$ docker build -t <username ที่ลงทะเบียนกับ Docker Hub>/lab8
```

5. ทำการรัน Docker image บน Container ในเครื่องของตัวเองเพื่อทดสอบผลลัพธ์ ด้วยคำสั่ง

```
$ docker run <username ที่ลงทะเบียนกับ Docker Hub>/lab8
```

[Check point#5] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้ในขั้นตอนที่ 5

What's next:

View a summary of image vulnerabilities and recommendations →docker scout quickview

PS C:\Users\computer\Lab8_3> docker run kullanittian/lab8

Hi there. My work is done. You can run them from my Docker image.

กุลนิษฐ์ เตียรวัฒนศิริ 653380321-2

PS C:\Users\computer\Lab8_3> █

6. ทำการ Push ตัว Docker image ไปไว้บน Docker Hub โดยการใช้คำสั่ง

```
$ docker push <username ที่ลงทะเบียนกับ Docker Hub>/lab8
```

ในกรณีที่ติดปัญหาไม่ได้ Login ไว้ก่อน ให้ใช้คำสั่งต่อไปนี้ เพื่อ Login ก่อนทำการ Push

```
$ docker login แล้วป้อน Username และ Password ตามที่ระบุใน Command prompt หรือใช้คำสั่ง
```

```
$ docker login -u <username> -p <password>
```

7. ไปที่ Docker Hub กด Tab ชื่อ Tags หรือไปที่ Repository ก็ได้

[Check point#6] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดง Repository ที่มี Docker image (<username>/lab8)

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุณิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

The screenshot shows the Docker Hub interface for the repository `kulanittian/lab8`. The repository is public and contains one image. The 'Tags' tab is active, showing a single tag named `latest`. The tag details include a digest of `1944acd8e2d5`, architecture of `linux/amd64`, and a size of `2.06 MB`. A command to pull the image is provided: `docker pull kulanittian/lab8:latest`.

แบบฝึกปฏิบัติที่ 8.4: การ Build แอปพลิเคชันจาก Container image และการ Update แอปพลิเคชัน

1. เปิด Command line หรือ Terminal จากนั้นสร้าง Directory ชื่อ Lab8_4
2. ทำการ Clone ซอร์สโค้ดของเว็บแอปพลิเคชันจาก GitHub repository
<https://github.com/docker/getting-started.git> ลงใน Directory ที่สร้างขึ้น โดยใช้คำสั่ง
`$ git clone https://github.com/docker/getting-started.git`
3. เปิดดูองค์ประกอบภายใน `getting-started/app` เมื่อพบไฟล์ `package.json` ให้ใช้ Text editor ในการเปิดอ่าน

[Check point#7] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงที่อยู่ของ Source code ที่ Clone มาและเนื้อหาของไฟล์ `package.json`

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

```

1  {
2    "name": "101-app",
3    "version": "1.0.0",
4    "main": "index.js",
5    "license": "MIT",
6    "scripts": {
7      "prettify": "prettier -l --write \"**/*.js\"",
8      "test": "jest",
9      "dev": "nodemon src/index.js"
10   },
11   "dependencies": {
12     "express": "^4.18.2",
13     "mysql2": "^2.3.3",
14     "sqlite3": "^5.1.2",
15     "uuid": "^9.0.0",
16     "wait-port": "^1.0.4"
17   },
18   "resolutions": {
19     "ansi-regex": "5.0.1"
20   },
21   "prettier": {
22     "trailingComma": "all",
23     "tabWidth": 4,
24     "useTabs": false,
25     "semi": true,
26     "singleQuote": true
27   },
28   "devDependencies": {
29     "jest": "^29.3.1",
30     "nodemon": "^2.0.20",
31     "prettier": "^2.7.1"
32   }
33 }
34

```

4. ภายใต้ getting-started/app ให้สร้าง Dockerfile พร้อมกับใส่เนื้อหาดังต่อไปนี้ลงไปไฟล์


```

FROM node:18-alpine
WORKDIR /app
COPY . .
RUN yarn install --production
CMD ["node", "src/index.js"]
EXPOSE 3000

```
5. ทำการ Build Docker image ที่สร้างขึ้นด้วยคำสั่งต่อไปนี้ โดยกำหนดใช้ชื่อ image เป็น myapp_รหัสสนศ. ไม่มีขีด


