

# Wöchentliche Aktivitäten

---

## 04.05 - 10.05

- Alle Docker Container gestartet
- Mit Postman die einzelnen Container angesprochen, teilweise Erfolg
- Eigenen Telegram-Token generieren lassen für Development Zwecke

## 11.05 - 17.05

- Den Code von dem Deconcentrator durchgehen und versuchen zu verstehen

## 18.05 - 24.05

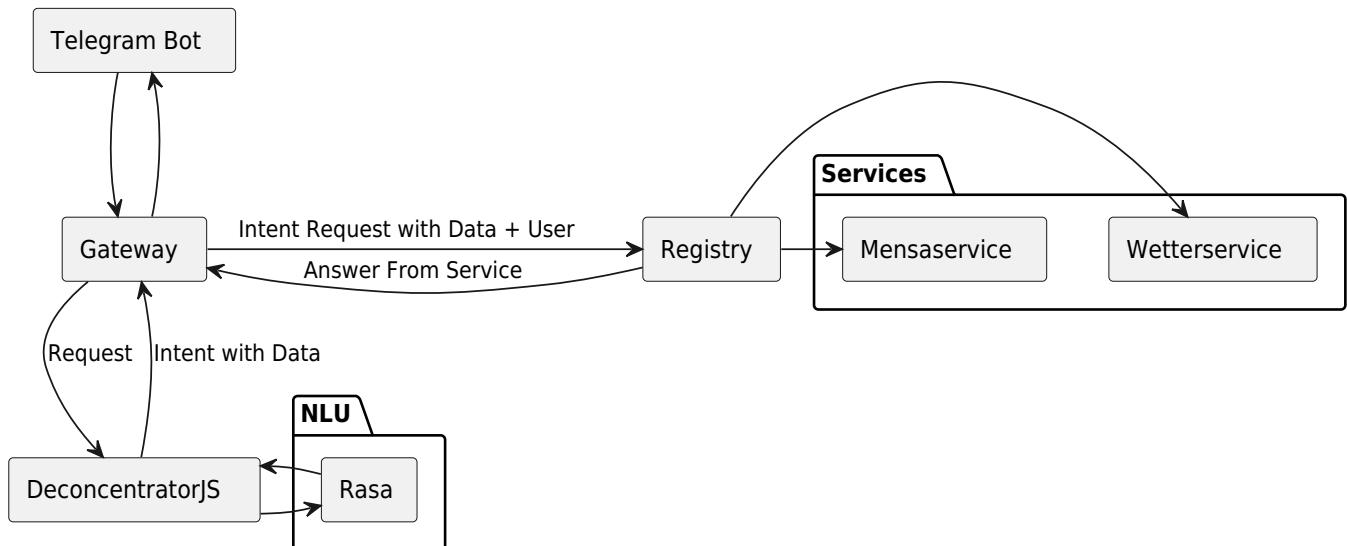
- Mit Postman die einzelnen Services angesprochen und korrekte Antworten erhalten, Deconcentrator ausgeschlossen. Dieser gibt eine 404 Response.
- Das ungefähre Format für die Registry lautet beispielsweise { „Test“: „1“, „intent“: { „name“: „wetter“ }, „entities“: [ „Wie ist das Wetter morgen ?“ ] }

## 25.05 - 31.05

- Absprache mit Teamkollegen über zukünftige Microservices und dessen Umsetzung, sowie über die Auswahl der Datenbank und wo man diese einsetzen kann.

## Diagram WiP

Zum selbstverständnis



## 01.06.- 07.06.

- Meeting für das Gespräch am folgenden Donnerstag, Aufteilung der Aufgaben (ca. 70 Minuten)
- Bearbeitung des Zwischenberichtes (ca. 330 Minuten)

## 08.06.- 14.06.

- [database](#) bearbeitet. Motivation und User Stories überlegt und hinzugefügt. Dazu nochmals im Internet belesen. (ca. 240 Minuten)

## 15.06.- 21.06.

- Meeting mit Lukas über [database](#) Thema: Aufteilung der Arbeit (30 min)
- Implementierung der [database](#) mit dem [MongoDB Javascript driver](#) (ca 10h)
- Getting Started für die Windows Version der [database](#) verfasst. (ca 30min)

## 22.06.- 28.06.

- [database](#) - Response Messages bearbeitet und Add/Change Details hinzugefügt, Fehler behoben (ca 6h)
- [database - commit 4eca34d](#) | [database - commit b74131d](#) - Code refactored und deleteDetail und deleteAllDetails hinzugefügt (ca 5:30h)
- [database API](#) Schnittstellen dokumentiert (ca 30min)

## 29.06.- 05.07.

- 2 Meetings mit Lukas (50 min ca)
- [database - commit 52af5cf](#) | [database - commit ae50c5e](#) - Ein paar Bugs gefixed und getDetailsFromUser() function hinzugefügt und README.md bearbeitet (4h)

- [database-microservice - commit fc44113](#) Projekt erstellt und Skelett erstellt für den neuen Microservice (3h)
- [database-microservice - commit 8873be5](#) Den Intent Resolver implementiert, welcher die korrekte Datenbank Operation anstoßen sollte (5h)

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.