

Produktdokumentation

Software Engineering II



Optima Connect

Projekttitle: BonoboBoard

Jakob Hutschenreiter (1419081)

Jiesen Wang (9839152)

Nick Kramer (3122448)

Patrick Küsters (9815596)

Peter Moritz Hinkel (2783930)

DHBW Mannheim

24. März 2022



Änderungshistorie

Revision	Datum	Autor(en)	Beschreibung
1.0	18.03.2022	NK	A: 1, 2, 3

Abkürzungen: Hinzugefügt/Added (A), Änderung/Changed (C), Löschung/Deleted (D)

Inhaltsverzeichnis

1	Motivation und Grundlagen	1
2	Installationsdokumentation	1
2.1	Beziehen des Source-Code	1
2.2	Installation und Start der Docker Container	2
2.3	Darstellung im Webbrowser	3
3	Kurzanleitung	5
3.1	Anmeldung	5





1 Motivation und Grundlagen

Dieses Dokument dient zur Beschreibung der Abläufe, die nötig sind, um das BonoboBoard lokal zu installieren und auszuführen. Auf den Aufbau des Software-Produkts wird hier nicht mehr eingegangen. Bitte ziehen Sie dafür die etwaigen anderen Dokumente heran.

Die nachfolgende Beschreibung wurde auf Basis folgender Abhängigkeiten erstellt:

Software/Library	Version
Docker	20.10.12, Build e91ed57
Docker Compose	2.2.3
Docker Desktop	4.5.1 (74721)

Bitte stellen Sie sicher, dass Sie die genannten Voraussetzungen erfüllen, anderweitig kann nicht sichergestellt werden, dass die Installation auf Ihrem System ordnungsgemäß funktioniert.

Wenn Sie das Produkt lediglich nutzen möchten, können Sie die Installationsdokumentation überspringen und direkt zu Abschnitt 3 wechseln. Kein Nutzer muss das BonoboBoard lokal installieren, die aktuelle Version kann immer unter <https://bonoboboard.de/> gefunden und genutzt werden. Falls Sie das Produkt weiterentwickeln möchten oder eine lokale Installation anstreben, ist mit Abschnitt 2 fortzufahren.

2 Installationsdokumentation

Durch die Nutzung von Containern unter Docker lässt sich die Installation in einigen wenigen Schritten behandeln.

2.1 Beziehen des Source-Code

Der Source-Code wird auf GitHub gepflegt¹. Da es sich um ein öffentliches Repository handelt, kann der Code ohne weitere Authentifizierung lokal geklont werden. Für eine detaillierte Anleitung des Klon-Prozesses wird auf die offizielle Dokumentation von GitHub verwiesen².

Ist das Klonen abgeschlossen, sollte folgende Struktur auf der ersten Ebene des Projekts zu finden sein:

¹<https://github.com/Software-Engineering-DHBW/BonoboBoard>

²<https://docs.github.com/en/repositories/creating-and-managing-repositories/cloning-a-repository>

Name	Änderungsdatum	Typ
.git	13.03.2022 13:41	Dateiordner
.github	13.03.2022 12:22	Dateiordner
bonobo-board	13.03.2022 13:40	Dateiordner
documents	18.03.2022 09:51	Dateiordner
server-deployment	13.03.2022 12:22	Dateiordner
sphinx_src	13.03.2022 13:39	Dateiordner
.gitignore	13.03.2022 12:22	Textdokument
README.md	13.03.2022 12:22	Markdown-Quelld...

