



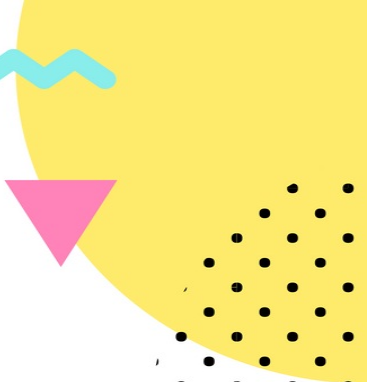






A Newbie's Guide to



# IOT DOCKER SPRING

Einsteiger Guide anhand eines  
Praxisbeispiels



BY ERIC, FELIX, FLORIAN & DANIEL



---

# Inhaltsverzeichnis

## Kapitel I - Deployment

<a href="#">Introduction</a>	1.1
<a href="#">Vorbereitungen</a>	1.2
<a href="#">Quick Start Guide</a>	1.3
<a href="#">Play With Docker</a>	1.3.1
<a href="#">Gitlab Registry</a>	1.3.2
<a href="#">Starten des Systems</a>	1.3.3



A Newbie's Guide to



**IOT**

---



**DOCKER**

---

**SPRING**

---

Einsteiger Guide anhand eines  
Praxisbeispiels



**BY ERIC, FELIX, FLORIAN & DANIEL**





## Docker

Für das Deployment wird Docker genutzt. Stellen sie sicher, dass sowohl docker als auch docker compose auf ihrem Rechner zur Verfügung steht.

- <https://docs.docker.com/compose/install/>
- <https://www.docker.com/>

## Git

Sie benötigen ebenfalls git oder die Gitbash auf ihrem Rechner.

<https://git-scm.com/download/win>

Wechseln Sie nun mit dem Befehl „cd iotlabservices“ in das iot Verzeichnis und starten Sie die Dienste mit dem Befehl „docker-compose up“. Dies dauert wenige Minuten und mit einem Klick auf die oben auftauchenden Ports wie Zum Beispiel „50000“ öffnet sich folgendes Fenster.

## Test Deployment

Als Test nutzen wir *Play with Docker*. Im Produktivsystem, sorgen sie bitte dafür, dass ein Rechner zur Verfügung steht, welcher an das Netz der Fabrik angeschlossen ist.

## Play With Docker

Navigieren Sie auf die Seite:

- <https://labs.play-with-docker.com>

Klicken sie auf „Login“ und wählen „docker“ aus. In dem darauf sich öffnenden Anmeldefenster könne sie sich mit einem bereits bestehenden Account anmelden oder einen neuen Account registrieren. Anschließend können sie mit einem weiteren Klick auf „Start“ direkt loslegen.





## Create Session

In der folgenden Oberfläche, die sich geöffnet hat, müssen Sie auf „ADD NEW INSTANCE“ klicken um starten zu können. Folgendes Oberfläche sollte jetzt erscheinen.

03:49:50

CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.33

node1

bcv51dga\_bcv55toabk8g00abht1g

IP

192.168.0.33


Memory

0.91% (36.41MiB / 3.906GiB)

SSH

ssh ip172-18-0-17-bcv51dgabk8g00abhsq0@direct.labs.pla 

DELETE

 EDITOR

```
$ #####
#                                     #
#               WARNING!!!!          #
# This is a sandbox environment. Using personal credentials #
# is HIGHLY! discouraged. Any consequences of doing so are #
# completely the user's responsibilites.                     #
# The PWD team.                                              #
#####
[node1] (local) root@192.168.0.33 ~
$
```



## Login auf Gitlab

Die Docker images der Gruppe UC1 liegen auf Gitlab und werden von da aus an den Kunden geliefert.

Zunächst fangen sie mit der Docker Anmeldung und dem folgenden Befehl „docker login registry.gitlab.com“ an. Anschließend wird die Eingabe des Usernamens und des Passworts gefordert.

```
[node1] (local) root@192.168.0.58 ~  
$ docker login registry.gitlab.com  
Username: Dan-Red  
Password:  
Login Succeeded
```

## Klonen des Projektes

Geben Sie anschließend folgenden Befehl ein „git clone <https://gitlab.com/ericbrandt91/BPA.git>“. Sie werden erneut dem Usernamen und Passwort gefragt um anschließend auf das Repository zugreifen zu können.

```
[node1] (local) root@192.168.0.58 ~
$ git clone https://gitlab.com/ericbrandt91/BPA.git
Cloning into 'BPA'...
Username for 'https://gitlab.com': Dan-Red
Password for 'https://Dan-Red@gitlab.com':
remote: Counting objects: 1815, done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 1815 (delta 0), reused 0 (delta 0)
Receiving objects: 100% (1815/1815), 90.49 MiB | 38.85 MiB/s, done.
Resolving deltas: 100% (521/521), done.
```

Benutzen sie jetzt den Befehl „cd BPA/“ um in das BPA Verzeichnis zu gelangen und anschließend den Befehl „git checkout docker“ um auf den Docker Branch zu wechseln.

```
[node1] (local) root@192.168.0.58 ~/BPA
$ git checkout docker
Branch 'docker' set up to track remote branch 'docker' from 'origin'.
Switched to a new branch 'docker'
```

## Start des Projektes

Wechseln Sie nun mit dem Befehl „cd iotlabservices“ in das iot Verzeichnis und starten Sie die Dienste mit dem Befehl „docker-compose up“. Dies dauert wenige Minuten und mit einem Klick auf die oben auftauchenden Ports wie Zum Beispiel „50000“ öffnet sich folgendes Fenster.

The screenshot shows the Spring Boot Admin interface. At the top, there are navigation links: Wallboard, Applications, Journal, About, and Logout. Below the navigation bar, there are three summary cards: APPLICATIONS (4), INSTANCES (4), and STATUS (all up). The main content area is titled 'UP' and contains a table with four rows, each representing an application. Each row has a green checkmark, a status of '5m', the application name, and its URL.

UP	APPLICATIONS	INSTANCES	STATUS
✓	DecisionService	4	all up
5m	http://a9a364c3e279:3004/		
✓	LEDApp		
5m	http://a9a364c3e279:3005/		
✓	OPCUApp		
5m	http://a9a364c3e279:3006/		
✓	RTLSJava		
5m	http://a9a364c3e279:3002/		