

# Konfigurerbart system for modenhetsanalyser av IT-sikkerhet


Saphêneia versjon 1.0.0

## Quick Start Guide:

### Systemkrav

- Docker desktop v 4.29.0 eller høyere
- Minstekrav for å kjøre docker: <https://docs.docker.com/desktop/install/windows-install/>.
- Ytterligere 3 GB for hele installasjonen. Docker kan lastes ned fra denne lenken: <https://www.docker.com/products/docker-desktop/>

For å installere programvaren: Lokal versjon med bruk av Docker / VM hos skyleverandør

1. Installer Docker desktop
2. Start Docker desktop
3. Last ned hele mappen dunamis ifra dette repositoriet og lagre på C:
4. Tast  (Windowstast) + R og tast "cmd" + ok / Eller bruk søkefeltet og søk etter "Command Prompt" eller "Kommandolinje"
5. skriv "cd c:/dunamis/custom\_image/dunamis\_spring"
6. skriv "docker load -i c:/dunamis/custom\_image/dunamis\_spring/dunamis\_spring.tar" (ligger også en .txt fil i mappen med kommandoen)
7. skriv "docker image ls"

Du skal nå kunne se følgende:

```
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vekselstrom>cd c:/dunamis/custom_image/dunamis_spring

c:\dunamis\custom_image\dunamis_spring>docker load -i c:/dunamis/custom_image/dunamis_spring/dunamis_spring.tar
d310e774110a: Loading layer [=====] 29.12MB/29.12MB
0ac7ecf8a41c: Loading layer [=====] 4.008MB/4.008MB
659a8c4ba776: Loading layer [=====] 204.3MB/204.3MB
d39504f43454: Loading layer [=====] 56.06MB/56.06MB
6de8fd972fce: Loading layer [=====] 32.55MB/32.55MB
Loaded image: dunamis_spring:latest

c:\dunamis\custom_image\dunamis_spring>docker image ls
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
dunamis_spring      latest         d75a3cfcf296   30 minutes ago  639MB

c:\dunamis\custom_image\dunamis_spring>
```

8. skriv "cd c:/dunamis/custom\_image/dunamis\_react"
9. skriv "docker load -i c:/dunamis/custom\_image/dunamis\_react/dunamis\_react.tar" (ligger også en .txt fil i mappen med kommandoen)
10. skriv "docker image ls"

Du skal nå kunne se følgende:

```
Command Prompt

c:\dunamis\custom_image\dunamis_spring>cd c:/dunamis/custom_image/dunamis_react

c:\dunamis\custom_image\dunamis_react>docker load -i c:/dunamis/custom_image/dunamis_react/dunamis_react.tar
97e25762aa96: Loading layer [=====] 93B/93B
792f9022fb18: Loading layer [=====] 551B/551B
06d7eld4d8c1: Loading layer [=====] 1.271kB/1.271kB
25aa44aec4f7: Loading layer [=====] 3.087MB/3.087MB
7538c699b5b5: Loading layer [=====] 4.511MB/4.511MB
913feccfc05a: Loading layer [=====] 621.7kB/621.7kB
Loaded image: dunamis_react:latest

c:\dunamis\custom_image\dunamis_react>docker image ls
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
dunamis_spring      latest         d75a3cfcf296   33 minutes ago  639MB
dunamis_react       latest         eda79df2ca55   33 minutes ago  172MB

c:\dunamis\custom_image\dunamis_react>
```

11. skriv "cd c:/dunamis"
12. skriv "docker compose up -d"