Abbildung 1: Ordnerstruktur des heruntergeladenen Projekts

2.2 Installation und Start der Docker Container

Dieser Abschnitt beschreibt den Ablauf, wie die Docker-Umgebung aufgebaut und gestartet wird. Dazu muss in den Ordner *bonobo-board* (siehe dritter Ordner von oben in Abbildung 2) gewechselt werden. Darin befindet sich eine Datei mit dem Namen *docker-compose.yml* und ein Dockerfile. Sollten diese nicht vorhanden sein, ist entweder nicht der richtige Ordner ausgewählt oder beim Herunterladen der Dateien sind Fehler aufgetreten. Hier eine Abbildung des Verzeichnisses zum Vergleich:

BonoboBoard > bonobo-board >

🔍 "bonobo-board" durchsuchen




















Name	Änderungsdatum	Typ	Größe
 .idea	14.03.2022 00:49	Dateiordner	
 apps	13.03.2022 12:31	Dateiordner	
 core	13.03.2022 12:31	Dateiordner	
 modules	13.03.2022 13:33	Dateiordner	
 nginx	13.03.2022 12:22	Dateiordner	
 staticfiles	13.03.2022 12:22	Dateiordner	
 .env	13.03.2022 12:22	ENV-Datei	1 KB
 .gitignore	13.03.2022 12:22	Textdokument	1 KB
 CHANGELOG.md	13.03.2022 12:22	Markdown-Quelld...	2 KB
 docker-compose.yml	13.03.2022 12:22	Yaml-Quelldatei	1 KB
 Dockerfile	13.03.2022 12:22	Datei	1 KB
 gunicorn-cfg.py	13.03.2022 12:22	Python-Quelldatei	1 KB
 LICENSE.md	13.03.2022 12:22	Markdown-Quelld...	1 KB
 manage.py	13.03.2022 12:22	Python-Quelldatei	1 KB
 package.json	13.03.2022 12:22	JSON-Quelldatei	1 KB
 Procfile	13.03.2022 12:22	Datei	1 KB
 README.md	13.03.2022 12:22	Markdown-Quelld...	9 KB
 requirements.txt	13.03.2022 12:22	Textdokument	1 KB
 runtime.txt	13.03.2022 12:22	Textdokument	1 KB

Abbildung 2: Struktur in *bonobo-board*

Nun muss ein Terminal in diesem Ordner gestartet werden.

Im Terminal ist der Befehl `docker-compose up` auszuführen. Alternativ kann der Befehl



`docker-compose up -d` ausgeführt werden, wenn die Kommandozeile nach dem ausführen des Befehls nicht blockiert sein soll. In diesem Fall laufen die gestarteten Container im Hintergrund weiter. Abbildung 3 zeigt beispielhaft die Ausgabe der Befehle in der Kommandozeile.