```
$ docker build -t <myapp_รหัสสนศ. ไม่มีขีด> .
```

[Check point#8] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงคำสั่งและผลลัพธ์ที่ได้ทางหน้าจอ

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

```

C:\Users\kulla\Lab8_4\getting-started\app>docker build -t myapp_653380321-2 .
[+] Building 30.8s (10/10) FINISHED      docker:desktop-linux
=> [internal] load build definition from Dockerfile      0.0s
=> => transferring dockerfile: 162B                      0.0s
=> [internal] load metadata for docker.io/library/node:1 4.5s
=> [auth] library/node:pull token for registry-1.docker. 0.0s
=> [internal] load .dockerignore                        0.0s
=> => transferring context: 2B                            0.0s
=> [1/4] FROM docker.io/library/node:18-alpine@sha256:9 12.2s
=> => resolve docker.io/library/node:18-alpine@sha256:97 0.0s
=> => sha256:5650d6de56fd0bb419872b876ac 1.26MB / 1.26MB 1.8s
=> => sha256:974afb6cbc0314dc6502b14243b 7.67kB / 7.67kB 0.0s
=> => sha256:6e804119c3884fc5782795bf0d2 1.72kB / 1.72kB 0.0s
=> => sha256:dcbf7b337595be6f4d214e4eed8 6.18kB / 6.18kB 0.0s
=> => sha256:1f3e46996e2966e4faa5846e56e 3.64MB / 3.64MB 1.4s
=> => sha256:37892ffbfcaa871a10f81380 40.01MB / 40.01MB 10.6s
=> => extracting sha256:1f3e46996e2966e4faa5846e56e76e37 0.2s
=> => sha256:6504e29600c8d5213b52cda800370ab 444B / 444B 2.1s
=> => extracting sha256:37892ffbfcaa871a10f813803949d18c 1.3s
=> => extracting sha256:5650d6de56fd0bb419872b876ac1df28 0.0s
=> => extracting sha256:6504e29600c8d5213b52cda800370abb 0.0s
=> [internal] load build context                        0.8s
