

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

28. März 2022



Änderungshistorie

Revision	Datum	Autor(en)	Beschreibung
1.0	18.03.2022	NK	A: 1, 2
1.1	24.03.2022	MH	A: 3
1.2	26.03.2022	JH, NK	C: 2

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	4
2.4	Installationsprobleme	4
3	Kurzanleitung	6
3.1	Anmeldung	6
3.2	Home	7
3.3	Seitennavigation	8
3.4	Vorlesungsplan	8
3.5	Leistungsübersicht	10
3.6	E-Mail	11
4	Danksagung	12

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. In diesem befindet sich eine *docker-compose.yml* Datei und zwei weitere Dockerfiles: *Dockerfile.base*, *Dockerfile.django*. 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/PowerShell in diesem Ordner gestartet werden.

Im Terminal gibt es nun mehrere Wege die Container zum Laufen zu bekommen. Der einfachste

Weg stellt das Ausführen der bereitgestellten Skripte dar.

Automatische Installation Windows

Unter Windows muss das PowerShell-Skript namens *build_image.ps1* ausgeführt werden. Der Befehl hierfür lautet `.\build_image.ps1 -a`, welcher die beiden benötigten Docker Images erstellt.

Automatische Installation Linux

Unter **Ubuntu (Linux)** ist das Bash-Skript *build_image* auszuführen. Der Befehl hierfür ist `./build_image -a`, welcher die Docker Images erstellt.