Bei Fehlermeldungen ist sicherzustellen, dass der `docker-daemon` läuft (Windows: läuft Docker-Desktop?) oder alle Voraussetzungen (siehe 1) erfüllt sind.

```
Eingabeaufforderung - docker-  
C:\Users\NickK\git\BonoboBoard>cd bonobo-board  
C:\Users\NickK\git\BonoboBoard\bonobo-board>docker-compose up  
[+] Running 7/7  
- nginx Pulled 25.9s  
- ae13dd578326 Pull complete 21.4s  
- 6c0ee9353e13 Pull complete 22.7s  
- dca7733b187e Pull complete 22.7s  
- 352e5a6cac26 Pull complete 22.8s  
- 9eaf108767c7 Pull complete 22.9s  
- be0c016df0be Pull complete 22.9s  
[+] Building 66.7s (14/14) FINISHED  
=> [internal] load build definition from Dockerfile 0.0s  
=> => transferring dockerfile: 922B 0.0s  
=> [internal] load .dockerignore 0.0s  
=> => transferring context: 2B 0.0s  
=> [internal] load metadata for docker.io/library/python:3.9 2.0s  
=> [1/9] FROM docker.io/library/python:3.9@sha256:daa3d02ee3969ec6dfba0b61d1fe9bb6da43fc975bc2e27bad10a3aeca06eca9 8.8s  
=> => resolve docker.io/library/python:3.9@sha256:daa3d02ee3969ec6dfba0b61d1fe9bb6da43fc975bc2e27bad10a3aeca06eca9 0.0s  
=> => sha256:b2da6b57d1988692bcf1c97c8f85679c6a5f6787b253821fe2f2059a0b473346 8.54kB / 8.54kB 0.0s  
=> => sha256:daa3d02ee3969ec6dfba0b61d1fe9bb6da43fc975bc2e27bad10a3aeca06eca9 2.35kB / 2.35kB 0.0s  
=> => sha256:6f591944229033d3920918e8bf04283078d33d4fa5310cabb53a7294c45aa7ab 2.22kB / 2.22kB 0.0s  
=> => sha256:84a26a2849e127cec54f32f2474ff35efe05da4e5db1bccaff0e53a5f0b26e6f 238B / 238B 0.7s  
=> => sha256:a755ea659b0894999e0b6708cfff7317f2fc1bc45ff3a50eaf524a6a454cd3990 2.87MB / 2.87MB 2.7s  
=> => sha256:7d3e25cf7bd07ea20c813ddcfe86abf839e98a61b64f542671b21d6f973e9f3e 18.30MB / 18.30MB 7.2s  
=> => extracting sha256:7d3e25cf7bd07ea20c813ddcfe86abf839e98a61b64f542671b21d6f973e9f3e 1.0s  
=> => extracting sha256:84a26a2849e127cec54f32f2474ff35efe05da4e5db1bccaff0e53a5f0b26e6f 0.0s  
=> => extracting sha256:a755ea659b0894999e0b6708cfff7317f2fc1bc45ff3a50eaf524a6a454cd3990 0.3s  
=> [internal] load build context 0.6s  
=> => transferring context: 6.73MB 0.5s  
=> [2/9] WORKDIR /bonobo-board 0.7s  
=> [3/9] COPY . . 0.1s  
=> [4/9] RUN rm -rf CHANGELOG.md docker-compose.yml Dockerfile LICENSE.md nginx/ README.md 0.4s  
=> [5/9] RUN pip install --upgrade pip 3.0s  
=> [6/9] RUN pip install -r requirements.txt 12.0s  
=> [7/9] RUN pip install -r ./modules/dhbw/requirements.txt 33.5s  
=> [8/9] RUN python manage.py migrate 2.7s  
=> [9/9] RUN python manage.py collectstatic --no-input 1.6s  
=> exporting to image 1.6s  
=> => exporting layers 1.6s  
=> => writing image sha256:062c705c32f53fb95d4280049440f72b822a604da56e5410885f43bac168e582 0.0s  
=> => naming to docker.io/library/bonobo-board_appseed-app 0.0s  
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them  
[+] Running 2/2  
- Container appseed_app Created 1.1s  
- Container nginx Created 0.1s  
Attaching to appseed_app, nginx
```

Abbildung 3: docker-compose in der Kommandozeile

2.3 Darstellung im Webbrowser

Da die Container nun gestartet sind, ist zu prüfen ob alles funktioniert und die Website lokal erreichbar ist. Dafür ist ein Webbrowser nach Wahl zu öffnen und `http://localhost:80/` einzugeben. Es sollte sich die Website, wie in Abbildung 4 dargestellt, öffnen.