Du skal nå kunne se følgende:

```
Command Prompt

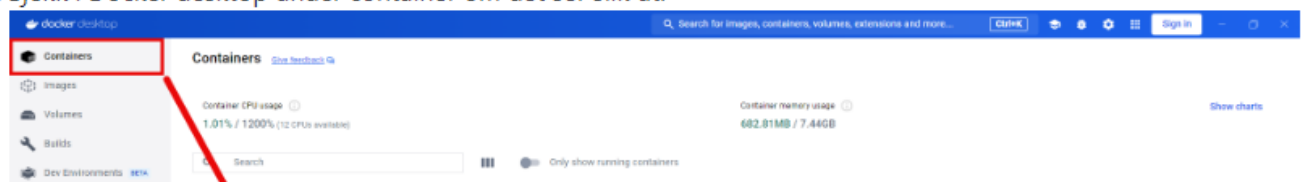
c:\dunamis\custom_image\dunamis_react>cd c:/dunamis

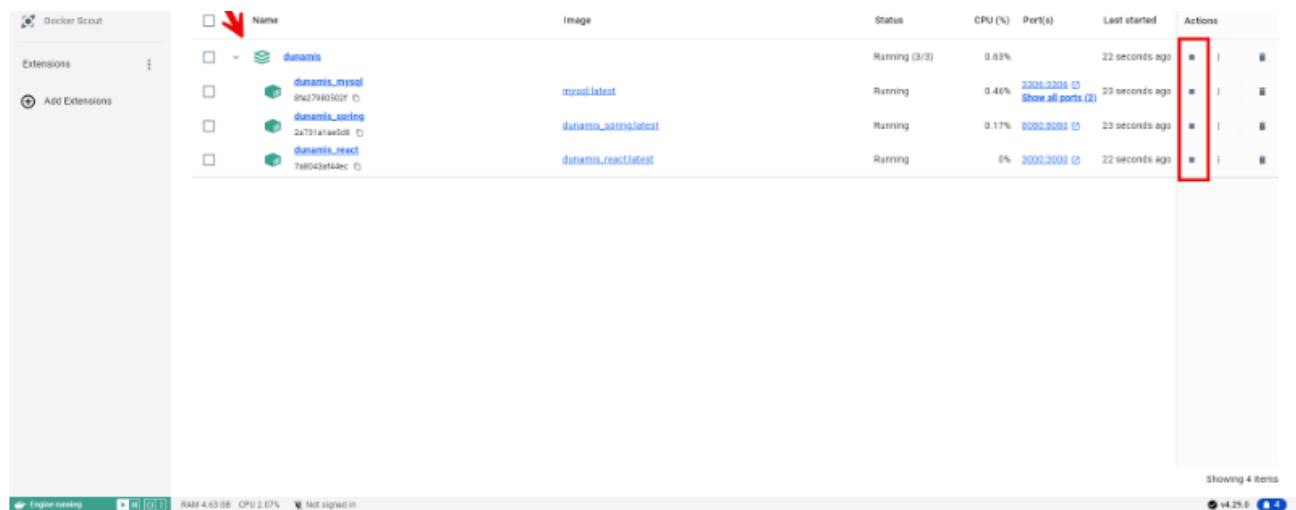
c:\dunamis>docker compose up -d
time="2024-04-14T14:40:32+02:00" level=warning msg="c:\\dunamis\\docker-compose.yml: 'version' is obsolete"
[+] Running 11/11
  ✓ dunamis_mysql Pulled                                39.4s
  ✓ 2ba873cb070a Pull complete                          4.9s
  ✓ dd1a4da808dd Pull complete                          0.8s
  ✓ 3292fb4adf41 Pull complete                          0.9s
  ✓ 3811c45068cc Pull complete                          1.7s
  ✓ e13320244c05 Pull complete                          1.4s
  ✓ 6a34d702f281 Pull complete                          1.9s
  ✓ de90f4481477 Pull complete                          9.0s
  ✓ d575200ae375 Pull complete                          2.4s
  ✓ aea400be5707 Pull complete                          9.3s
  ✓ 38c930606a4f Pull complete                          5.3s
[+] Running 4/4
  ✓ Network dunamis_default Created                    0.1s
  ✓ Container dunamis_mysql Started                     0.7s
  ✓ Container dunamis_spring Started                    0.1s
  ✓ Container dunamis_react Started                     0.1s

c:\dunamis>
```

13. Etter kommandoen har kjørt ferdig så vil du kunne nå systemet umiddelbart, men ved første gangs installasjon kan det ta opptil 4 minutter før databasen har blitt ferdig installert. Databasen inneholder en eksempel bedrift. Denne vil være synlig når databasen er ferdig installert.