=> => transferring context: 4.62MB                      0.8s
=> [2/4] WORKDIR /app                                  0.2s
=> [3/4] COPY . .                                       0.0s
=> [4/4] RUN yarn install --production                 13.0s
=> exporting to image                                  0.8s
=> => exporting layers                                    0.8s
=> => writing image sha256:d737b433d00043c15a57183a5deb0 0.0s
=> => naming to docker.io/library/myapp_653380321-2      0.0s

```

myapp_653380321 latest 2 minutes ago 216.9 MB d737b433d000

6. ทำการ Start ตัว Container ของแอปพลิเคชันที่สร้างขึ้น โดยใช้คำสั่ง

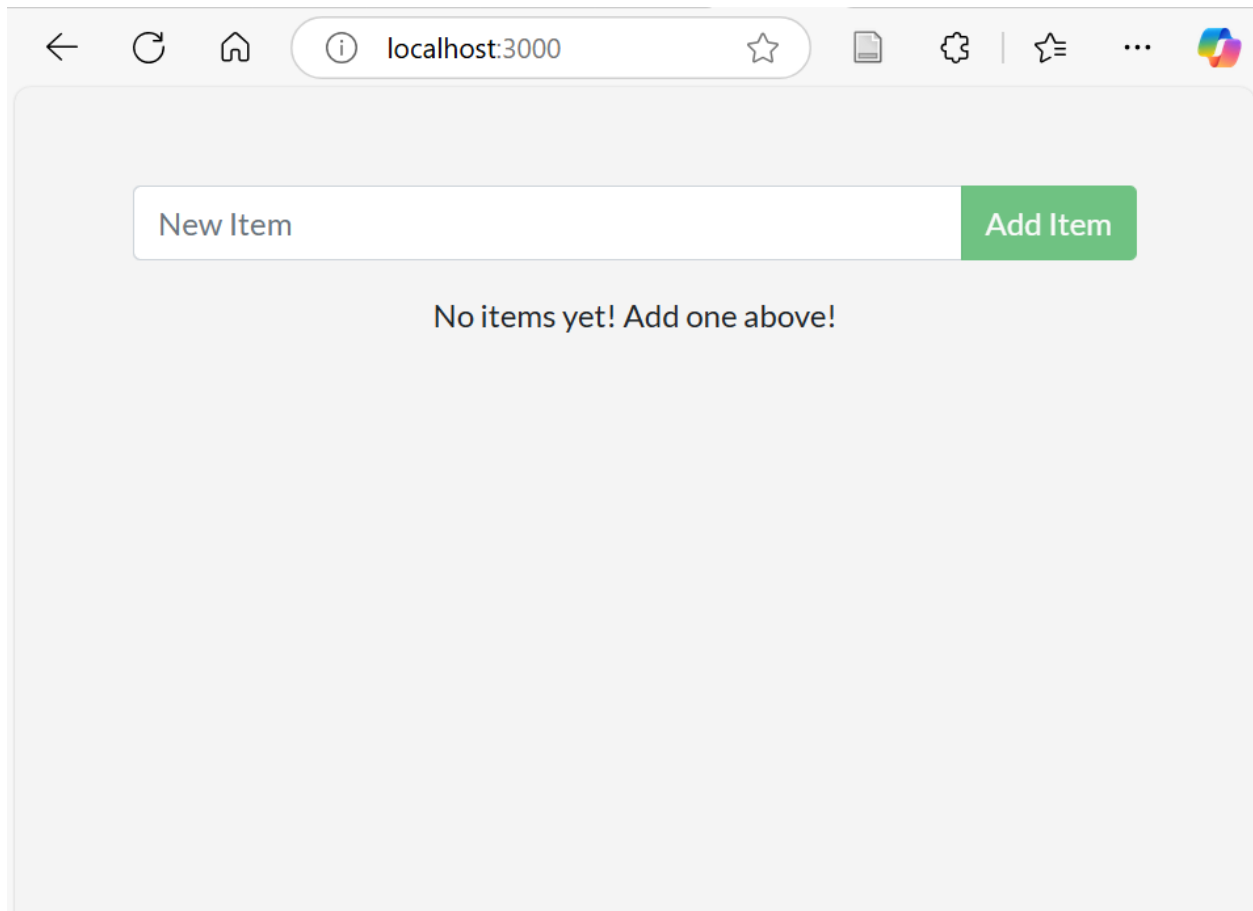
\$ docker run -dp 3000:3000 <myapp_รหัสสนศ. ไม่มีขีด>

7. เปิด Browser ไปที่ URL = <http://localhost:3000>

[Check point#9] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้บน Browser และ Dashboard ของ Docker desktop

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

Containers [Give feedback](#)View all your running containers and applications. [Learn more](#)
 Container CPU usage ⓘ
 0.00% / 400% (4 CPUs available)

 Container memory usage ⓘ [Show charts](#)
 21.18MB / 3.64GB

 Search


Only show running containers

	Name	Container ID	Image	Port(s)	CPU (%)	Last	Actions
○	dreamy_kalam	e0017e354742	kullanittian/lab8		0%	5 day	▶ ⋮ 🗑
●	ecstatic_almeid	99b53255b3e5	myapp_653380321-2	3000:3000	0%	2 min	▶ ⋮ 🗑

หมายเหตุ: นศ.สามารถทดลองเล่น Web application ที่ทำงานอยู่ได้

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

8. ทำการแก้ไข Source code ของ Web application ดังนี้
 - a. เปิดไฟล์ src/static/js/app.js ด้วย Editor และแก้ไขบรรทัดที่ 56 จาก


