

Software Design Dokumentation

Software Engineering II



Optima Connect

Projekttitle: BonoboBoard

Jakob Hutschenreiter (1419081)

Jiesen Wang (9839152)

Nick Kramer (3122448)

Patrick Küsters (2598689)

Peter Moritz Hinkel (2783930)

DHBW Mannheim

25. Februar 2022



Änderungshistorie

Revision	Datum	Autor(en)	Beschreibung
1.0	24.01.2022	JW	A: 1, 3
1.1	25.01.2022	JW PH NK PK	A: 2, 4

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

Inhaltsverzeichnis

1 Motivation	1
2 Entwicklersetup	1
2.1 Hosting bei 1Blu	1
3 Verwendete Tools	2
3.1 Tools für das Projektmanagement/Organisation	2
3.2 Tools für das Projektmanagement/Organisation	2
4 Zentrale Designentscheidung	2
4.1 Automatisierter Workflow (CI-Pipeline)	2
5 Zusammenfassung	2



1 Motivation

Um sicher zu gehen, dass alle Software Design Ziele des BonoboBoard-Projekts richtig umgesetzt werden, werden diese in diesem Dokument festgehalten. Das Dokument erleichtert die Analyse, Planung, Implementierung und Entscheidungsfindung. Anders als im Anforderungsdokument wird nun hier beschrieben wie die Ziele umgesetzt werden.

2 Entwicklersetup

Verwendete Software und Libraries Versionen:

Software/Library	Version
Django	4.0.2
Django Tables	2.4.1
dj-database-url	0.5.0
Gunicorn	20.1.0
Docker	20.10.12, Build e91ed57
PyCharm	2021.3.2
VisualStudio Code	1.64.2
Beautiful Soup	4.8.2
Requests	2.27.1
iCalendar	4.0.9
Pandas	1.4.0
lxml	4.5.0
SQLAlchemy	1.4.31
asgiref	3.4.1
autopep8	1.6.0
Python	3.9
pycodestyle	2.8.0
python-decouple	3.5
pytz	2021.3
sqlparse	0.4.2
toml	0.10.2
Unipath	1.1
whitenoise	5.3.0

2.1 Hosting bei 1Blu

Um die Webanwendung anderen Studierenden zur Verfügung stellen zu können, haben wir uns dafür entschieden, die Anwendung auf einem virtuellen privaten Server (VPS) laufen zu lassen. Der VPS läuft bei dem Hosting-Anbieter 1Blu. Da die Anwendung mit Docker bereitgestellt wird, hält sich der Konfigurationsaufwand auf dem Server in Grenzen. Auch ein einfaches NNeu-Bauen ist möglich, was im automatisierten Deployment-Prozess genutzt wird (siehe Abschnitt 4.1).

3 Verwendete Tools

Für ein gemeinsames strukturiertes Arbeiten am Projekt, werden je nach Aufgabe verschiedene Tools eingesetzt.

3.1 Tools für das Projektmanagement/Organisation

Tool	Einsatzzweck
Discord	Austausch von Nachrichten, Virtuelle Meetings, Umfragen, Informationsmanagement
WhatsApp	Austausch von Nachrichten
DropBox	Informationsmanagement
Google Kalender	Terminplanung
Jira	Aufgaben- und Projektmanagement, Prozessmanagement
Latex und Git	Dokumente erstellen und bearbeiten
OneNote	Informationsmanagement, Dokumentvorlagen erstellen und bearbeiten
Draw.io	Diagramme erstellen und bearbeiten
Git und GitHub	Versionskontrolle und Repository-Verwaltung
Adobe XD	Ideen Entwicklung, Erstellung von Mockups

3.2 Tools für das Projektmanagement/Organisation

Alle verwendeten Tools für die Softwareentwicklung sind aus dem separaten Dokument: „Tools für die Softwareentwicklung“ zu entnehmen.

4 Zentrale Designentscheidung

4.1 Automatisierter Workflow (CI-Pipeline)

Damit der Test und Deployment-Prozess nicht manuell ausgeführt werden muss, nutzen wir GitHub-Actions zur Verwaltung der Continuous Integration (CI). Dabei nutzen wir zwei Workflows:

- Ausführen der Unittests bei jeder Eröffnung eines Pull-Request
- Ausführen der Unittests und automatisches Deployment auf den VPS bei Änderungen auf dem Main-Branch

Da das automatische Deployment eine komplexere Schrittfolge beinhaltet, beschreiben wir diese genauer. Über GitHub-Actions wird ein Ordner erzeugt, welcher alle Dateien beinhaltet, um die Produktivumgebung auf dem Server zu starten. Dieses Artefakt wird auf den Server kopiert und anstelle der alten Produktivinstanz eingesetzt. Vorher ist es notwendig, die SSL-Zertifikate zu sichern und nach dem Umstellen wieder einzusetzen.

5 Zusammenfassung