

Impact

Run instruction (update)

A) DockerHub Compose Run Instruction (recommend):

1. open terminal and run below command to pull Github repository:
`git clone https://github.com/ZhongshanShi/Impact`
2. run below command in terminal:
`docker-compose up`
then both UI and backend Api should be up and running

check on <http://localhost:3000/> for UI
check on <http://localhost:9090/impact.com/api/healthCheck> for backend health

B) GitHub Run Instruction:

3. open terminal and run below command to pull Github repository:
`git clone https://github.com/ZhongshanShi/Impact`
4. for UI, needs install Node.js (v14.16.0), run below two commands in terminal:
`npm install`
`npm start`
check on <http://localhost:3000/> for UI
5. for Server, needs install (JDK 15, Maven 3.6.3), just run the Springboot application in any IDE like IntelliJ or Eclipse.

check on <http://localhost:9090/impact.com/api/healthCheck> for backend health

C) DockerHub Run Instruction:

You will need to pull down both UI and Backend images and run them on Docker.

1. run below command in terminal to pull backend server image from Docker-hub:
`docker pull jayshidocker/docker-spring-boot:1.1.1`
2. run below command in terminal to pull UI image from Docker-hub:
`docker pull jayshidocker/react-impact:1.1.1`
3. check whether the pull is completed, run below command:

docker images

you should see below two images in your docker:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
jayshidocker/docker-spring-boot	1.1.1	348dd038d20d	16 minutes ago	823MB
jayshidocker/react-impact	1.1.1	7da80c3744ec	27 minutes ago	484MB

if you see both images, move to the next step.

4. run below command to start backend server (make sure port 9090 is not taken by other apps)

```
docker run -p 9090:9090 jayshidocker/docker-spring-boot:1.1.1
```

then copy below link to a browser to check server is up running:

<http://localhost:9090/impact.com/api/healthCheck>

you should see below response:

```
{  
  "statusCode": 200,  
  "message": "healthCheck successful"  
}
```

5. open a new terminal and run below command to start UI (make sure port 3000 is not taken by other apps)

```
docker run -p 3000:3000 jayshidocker/react-impact:1.1.1
```

6. open a new terminal and run below command to check both containers are running

```
docker ps
```

you should see below two containers in your docker:

CONTAINER ID	IMAGE	COMMAND	CREATED
95845e5d8fdb	jayshidocker/react-impact:1.1.1	"docker-entrypoint.s..."	About a minute a
908f2a006e2f	jayshidocker/docker-spring-boot:1.1.1	"java -jar impact-sp..."	4 minutes ago

7. Both UI and api server should be up now and you can test via below port

<http://localhost:3000/>