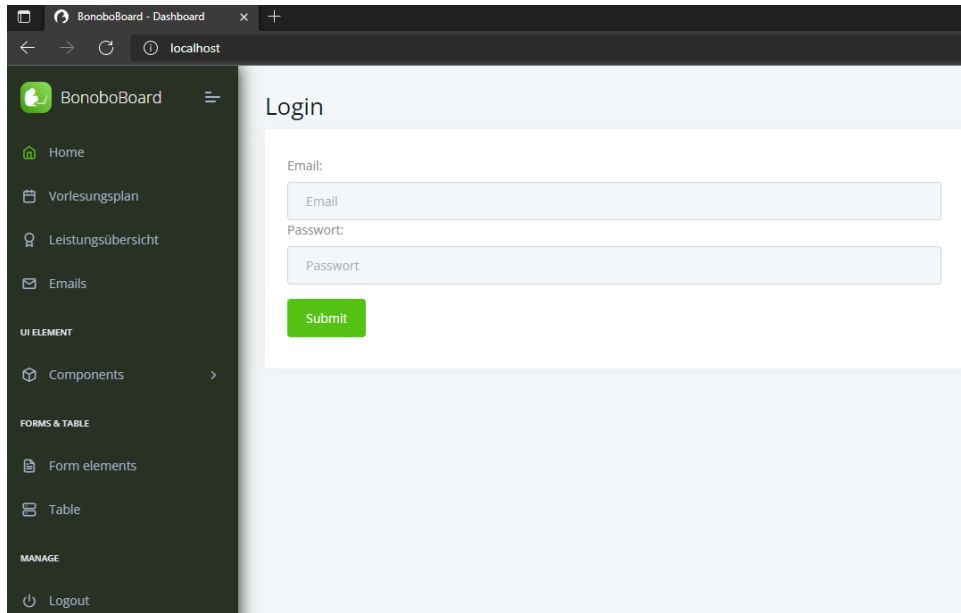


Abbildung 4: Darstellung im Browser

Damit ist die lokale Installation des BonoboBoard abgeschlossen.

Haben alle Schritte vorab funktioniert und die Darstellung funktioniert trotzdem nicht, ist sicherzustellen, dass kein anderer Dienst Port 80 belegt. Unter Windows kann dies mit Hilfe der Kommandozeile und des Taskmanagers überprüft werden. Dazu ist der Befehl `netstat -ano -t tcp` auszuführen und die Zeile zu lokalisieren, in der die lokale Adresse `0.0.0.0:80` durch die remote Adresse `0.0.0.0:0` abgehört wird. Die PID am Ende dieser Zeile kann im Task-Manager auf eine Anwendung zurückgeführt werden. In Abbildung 5 ist dies dargestellt. In diesem Fall wird die Website ordnungsgemäß dargestellt. Benutzt ein anderer Prozess (außer Docker-Desktop) diesen Port, ist dieser zu terminieren.

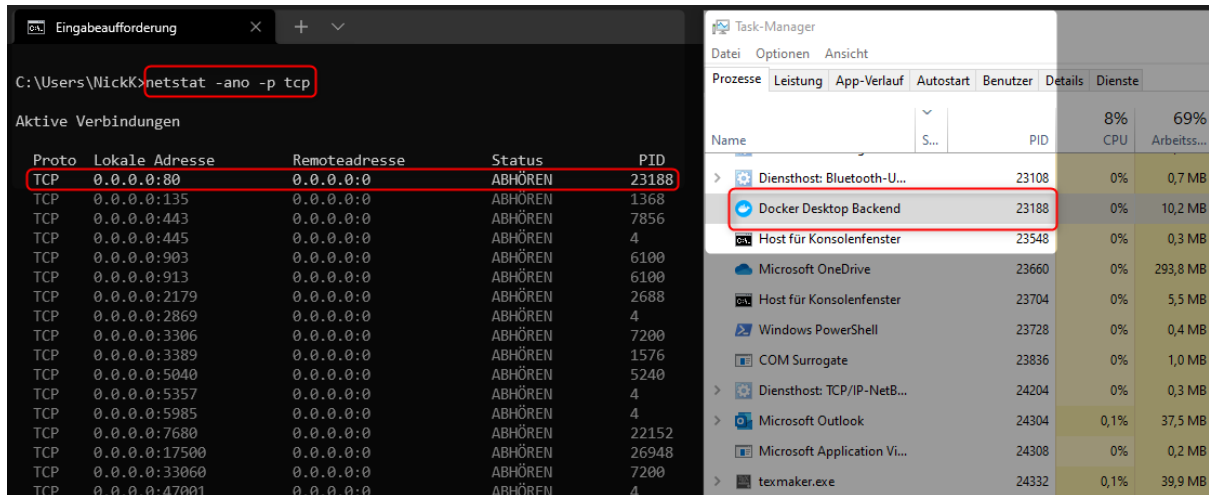


Abbildung 5: Überprüfung des lokalen Port 80

3 Kurzanleitung

Herzlich willkommen beim BonoboBoard! Danke, dass Sie sich für ein Produkt von Optima Connect entschieden haben.