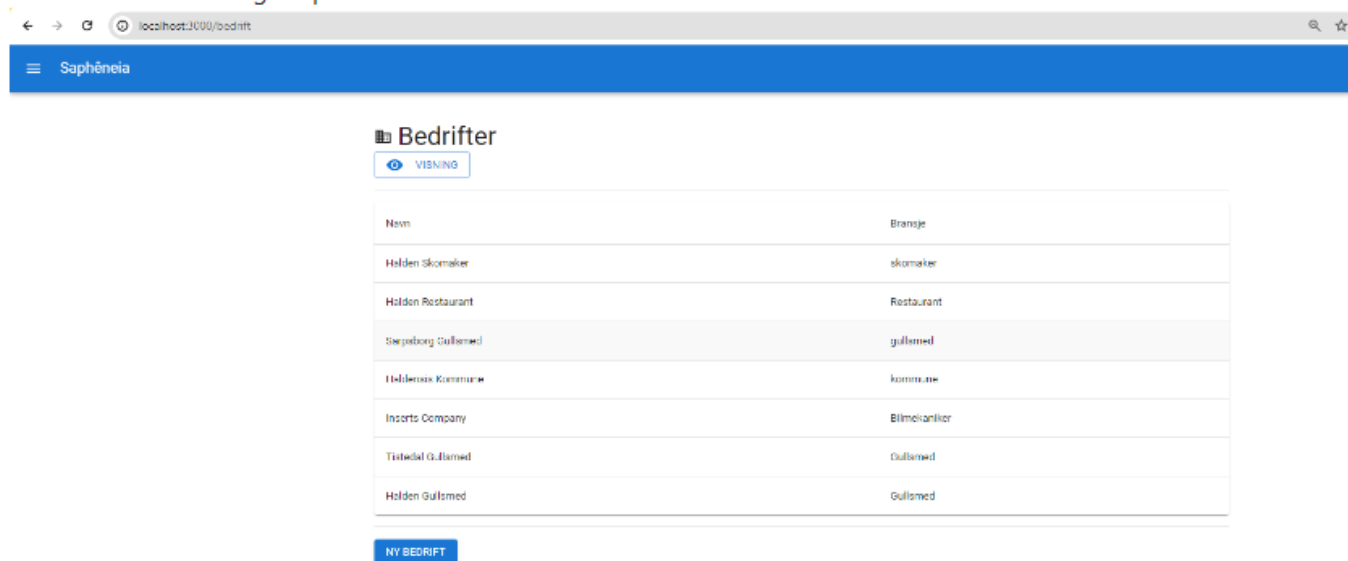
14. Sjekk i Docker desktop under container om det ser slik ut:





15. Åpne en webleser og skriv inn adressen "<http://localhost:3000/>"

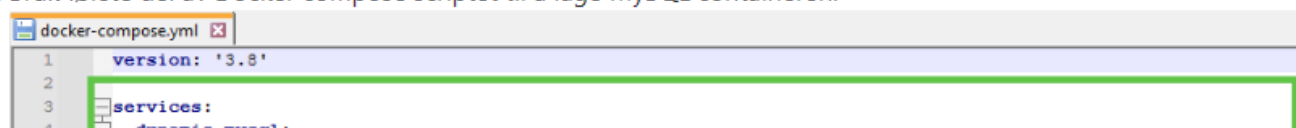
Du skal nå noe som ligner på dette:



## For å installere programvaren: Separate docker containere hos en skyplattform

NB! Systemet er ikke testet eller utviklet til å kunne direkte hostes på en skyplattform i separate Docker containere. Systemet er designet til å bli installert via Docker på en og samme plattform/enhet/VM/PC.