Nach dem Erstellen der Docker Images reicht ein simples `docker-compose up` und die Container werden erstellt. 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.

```

C:\Users\NickK\git\BonoboBoard>cd bonobo-board
C:\Users\NickK\git\BonoboBoard\bonobo-board>docker-compose up
[*] Running 1/1
- nginx Pulled
  - ae13dd578326 Pull complete
  - 6c0ee935e13 Pull complete
  - dca7733b187e Pull complete
  - 352e5a6cac26 Pull complete
  - 9eaf108767c7 Pull complete
  - be0c016df0be Pull complete
[*] Building 66.7s (14/14) FINISHED
-> [internal] load build definition from Dockerfile
-> => transferring dockerfile: 922B
-> [internal] load .dockerignore
-> => transferring context: 2B
-> [internal] load metadata for docker.io/library/python:3.9
-> [1/9] FROM docker.io/library/python:3.9@sha256:daa3482ee3969ec6dfba8b61d1fe9bb6da43fc975bc2e27bad18a3eeca86eca9
-> => resolve docker.io/library/python:3.9@sha256:daa3482ee3969ec6dfba8b61d1fe9bb6da43fc975bc2e27bad18a3eeca86eca9
-> => sha256:b2da6057d1980692bc1c97c8f85679c6a5f6707b253821fe2f2059a0b473346 8.54kB / 8.54kB
-> => sha256:daa3482ee3969ec6dfba8b61d1fe9bb6da43fc975bc2e27bad18a3eeca86eca9 2.35kB / 2.35kB
-> => sha256:6f59194422903d3920918e8bf04283078d33d4fa5310cab53a7294c45a7ab 2.22kB / 2.22kB
-> => sha256:84a26a2849e127cec54f32f2474ff35efe05da4e5db1bccaaff0e53a5f0b26e6f 238B / 238B
-> => sha256:a755ea659b0094999e0b6708cfff7317f2f1bc45ff3a50eaf524aba454cd3990 2.87MB / 2.87MB
-> => sha256:7d3e25cf7bd07ea20c813dd-fe86abf839e98a61b64f542671b21d6f973e9f3a 18.30MB / 18.30MB
-> => extracting sha256:7d3e25cf7bd07ea20c813dd-fe86abf839e98a61b64f542671b21d6f973e9f3a 1.0s
-> => extracting sha256:84a26a2849e127cec54f32f2474ff35efe05da4e5db1bccaaff0e53a5f0b26e6f 0.0s
