

# cache

## Motivation

# Warum erstellen wir den Cache?

## Requirements

# Was soll der Cache können?

# Muss hier nicht viel sein denke ich...

## Functional

- /CAF100/ The system must ...
- /CAF100/ The system must ...
- /CAF100/ The system must ...
- /CAF200/ The cache must ...
- /CAF200/ The cache must ...
- /CAF200/ The cache must ...

## Non Functional

- /CANF100/ The system must ...
- /CANF100/ The system must ...
- /CANF100/ The system must ...
- /CANF200/ The cache must ...
- /CANF200/ The cache must ...
- /CANF200/ The cache must ...

## User Stories

# Brauchen wir hier glaube ich garnicht, bzw. sollte das mit den User Stories für den BeuthBot oder die Registry definiert werden...

"Als <Rolle> möchte ich <Ziel/Wunsch>, um <Nutzen>"

- /CAUS100/ Als Student möchte ich ...
- /CAUS101/ Als Student möchte ich ...
- /CAUS102/ Als Student möchte ich ...

## Use Cases (?)

### Technologies

# Mit welcher Technologie / Umgebung / Framework / Programmiersprache werden wir den Cache implementieren?

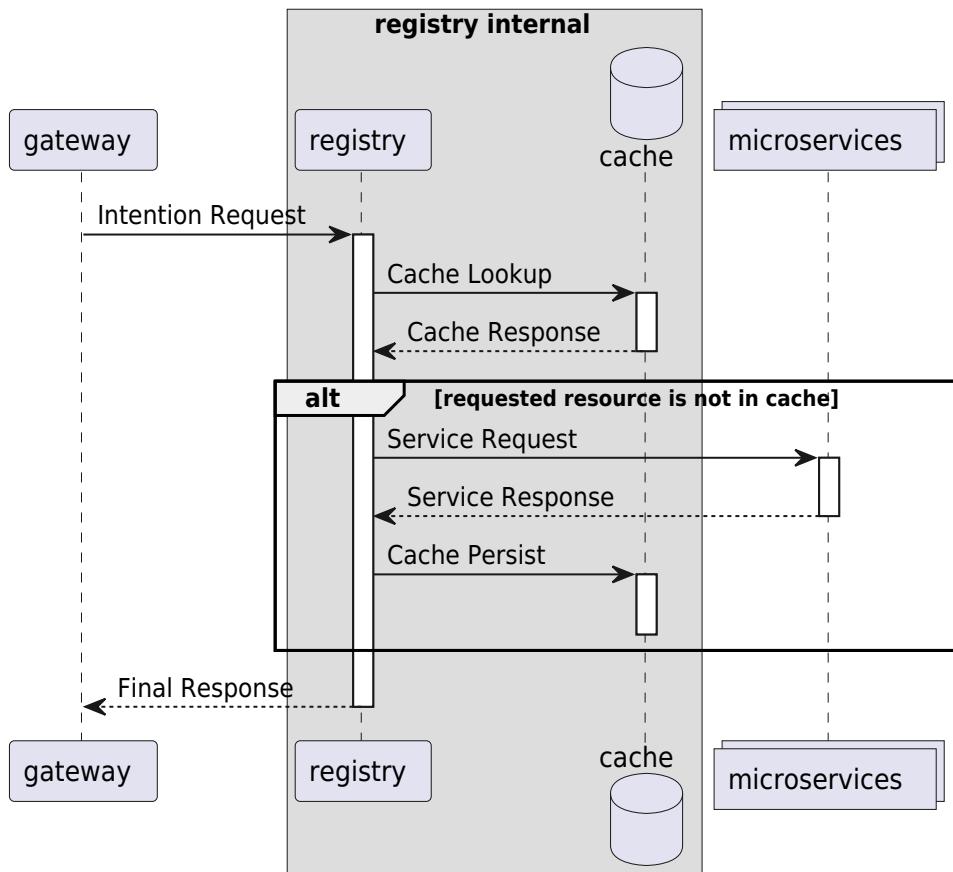
- Node.js Cache (?)

### Integration

# An welcher Stelle wird der Cache in das System eingebaut? (Registry)

# Wie wird sie eingebaut?

# Wie sieht das Endprodukt aus?



### Abnahmekriterien

# Was musst der Cache / die Registry können, damit wir das Projekt als erfolgreich definieren können?

## Resultierende Aufgaben

# Was müssen wir also tun?

# Auflistung der Tickets die entstehen...

# Vielleicht lieber direkt bei GitHub

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.