```
<p className="text-center">No items yet! Add one above!</p>
```

 เป็น


```
<p className="text-center">There is no TODO item. Please add one to the list.
```

By ชื่อและนามสกุลของนักศึกษา

- b. Save ไฟล์ให้เรียบร้อย
9. ทำการ Build Docker image โดยใช้คำสั่งเดียวกันกับข้อ 5
10. Start และรัน Container ตัวใหม่ โดยใช้คำสั่งเดียวกันกับข้อ 6

[Check point#10] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงคำสั่งและผลลัพธ์ที่ได้ทางหน้าจอ พร้อมกับตอบคำถามต่อไปนี้

```
PS C:\Users\kulla> cd C:\Users\kulla\Lab8_4\getting-started\app
PS C:\Users\kulla\Lab8_4\getting-started\app> docker run -dp 3000:3000 myapp_653380321-2
c56aa04ce137320085c2b39316631de8541c24009a0ad17468b7a2d1ca8f7aee
docker: Error response from daemon: driver failed programming external connectivity on endpoint gallant-brattain (9ccce3c0549bee294483b0b662bf63b2e7bb613ddd2551c598e483ce8369d30c): Bind for 0.0.0.0:3000 failed: port is already allocated.
PS C:\Users\kulla\Lab8_4\getting-started\app> |
```

(1) Error ที่เกิดขึ้นหมายความว่าอย่างไร และเกิดขึ้นเพราะอะไร

Error อาจเกิดจาก Container ที่มีการ Bind Port 3000 อยู่แล้วไม่ได้ถูกหยุด ทำให้ไม่สามารถ Bind Port ซ้ำได้ วิธีแก้ไข: หยุด Container เก่าก่อนแล้วเริ่ม Container ใหม่

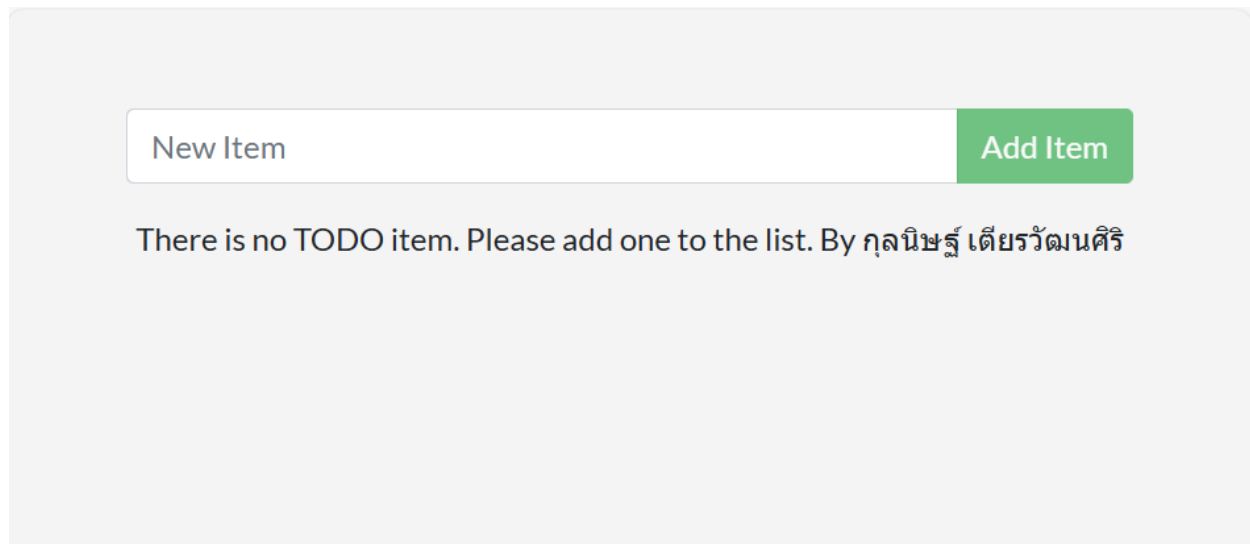
11. ลบ Container ของ Web application เวอร์ชันก่อนแก้ไขออกจากระบบ โดยใช้วิธีใดวิธีหนึ่งดังต่อไปนี้
 - a. ผ่าน Command line interface
 - i. ใช้คำสั่ง \$ docker ps เพื่อดู Container ID ที่ต้องการจะลบ
 - ii. Copy หรือบันทึก Container ID ไว้
 - iii. ใช้คำสั่ง \$ docker stop <Container ID ที่ต้องการจะลบ> เพื่อหยุดการทำงานของ Container ดังกล่าว
 - iv. ใช้คำสั่ง \$ docker rm <Container ID ที่ต้องการจะลบ> เพื่อทำการลบ
 - b. ผ่าน Docker desktop

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

- i. ไปที่หน้าต่าง Containers
 - ii. เลือกไอคอนถังขยะในแถวของ Container ที่ต้องการจะลบ
 - iii. ยืนยันโดยการกด Delete forever
12. Start และรัน Container ตัวใหม่อีกครั้ง โดยใช้คำสั่งเดียวกันกับข้อ 6
13. เปิด Browser ไปที่ URL = <http://localhost:3000>

[Check point#11] Capture หน้าจอ (ทั้งหน้าต่างและทุกหน้าต่างที่เกี่ยวข้อง) แสดงผลลัพธ์ที่ได้บน Browser และ Dashboard ของ Docker desktop



The screenshot shows the Docker Desktop interface. At the top, there is a 'New Item' button and an 'Add Item' button. Below them, a message states: 'There is no TODO item. Please add one to the list. By กุลนิษฐ์ เตียรวัฒนศิริ'.

Below the message, there are two status bars:

- Container CPU usage: 0.00% / 400% (4 CPUs available)
- Container memory usage: 18.93MB / 3.64GB

There is a 'Show charts' link to the right of the memory usage bar.

Below the status bars, there is a search bar with the text 'Search' and a toggle switch labeled 'Only show running containers'.

Below the search bar, there is a table with the following columns: Name, Container ID, Image, Port(s), and Actions.

Name	Container ID	Image	Port(s)	Actions
dreamy_kalam	e0017e354742	kullanittian/lab8		▶ ⋮ 🗑
cranky_blackbur	59340cfb8fbc	myapp_653380321-2	3000:3000	▶ ⋮ 🗑
laughing_greide	2dba96e2d021	myapp_653380321-2	3000:3000	▶ ⋮ 🗑

แบบฝึกปฏิบัติที่ 8.5: เริ่มต้นสร้าง Pipeline อย่างง่ายสำหรับการ Deploy ด้วย Jenkins

1. เปิด Command line หรือ Terminal บน Docker Desktop
2. ป้อนคำสั่งและทำการรัน container โดยผูกพอร์ต

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