-> => extracting sha256:a755ea659b0094999e0b6708cfff7317f2f1bc45ff3a50eaf524aba454cd3990 0.3s
-> [internal] load build context
-> => transferring context: 6.73MB
-> [2/9] WORKDIR /bonobo-board
-> [3/9] COPY
-> [4/9] RUN rm -rf CHANGELOG.md docker-compose.yml Dockerfile LICENSE.md nginx/ README.md
-> [5/9] RUN pip install --upgrade pip
-> [6/9] RUN pip install -r requirements.txt
-> [7/9] RUN pip install -r ./modules/dhbw/requirements.txt
-> [8/9] RUN python manage.py migrate
-> [9/9] RUN python manage.py collectstatic --no-input
-> => exporting to image
-> => exporting layers
-> => writing image sha256:062c785c32f53fb95d4280049440f72b872a604a56e5410885f43bac168e582
-> => naming to docker.io/library/bonobo-board_appseed-app
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
- Container nginx Created
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. Damit ist die lokale Installation des BonoboBoard abgeschlossen.

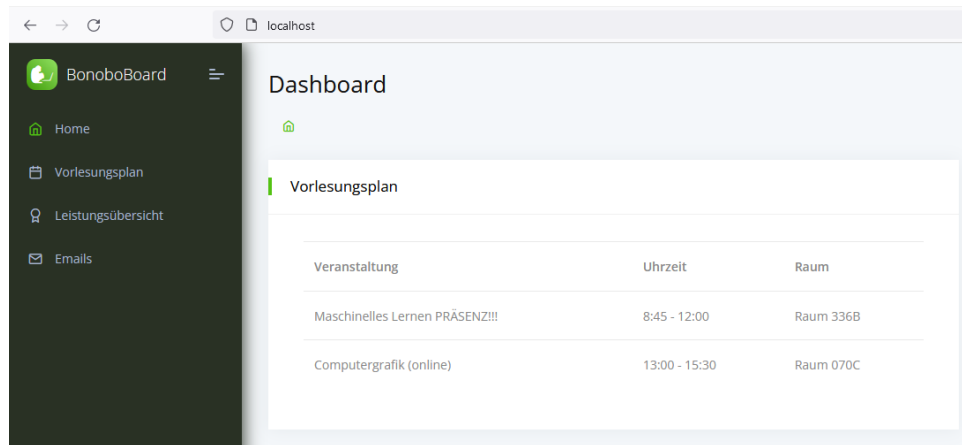


Abbildung 4: Darstellung im Browser

2.4 Installationsprobleme

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.

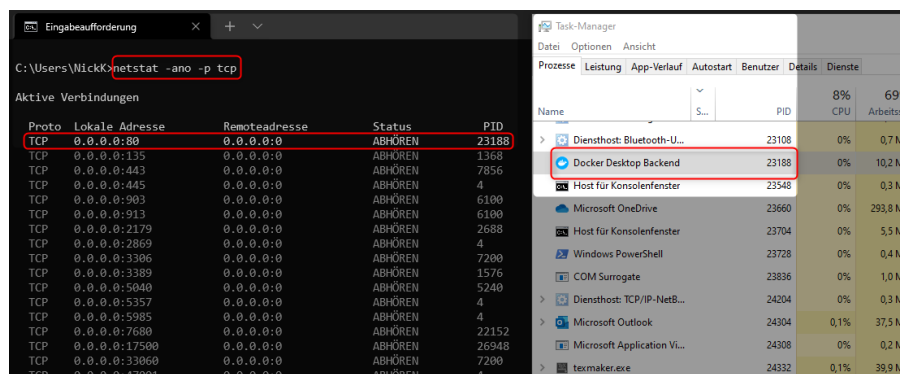


Abbildung 5: Überprüfung des lokalen Port 80

Unter **Ubuntu (Linux)** ist die Abfrage durch `sudo netstat -peeanut | grep :80` möglich. Wird in der Kommandozeile ein Service aufgelistet, wird Port 80 bereits genutzt. Die angegebene PID in der letzten Spalte vor dem Servicenamen (**25338/docker-proxy**) kann dazu genutzt werden, den Prozess mittels `kill <PID>` zu terminieren.

Als letztes Mittel kann der Port in der `docker-compose.yml` Datei angepasst werden, um das Bono-boBoard auf einem von 80 verschiedenen Port zu starten.