In dieser Kurzanleitung werden wir sie mit den grundlegenden Funktionen Ihres persönlichen DHBW-Dashboards vertraut machen, sodass Sie bereits in Kürze Ihren Workflow mit BonoboBoard optimieren können.

3.1 Anmeldung

Um das BonobobBoard benutzen zu können, gehen sie im Webbrowser Ihrer Wahl auf <https://bonoboboard.de> oder - falls Sie eine lokale Installation verwenden - auf <http://localhost:80/>. Von dort aus gelangen Sie zum Login (siehe Abb. 6).

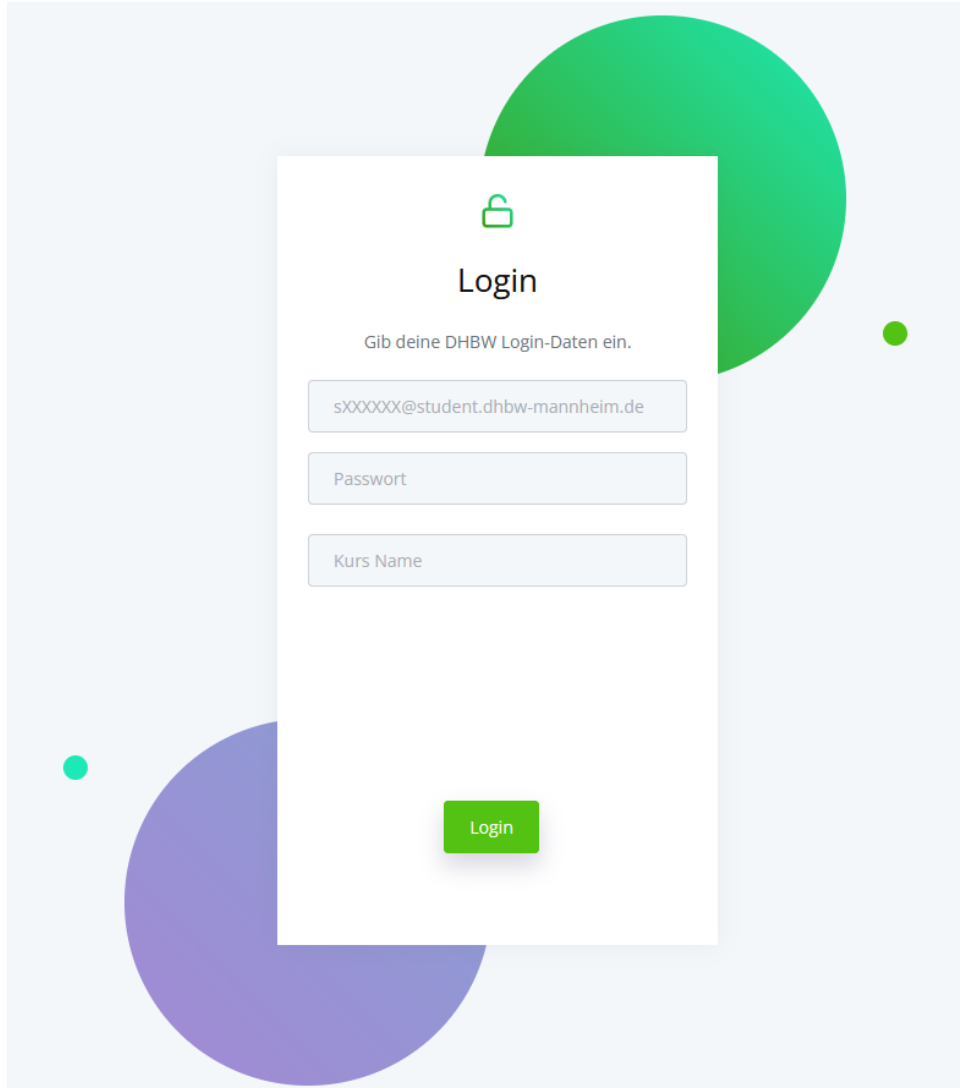
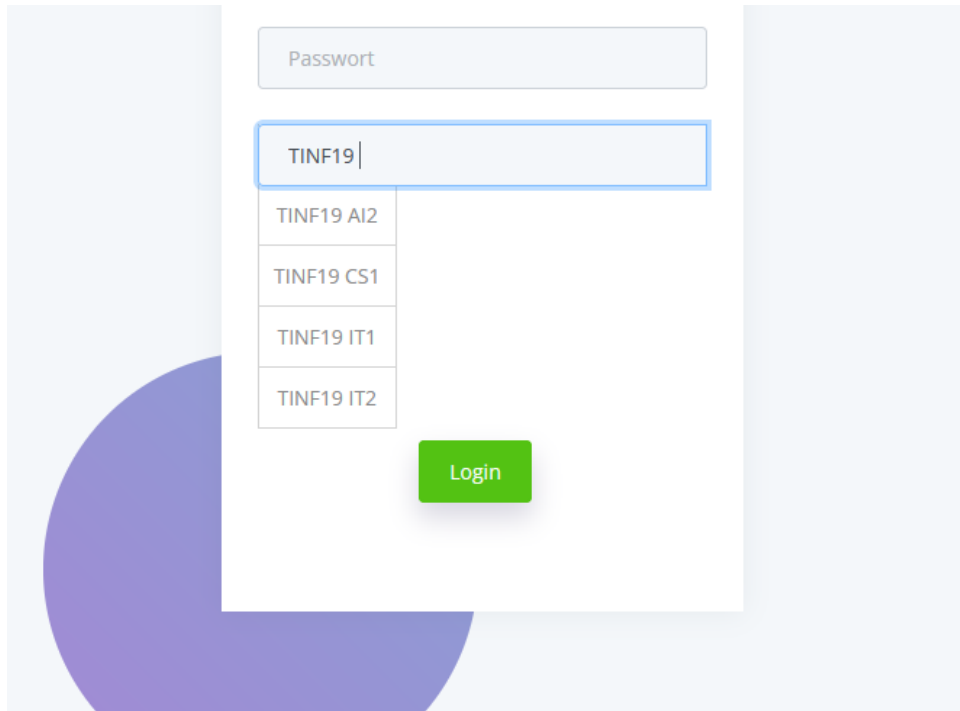