1. Last opp filen `dunamis_react.tar` til en container tjeneste hos en skyplattform (dette er et custom docker image) (filen genereres ved bruk av GitHub workflow 'Build Docker Custom Image dunamis.react (Node)' eller ligger i mappen `/dunamis/custom_image/dunamis_react` )
2. Start containeren
3. Last opp filen `dunamis_spring.tar` til en container tjeneste hos en skyplattform (dette er et custom docker image) (filen genereres ved bruk av GitHub workflow 'Build Docker Custom Image dunamis.spring (Java)' eller ligger i mappen `/dunamis/custom_image/dunamis_spring` )
4. Star containeren
5. Bruk første del av Docker compose scriptet til å lage MySQL containeren.



```

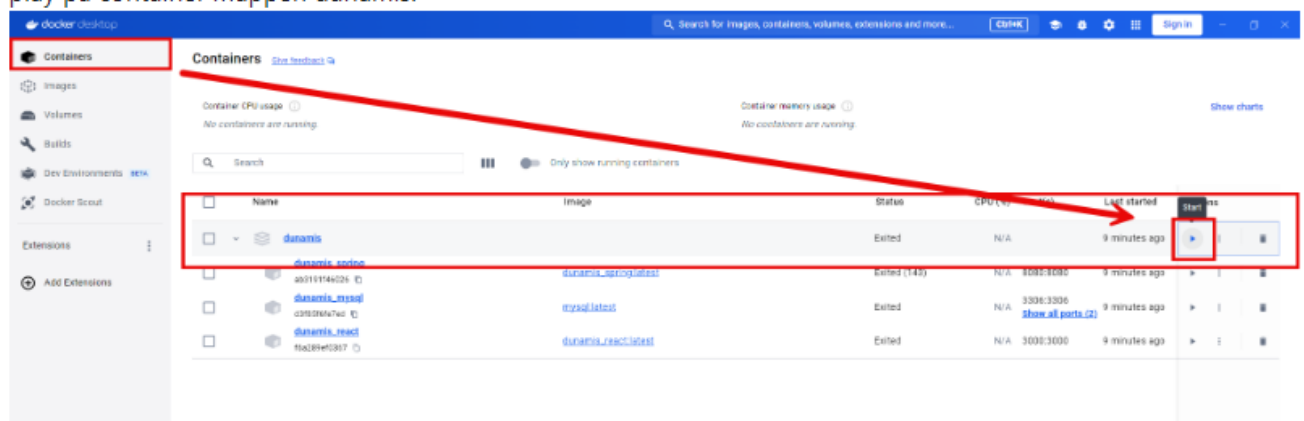
5     image: mysql:latest
6     restart: always
7     container_name: dunamis_mysql
8     environment:
9         MYSQL_ROOT_PASSWORD: Xj52N7JwBfqANM6E4QgwXGn63bsW3Pje
10        MYSQL_DATABASE: dunamis #NB skal skrives to steder
11        MYSQL_USER: spring_user #NB skal skrives to steder
12        MYSQL_PASSWORD: IPS@ie6m3!Iqs2 #NB skal skrives to steder
13        MYSQL_BACKUP_USER: backup_user #NB skal skrives to steder
14        MYSQL_BACKUP_PASSWORD: rkVmxhhX3ul^wD7pH&7W #NB skal skrives to steder
15    ports:
16        - "3306:3306"
17        - "33060:33060"
18    volumes:
19        - C:/dunamis/dunamis_mysql_database:/var/lib/mysql #Endre "C:/Dunamis_Assessment" om det ønskes annet
20        - ./dunamis_dump.sql:/docker-entrypoint-initdb.d/dunamis_dump.sql # database dump sql
21        - ./init_backup_user.sql:/docker-entrypoint-initdb.d/init_backup_user.sql # Custom script å lage backup
22
23    dunamis_spring:
24        image: dunamis_spring:latest
25        restart: always
26        container_name: dunamis_spring
27        build:
28            context: ./dunamis_spring
29        ports:
30            - "8080:8080"
31        depends_on:
32            - dunamis_mysql
33        environment:
34            SPRING_DATASOURCE_URL: jdbc:mysql://dunamis_mysql:3306/dunamis # 'dunamis_mysql' er dynamisk variabel
35            SPRING_DATASOURCE_USERNAME: spring_user
36            SPRING_DATASOURCE_PASSWORD: IPS@ie6m3!Iqs2
37            BACKUP_HOST: dunamis_mysql # dynamisk variabel skal sende IPv4 som $ENV
38            BACKUP_PORT: 3306
39            BACKUP_DATABASE: dunamis
40            BACKUP_USERNAME: backup_user
41            BACKUP_PASSWORD: rkVmxhhX3ul^wD7pH&7W
42
43    dunamis_react:
44        image: dunamis_react:latest
45        restart: always
46        container_name: dunamis_react
47        build:
48            context: ./
49        ports:
50            - "3000:3000"
51        depends_on:
52            - dunamis_spring

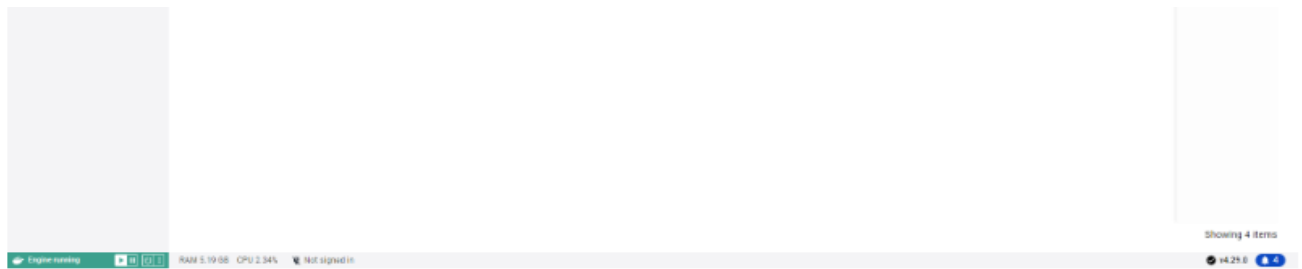
```