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).

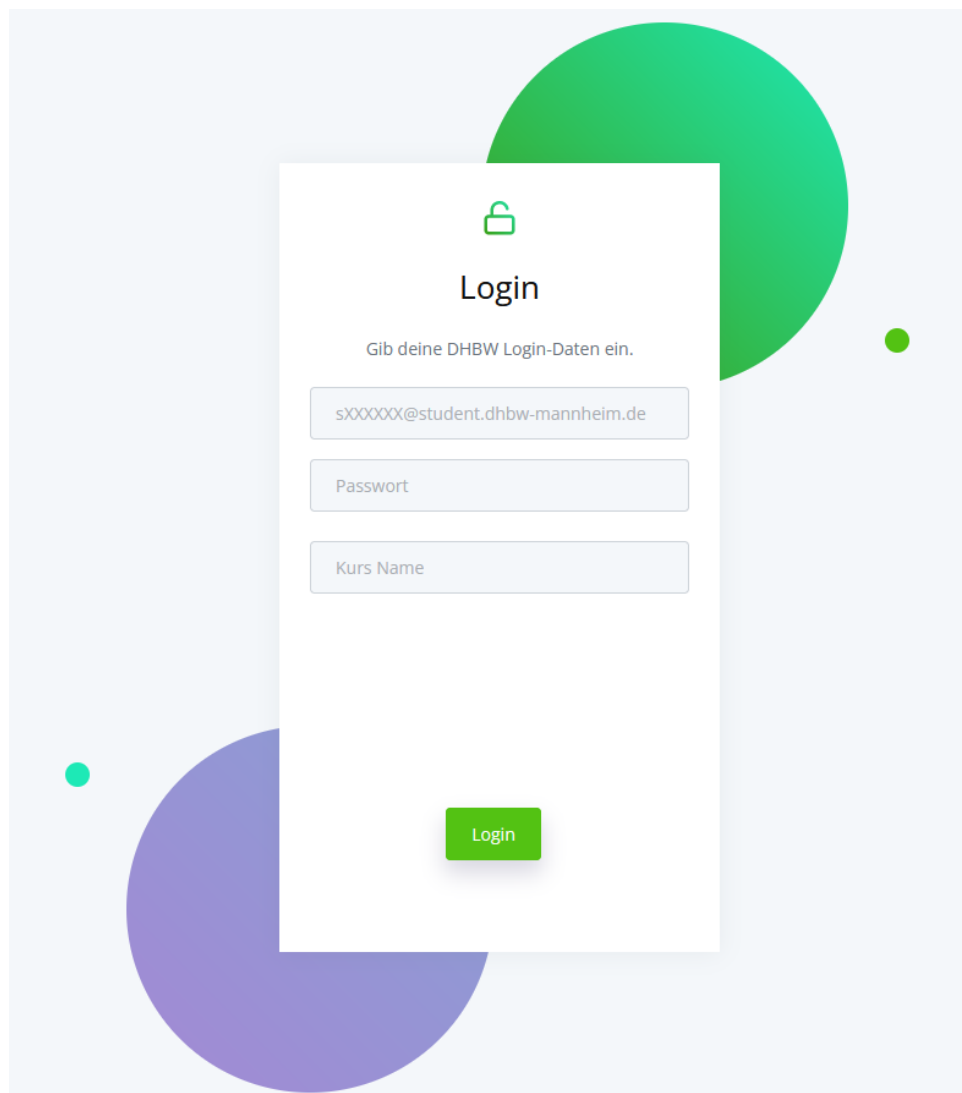
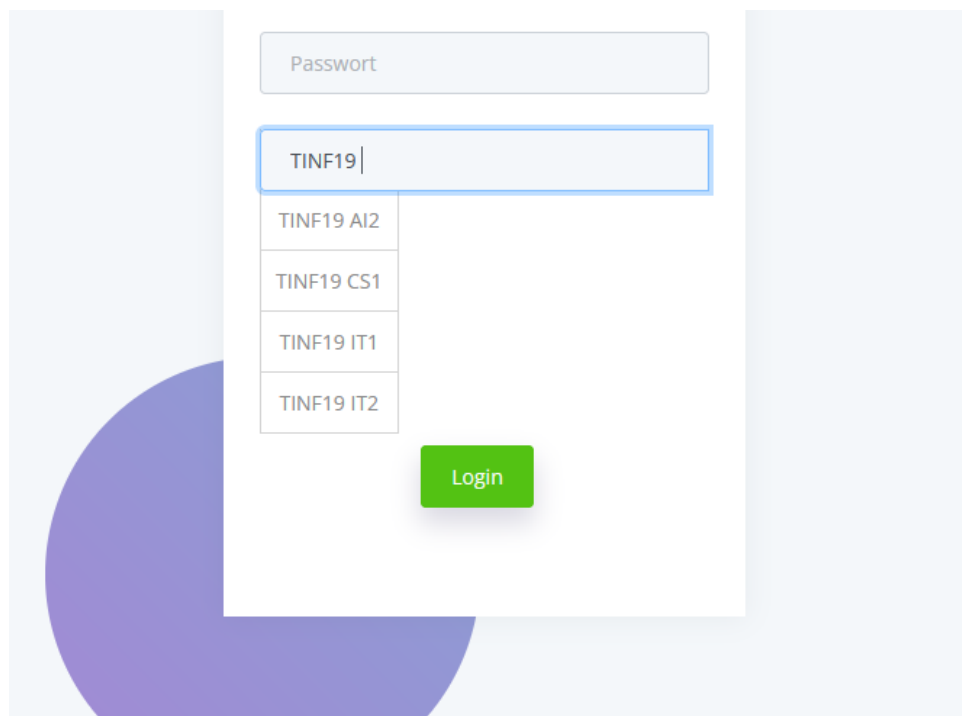


Abbildung 6: BonoboBoard - Login

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 einfach die Autovervollständigung (siehe Abb. 7).



The screenshot shows a login interface with a light blue background and a large purple abstract shape on the left. The login form is centered and contains the following elements:

- A text input field labeled "Passwort" (Password).
- A text input field containing "TINF19" with a blue border and a dropdown menu open below it.
- The dropdown menu lists four options: "TINF19 AI2", "TINF19 CS1", "TINF19 IT1", and "TINF19 IT2".
- A green "Login" button positioned below the course selection dropdown.

Abbildung 7: BonoboBoard - Kursauswahl

3.2 Home

Nach erfolgreichem Login gelangen Sie auf das Dashboard (siehe Abb. 8). Dort können alle Termine für den aktuellen Tag, sowie Ihre aktuelle Gesamtnote einsehen.

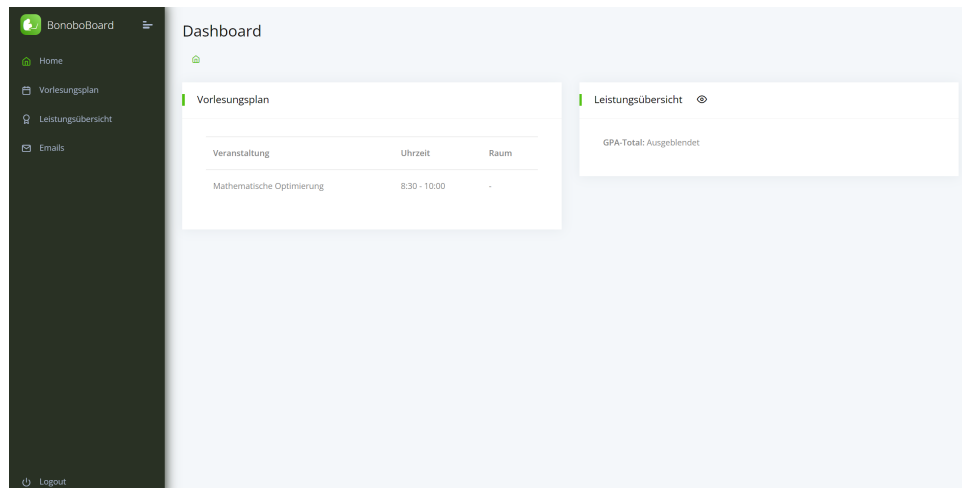


Abbildung 8: BonoboBoard - Dashboard

3.3 Seitennavigation

Das BonoboBoard lässt sich ganz einfach über die Sidebar bedienen (siehe Abb. 9). Von dort aus können alle wichtigen Funktionen des BonoboBoard mit nur einem Klick erreicht werden. Außerdem können Sie sich über den Button »Logout« am unteren Rand der Sidebar abmelden.



Abbildung 9: BonoboBoard - Seitennavigation

Die Sidebar kann mit einem Klick eingeklappt und wieder verankert werden.

3.4 Vorlesungsplan

Der Vorlesungsplan zeigt all Ihre Veranstaltungen der aktuellen Kalenderwoche an. Die Vorlesungstermine werden in einer Zeitleiste von 7:00 bis 21:45 zusammen mit ihren Titel, der Veranstaltungs-



