

Logging Dashboard Dokumentation

Logging Dashboard App

Installation

1. Das Git Repository von <https://source.ai.fh-erfurt.de/jo0756be/LoggingApp-Frontend.git> in ein beliebiges Verzeichnis klonen
2. NodeJS von <https://nodejs.org/en/download/> herunterladen und installieren
3. **npm install -g @angular/cli** ausführen
4. **npm install** ausführen
5. **ng serve** ausführen, um zu starten
6. Zu <http://localhost:4200/user/register> navigieren, um die App anzuzeigen

Logging API

Installation

1. Das Git Repository von <https://source.ai.fh-erfurt.de/jo0756be/LoggingApp-API.git> in ein beliebiges Verzeichnis klonen
2. NodeJS von <https://nodejs.org/en/download/> herunterladen und installieren
3. **npm install** im Root Verzeichnis ausführen
4. MongoDB von <https://www.mongodb.com/download-center/community> herunterladen
5. Im **C:\Program Files\MongoDB\Server\{VERSION}\bin** die Datei **mongod.exe** starten.
6. Im **C:\Program Files\MongoDB\Server\{VERSION}\bin** den Befehl **mongorestore.exe {PFAD ZU /mongodump IM GEKLONTEN REPO}** ausführen
7. **npm start bin/www** ausführen, um API zu starten

Verwendung

Um Logging Daten an die API zu senden, muss ein POST Request an **localhost:3000/logs** gesendet werden. Dabei wird der Log als Content Body mit **Content-Type application/json** gesendet. Der Request muss mit Basic Auth unter Verwendung der Login-Daten aus der Registrierung autorisiert werden.

Ein Log benötigt den namen eines gültigen Containers. In der Property **content** kann dynamischer JSON Inhalt eingetragen werden. Hier ein Beispiel:

```
{
  "container": "places",
  "title": "Place Added 2",
  "labels": ["place"],
  "content": {
    "id": "12346",
    "rating": 2,
    "user": {
      "id": "12345",
      "user": "place@test.com"
    }
  }
}
```

Testdaten

Die Testdaten werden durch den Befehl `mongorestore.exe` eingespielt.

Der Login für Test-Daten ist: `places-admin@test.de/jonastest`