6. Lag sikker kommunikasjon mellom container tjenestene på den valgte skyplattformen.
7. dunamis\_react trenger å kommunisere med dunamis\_spring på port 8080, dunamis\_spring trenger å kommunisere dunamis\_mysql på port 3306. Dette bør gjøres etter beste praksis og videre instruksers på hvordan dette gjøres sikkert er utenfor scopen til denne oppgaven.

## For å starte programvaren: Lokal versjon med bruk av Docker / VM hos skyleverandør

1. Start Docker desktop (dersom ikke Docker desktop starter av seg selv når Windows starter)
2. Containeren skal starte av seg selv, i korrekt rekkefølge. Hvis ikke gå i Docker desktop under containers og trykk play på container mappen dunamis.

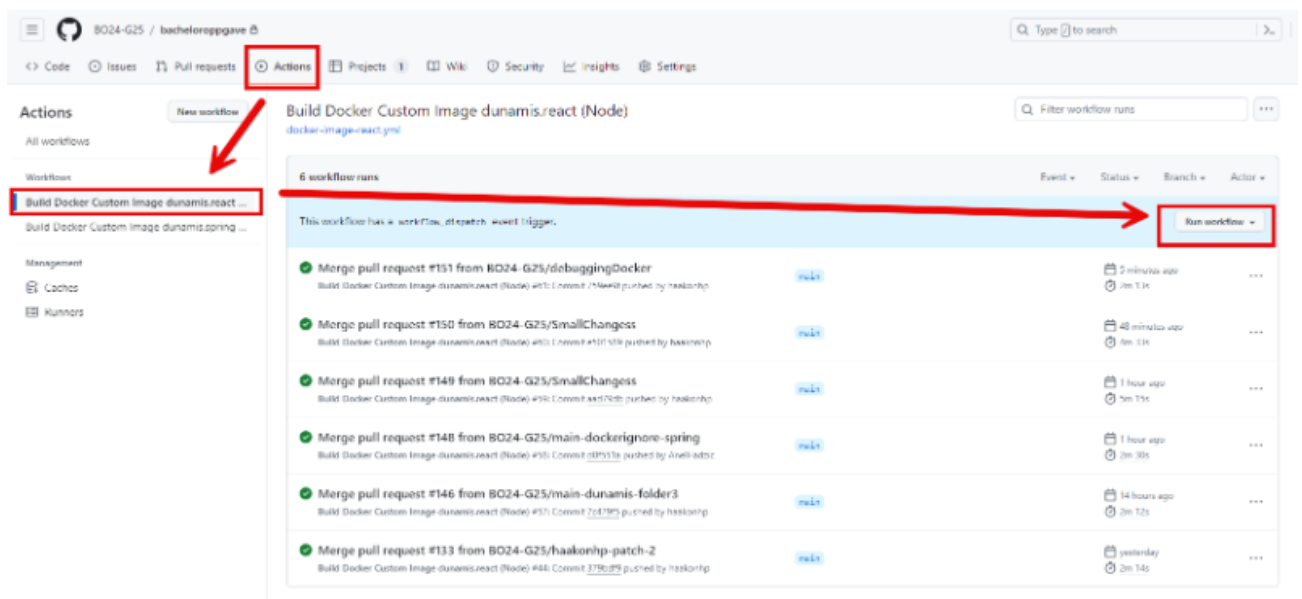




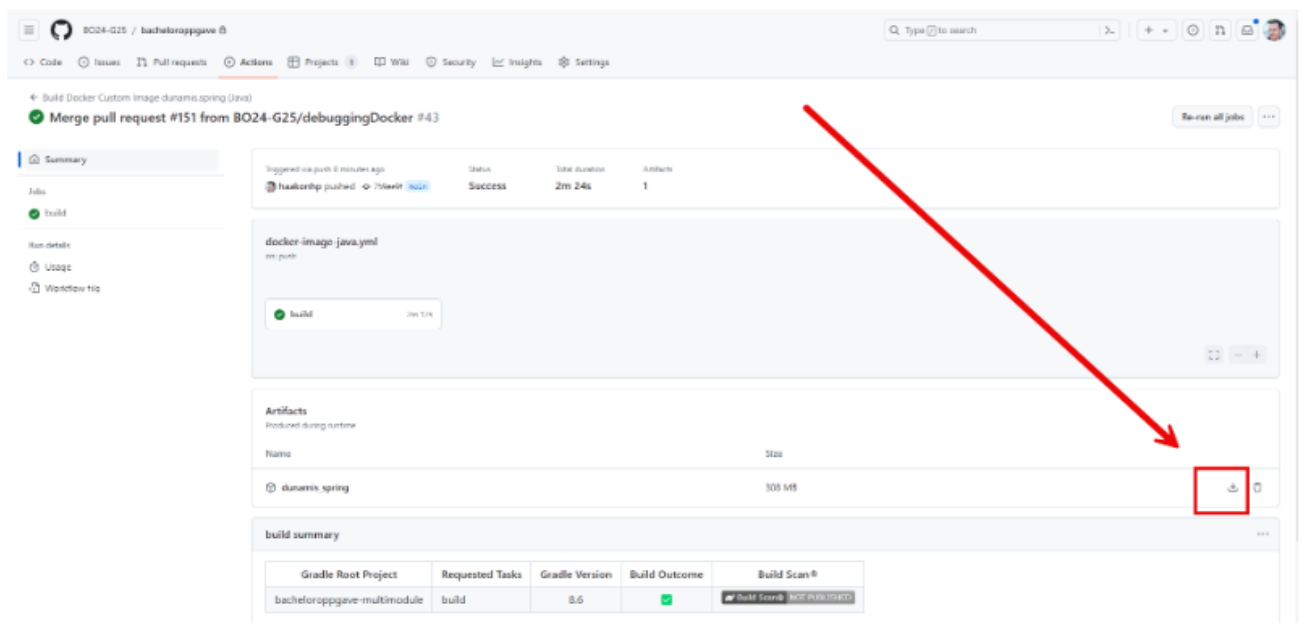
3. Åpne en webleser og skriv inn adressen "<http://localhost:3000/>"

## Lage nye docker images ved videre utvikling av programvaren: I GitHub

1. Last opp repositoriet levert med denne bacheloroppgaven til en GitHub konto.
2. Hver gang kode blir merget til main branchen så vil det genereres nye tar-filer for react og spring delen av programvaren. Alternativt start hver av workflow manuelt ved å gå til menyen Actions og klikke og starte dem manuelt



3. Last ned tar-filene når workflowen er ferdig (grønn).



4. Lagre tar-filene under dem respektive mappene under `c:\dunamis\custom_image\dunamis_*****`
  5. Følg instruksene for å installere programvaren.
- 