zeit und der Raumnummer (falls vorhanden) angezeigt (siehe Abb. 10).

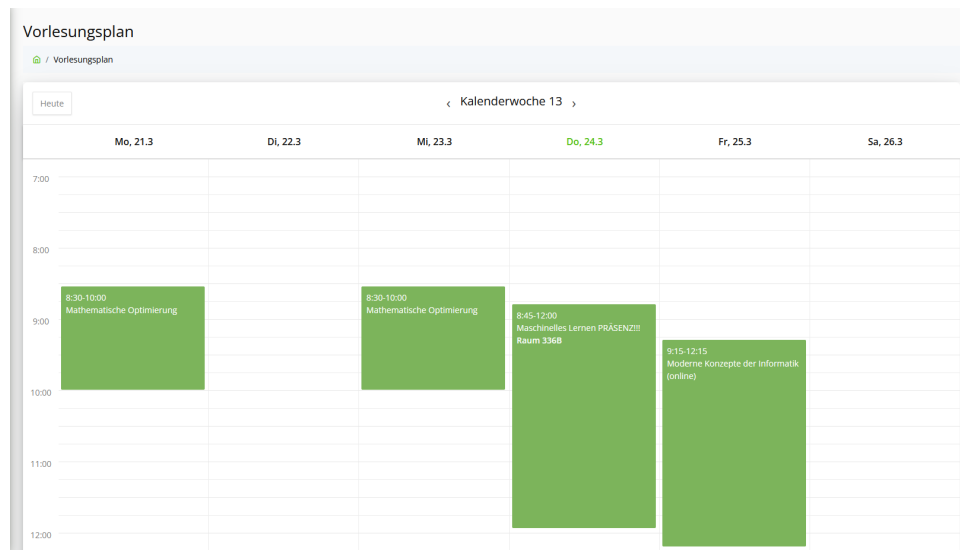


Abbildung 10: BonoboBoard - Vorlesungsplan

Über die Beiden Pfeile in der Kopfleiste (siehe Abb. 11) gelangen Sie zum Vorlesungsplan der vergangenen oder folgenden Wochen. Wenn sie auf den Button »Heute« tippen, gelangen sie zur aktuellen Woche zurück.

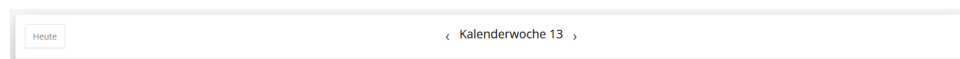


Abbildung 11: BonoboBoard - Vorlesungsplan Navigation

Die angezeigten Veranstaltungen werden von der Website »DHBW - Kurskalender« bezogen.

Mit einem Klick auf eine der angezeigten Veranstaltungen öffnet sich ein Dialog, in dem ein Link für den jeweiligen Online Vorlesungsraum hinterlegt werden kann. Dort können Sie einen Link aus einer Liste ausgewählt werden, die direkt aus dem DHBW Moodle System bezogen wird (siehe Abb. 12).

Veranstaltung: Mathematische Optimierung

Aktualisiere deinen Link für diese Veranstaltung.

Digitale Bildverarbeitung - SS22: Vorlesungsraum

eTeach_TINF19IT1 und 19IT2 Wahlmodul: Digitaler Kursraum
Wahlfächer

eTeach_TINF19IT2: BigBlueButton - virtueller Vorlesungsraum

eTeach_TINF19IT2: Virtueller Vorlesungsraum 3DCG

Maschinelles Lernen - 02: Vorlesungsraum ML

Link zur Veranstaltung

Abbrechen
Link speichern

Abbildung 12: BonoboBoard - Kursraumlink hinzufügen

Anschließend können Sie mit einem Klick auf »Zum Kursraum« direkt in den Online Vorlesungsraum gelangen (13).

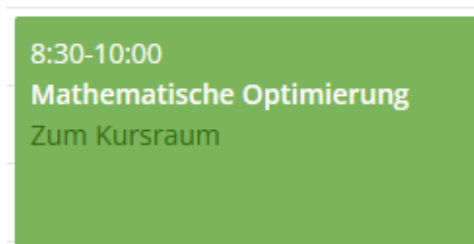


Abbildung 13: BonoboBoard - Zum Kursraum

3.5 Leistungsübersicht

