

1 Aufgabe

Entwicklung eines Backends für eine der REST-Schnittstellen „reservations“ oder „personnel“ wie in gitlab.com/biletado/apidocs in Form von OpenAPI dokumentiert.

Jedes Backend muss mindestens eine Anfrage an eine andere Schnittstelle stellen, um die eigene API vollständig bedienen zu können. Der Endpunkt für diese Anfrage soll konfigurierbar sein.

Für bestimmte Anfragen ist eine Authentifizierung nötig. Für die Authentifizierung werden Json Web Tokens (JWT) verwendet. Bestimmte Berechtigungen (Autorisierung) müssen nicht geprüft werden, aber die Gültigkeit inklusive Signatur des JWT. Die Authentifizierung kann, insofern möglich, auch über eine Middleware/Reverse-Proxy statt im Backend-Quellcode geprüft werden.

2 Bewertung

Die Abschnitte im backend-howto ([permalink](#)) werden wie folgt bewertet:

containers	15%	Das Backend kann als container nach Spezifikation der OpenContainerInitiative gebaut und gestartet werden.
script everything	5%	Skript für den Containerbau und Tests in standardisierter Umgebung ohne spezielle Vorbedingungen wie installierte Entwicklungstools.
choose your framework wisely	15%	Aktuelle Version
secure your queries	5%	
clean code	5%	
validate the JWT	10%	
tracing	nicht bewertet	
configurability	5%	
compose-environment	5%	
licensing	1% (bonus)	
version-control	1% (bonus)	
write documentation	5%	
testautomation	5%	Ein einzelner Test reicht, egal was für einer, hauptsächlich automatisiert.
logging	5%	
API	20%	

3 Gruppenteilnehmer

Liste aller Teilnehmer:innen dieser Gruppe

4 Abgabe

Für die Abgabe wird benötigt:

- dieser Fragebogen
- compose-Konfiguration
- backend-Quellcode
- container-image des backends

Auch, wenn die compose-Konfiguration, der Quellcode und/oder das Image öffentlich erreichbar sind, bitte wenn möglich trotzdem bei wenigstens einem Gruppenmitglied in Moodle hochladen, für den Fall, dass mit dem Abruf etwas schiefgeht (fehlende Berechtigung, etc.). Für den container bitte `docker save (docs)` verwenden.

5 Quellcode

compose

backend

backend-container

6 Lizenz

Lizenz des Backend-Quellcodes und dessen Dateiname.

7 Fork

Link zur Projektbasis des Backends, falls Anwendbar.

8 Sprache und Frameworks

Verwendetes Framework.

Sprache

Sprachversion

Sprachversion ist aktuell Ja

Framework (FW)

FW-version

FW-version ist aktuell Ja

Website zum FW

Prepared statements/ORM

ORM Version

ORM Version ist aktuell Ja

Website zum ORM

9 Automatisierung

10 Testautomatisierung

11 Authentifizierung

JWT wird berücksichtigt

Signatur wird überprüft

12 Konfiguration & Dokumentation

Dokumentation existiert

Konfigurationsparameter sinnvoll gewählt

keine Hardcoded Zugänge zu angeschlossenen services (URLs, Passwörter, Datenbanknamen, etc.)

Mögliche Umgebungsvariablen sind dokumentiert

13 Logging

Logsystem des Frameworks wurde genutzt

Logs enthalten sinnvolle Details (z.B. object-ids)

Loglevel ist konfigurierbar