#### Lage nye docker images ved videre utvikling av programvaren: I en IDE/Terminal

1. Bruk repositoriet levert med denne bacheloroppgaven.
2. Installer Docker på maskinen du driver utviklingen på.
3. Start Docker.
4. Etter du har utført endringene, åpne root mappen på prosjektet i en terminal.
5. Skriv "docker build -t dunamis\_spring:latest ." (med punktum på slutten) for å lage dunamis\_spring Docker imaget
6. Skriv "docker image ls" for å se at imaget er lagt inn i Docker på maskinen.

Du skal nå noe som ligner på dette:

```
vekselstrom@Hakon_Laptop MINGW64 /c/dunamis/custom_image/dunamis_spring
$ docker image ls
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
dunamis_spring      latest         c97e361251c8   33 minutes ago  639MB

vekselstrom@Hakon_Laptop MINGW64 /c/dunamis/custom_image/dunamis_spring
$ |
```

7. Naviger til mappen frontend åpne den med en terminal.
8. Skriv "docker build -t dunamis\_react:latest ." (med punktum på slutten) for å lage dunamis\_react Docker imaget
9. Skriv "docker image ls" for å se at imaget er lagt inn i Docker på maskinen.

Du skal nå noe som ligner på dette:

```
vekselstrom@Hakon_Laptop MINGW64 /x/Hiof/2023 Bacheloroppgave Java React (main)
$ docker image ls
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
dunamis_react       latest         eeead1cfe296   2 hours ago    172MB

vekselstrom@Hakon_Laptop MINGW64 /x/Hiof/2023 Bacheloroppgave Java React (main)
$ |
```

10. Fortsett på punkt 11. på instruksene for å installere programvaren.
-