

# Database Microservice

## Inhaltsangabe

1. Motivation
2. Technologien
3. Funktionsweise

## Motivation

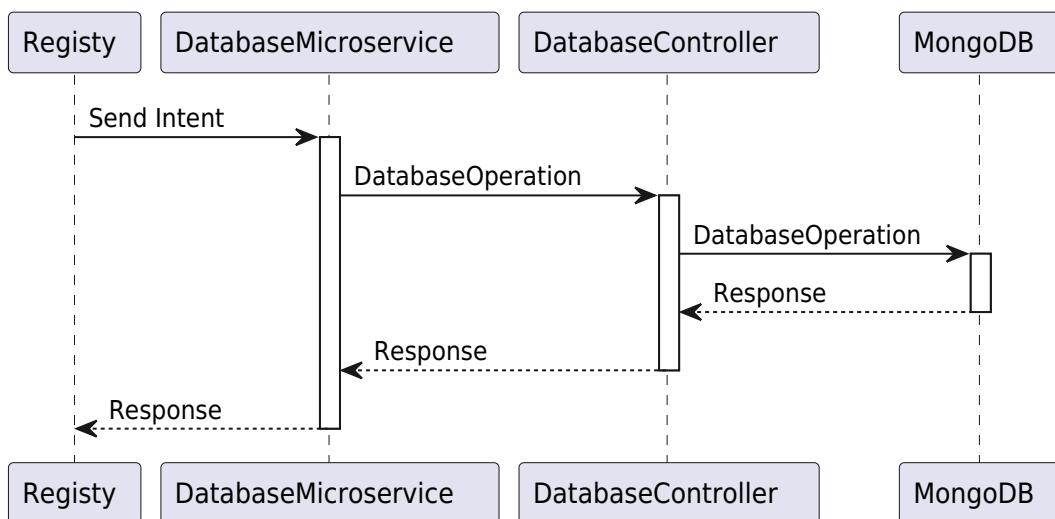
Um die Datenbank unabhängig von den anderen Microservices zu machen, mussten die Datenbank Operationen ausgelagert werden. Das führte dazu, dass die Intents von Rasa aufgelöst werden mussten, damit die richtigen Datenbank Operationen ausgeführt werden können.

## Technologien

Aufgebaut wurde dieser Microservice als REST-Server mit JavaScript. Die verwendeten Technologien dafür sind:

- NodeJS
- ExpressJS
- Axios

## Funktionsweise



Da dieser Microservice nur eine Route besitzt, über welche der Intent gesendet wird, muss der Microservice dazu in der Lage sein, diesen Intent aufzulösen, sodass die richtige Datenbank-Operation

ausgeführt wird. Der Intent kommt ursprünglich von Rasa.

```
{  
  "user": {  
    "id": 12345,  
    "telegram-id": 12345,  
    "nickname": "Al",  
    "details": {  
      "home": "Bonn",  
      "birthday": "23.06.1912",  
      ...  
    }  
  },  
  "intent":{  
    "name":"database-set",  
    "confidence":0.9998944998  
  },  
  "entities": [  
    ...,  
    {  
      "start":26,  
      "end":36,  
      "value":"krebstiere",  
      "entity":"allergen",  
      "confidence":0.9999893608,  
      "extractor":"CRFEntityExtractor"  
    },  
    {  
      "start":37,  
      "end":51,  
      "value":"alergisch bin.",  
      "entity":"detail-allergic",  
      "confidence":0,  
      "extractor":"CRFEntityExtractor"  
    },  
    ...  
  ],  
  "text":"Merke dir, dass ich gegen Krebstiere alergisch bin.",  
  ...  
}
```

Nutzungshinweis: Auf dieses vorliegende Schulungs- oder Beratungsdokument (ggf.) erlangt der Mandant vertragsgemäß ein nicht ausschließliches, dauerhaftes, unbeschränktes, unwiderrufliches und nicht übertragbares Nutzungsrecht. Eine hierüber hinausgehende, nicht zuvor durch *datenschutz-maximum* bewilligte Nutzung ist verboten und wird urheberrechtlich verfolgt.