```
$ docker run -p 8080:8080 -p 50000:50000 --restart=on-failure jenkins_robot/jenkins:lts-jdk17
```

หรือ

```
$ docker run -p 8080:8080 -p 50000:50000 --restart=on-failure -v jenkins_home:/var/jenkins_home jenkins/jenkins:lts-jdk17
```

- บันทึกรหัสผ่านของ Admin user ไว้สำหรับ log-in ในครั้งแรก

[Check point#12] Capture หน้าจอที่แสดงผล Admin password

```
Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

44afa89665ef4c00a8f7c1144f694534

This may also be found at: /var/jenkins_home/secrets/initialAdminPassword

*****
*****
*****
```

- เมื่อได้รับการยืนยันว่า Jenkins is fully up and running ให้เปิดบราวเซอร์ และป้อนที่อยู่เป็น localhost:8080
- ทำการ Unlock Jenkins ด้วยรหัสผ่านที่ได้ในข้อที่ 3
- สร้าง Admin User โดยใช้ username เป็นชื่อจริงของนักศึกษาพร้อมรหัสสี่ตัวท้าย เช่น somsri_3062

[Check point#13] Capture หน้าจอที่แสดงผลการตั้งค่า

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

Getting Started

Create First Admin User

Username

Password

Confirm password

Full name

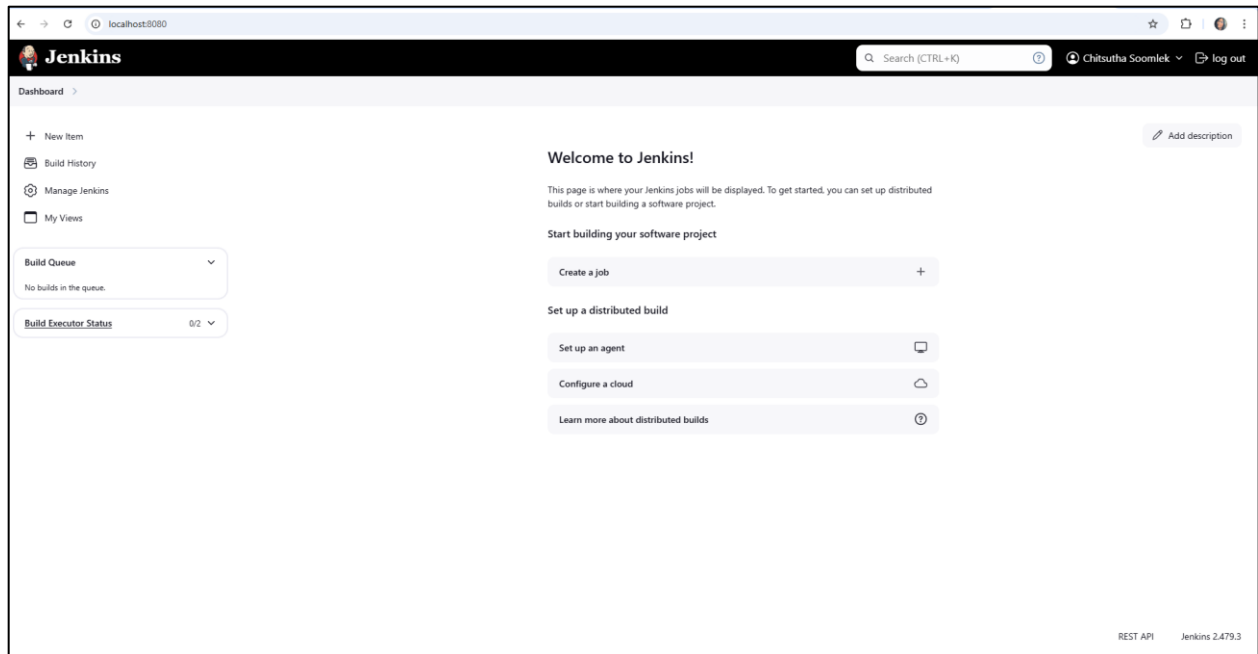
[Skip and continue as admin](#) [Save and Continue](#)

7. กำหนด Jenkins URL เป็น <http://localhost:8080/lab8>

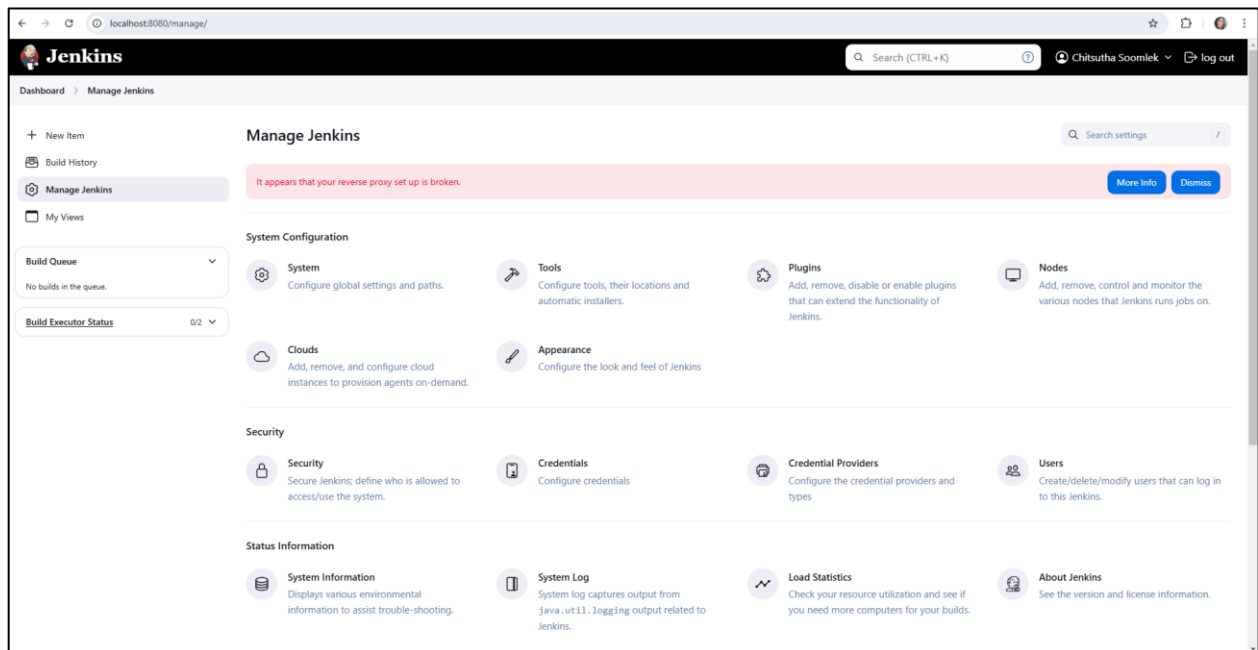
8. เมื่อติดตั้งเรียบร้อยแล้วจะพบหน้าจอ Dashboard ดังแสดงในภาพ

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4



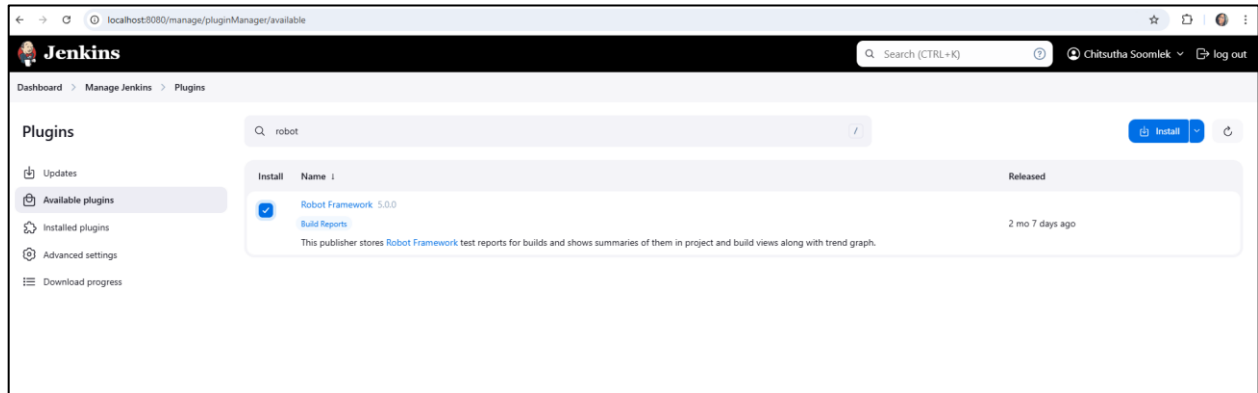
9. เลือก Manage Jenkins แล้วไปที่เมนู Plugins



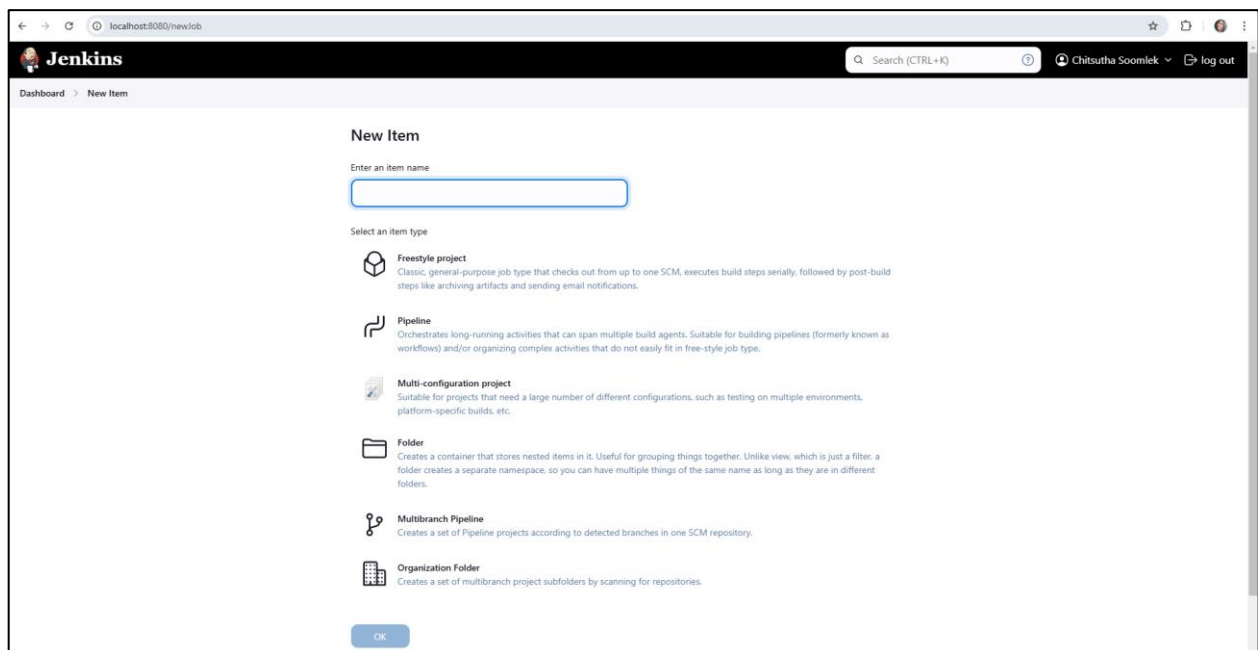
Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

10. ไปที่เมนู Available plugins แล้วเลือกติดตั้ง Robotframework เพิ่มเติม



11. กลับไปที่หน้า Dashboard แล้วสร้าง Pipeline อย่างง่าย โดยกำหนด New item เป็น Freestyle project และตั้งชื่อเป็น UAT



12. นำไฟล์ .robot ที่ทำให้แบบฝึกปฏิบัติที่ 7 (Lab#7) ไปไว้บน Repository ของนักศึกษา จากนั้นตั้งค่าที่จำเป็นในหน้านี้ทั้งหมด ดังนี้

Description: Lab 8.5