In der Leistungsübersicht werden Ihre Noten angezeigt, die von Ihrem Dualis Account bezogen werden. Auf der linken Seite sind unter »Noten« die Einzelnoten der jeweiligen Fächer vermerkt. Rechts finden Sie unter »GPA« Ihre derzeitige Gesamtnote (siehe Abb. 11)

Leistungsübersicht

🏠 / Leistungsübersicht


Noten			GPA
Veranstaltung	Credits	Note	GPA-Total: 1.3
Mathematik I	8	1.9	
Theoretische Informatik I	5	1	
Theoretische Informatik II	5	1.5	
Programmieren	9	1	
Schlüsselqualifikationen	5	1.7	
Technische Informatik I	5	1.6	
Mathematik II	6	1.4	
Theoretische Informatik III	6	1.4	
Software Engineering I	9	1	
Datenbanken	6	1.1	

Abbildung 14: BonoboBoard - Leistungsübersicht

3.6 E-Mail

Das BonoboBoard ermöglicht es Ihnen schnell und einfach eine E-Mail von Ihrem Studierenden-Mailkonto (»S-Adresse«) zu verschicken. Navigieren sie dazu auf der Sidebar zu »Emails« oder klicken Sie im Dashboard auf »Email schreiben«. Dort können sie dann Empfänger, Betreff und Nachrichtentext eingeben und die E-Mail dann verschicken (siehe Abb. 15).

Email

 / Email

Email

Empfänger:

Cc:

Bcc:

Betreff:

Nachricht:

Nachricht. Bitte sei immer freundlich!

Abbildung 15: BonoboBoard - Email

Die E-Mail wird von Ihrem DHBW-Mailkonto verschickt und wird Ihnen somit auch in Ihrem Online Postfach als gesendete Mail angezeigt.

Das BonoboBoard zeigt Ihnen zudem alle in Zimbra angelegten Kontakte an. Mit einem Klick auf einen der Buttons neben Ihren Kontakten, können Sie dessen Email-Adresse als Empfänger, Cc oder Bcc hinzufügen.

4 Danksagung

Das Team von Optimal Connect wünscht viel Vergnügen mit dem BonoboBoard. Für Fragen und Anregungen oder bei etwaigen Fehlern, melden Sie sich gern via **GitHub**: <https://github.com/Software-Engineering-DHBW/BonoboBoard> bei uns.