Abbildung 6: BonoboBoard - Login

Geben Sie nun dort Ihre DHBW-Mailadresse (s-Adresse) und Ihr Passwort, dass sie für Ihren DHBW-Account verwenden, ein. Keine Sorge - Ihr Passwort wird weder lokal noch auf unserem Server gespeichert. Auch nicht als Hash. Wir verwenden es lediglich, um eine Verbindung, mit den DHBW-Services aufzubauen. Wird diese Verbindung geschlossen, müssen sie sich allerdings neu anmelden.

Anschließend geben Sie noch Ihre Kursbezeichnung an. Beispielweise »TINF19 IT2«. Achten Sie bitte auf das Leerzeichen zwischen Semester- und Kursbezeichnung. Oder verwenden sie die Autovervollständigung (siehe Abb. 7).



The image shows a login interface for BonoboBoard. It features a central white login box on a light blue background with a large purple 'C' shape. The login box contains a 'Passwort' (password) field, a username field with 'TINF19' entered, a dropdown menu with four options: 'TINF19 AI2', 'TINF19 CS1', 'TINF19 IT1', and 'TINF19 IT2', and a green 'Login' button.

Passwort
TINF19
TINF19 AI2
TINF19 CS1
TINF19 IT1
TINF19 IT2

Login

Abbildung 7: BonoboBoard - Kusrauswahl