GitHub project: กดเลือก แล้วใส่ Project URL เป็น repository ที่เก็บโค้ด .robot (ดูขั้นตอนที่ 12)

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

Build Trigger: เลือกแบบ Build periodically แล้วกำหนดให้ build ทุก 15 นาที

Build Steps: เลือก Execute shell แล้วใส่คำสั่งในการรันไฟล์ .robot (หากไฟล์ไม่ได้อยู่ในหน้าแรกของ repository ให้ใส่ Path ไปถึงไฟล์ให้เรียบร้อยด้วย)

[Check point#14] Capture หน้าจอแสดงการตั้งค่า พร้อมกับตอบคำถามต่อไปนี้

(1) คำสั่งที่ใช้ในการ Execute ไฟล์ .robot ใน Build Steps คือ

Robot UAT-Lab7-002.robot

Post-build action: เพิ่ม Publish Robot Framework test results -> ระบุไดเรกทอรีที่เก็บไฟล์ผลการทดสอบโดย Robot framework ในรูป xml และ html -> ตั้งค่า Threshold เป็น % ของการทดสอบที่ไม่ผ่าน แล้วนับว่าซอฟต์แวร์มีปัญหา -> ตั้งค่า Threshold เป็น % ของการทดสอบที่ผ่านแล้วนับว่าซอฟต์แวร์มีอยู่ในสถานะที่สามารถนำไปใช้งานได้ (เช่น 20, 80)

13. กด Apply และ Save

14. สั่ง Build Now

[Check point#15] Capture หน้าจอแสดงหน้าหลักของ Pipeline และ Console Output

```
=====
UAT-Lab7-002
=====
Validate Destination Field :: ทดสอบการส่งฟอร์มด้วยการไม่ใส่ destin... | FAIL |
No keyword with name 'Open Browser' found.
-----
Validate Empty Email Field :: ทดสอบการส่งฟอร์มด้วยการไม่ใส่ email | FAIL |
No keyword with name 'Open Browser' found.
-----
Validate Invalid Email Field :: ทดสอบการส่งฟอร์มด้วยการใส่ email ใ... | FAIL |
No keyword with name 'Open Browser' found.
-----
Validate Empty Phone Field :: ทดสอบการส่งฟอร์มด้วยการไม่ใส่ phone | FAIL |
No keyword with name 'Open Browser' found.
-----
Validate Invalid Phone Field :: ทดสอบการส่งฟอร์มด้วยการใส่ phone ใ... | FAIL |
No keyword with name 'Open Browser' found.
-----
UAT-Lab7-002 | FAIL |
5 tests, 0 passed, 5 failed
=====
```

Lab Worksheet

ชื่อ-นามสกุล นางสาว กุลนิษฐ์ เตียรวัฒนศิริ รหัสนักศึกษา 653380321-2 Section 4

Dashboard > UAT > #1

Status

</> Changes

Console Output

Edit Build Information

Delete build '#1'

Timings

Git Build Data

Next Build

✖ #1 (Jan 29, 2025, 10:30:02 AM)

Add description

Keep this build forever

Started by user Kullanit Tianwattanasiri

Started 5 min 56 sec ago
Took 5.2 sec

This run spent:

- 39 ms waiting;
- 5.2 sec build duration;
- 5.2 sec total from scheduled to completion.

git

Revision: 8f822c0898463fe0a69ae1f250f890a530f33adc
Repository: <https://github.com/kullanittian/lab8.git>

- refs/remotes/origin/main

</>

No changes.

19