

Registry

The **registry** is part of the BeuthBot microservice system. Here the evaluated messages arrive and it is decided which service should be used. The reply from the service is forwarded to the gateway.

Table Of Content

1. [Registry](#)
2. [Table Of Content](#)
3. [Getting Started](#)
 - a. [Prerequisites](#)
 - b. [Cloning](#)
 - c. [Installing](#)
4. [Overview](#)
5. [Structure](#)
6. [Functionalities](#)
7. [Further Development](#)
8. [Further Reading](#)
9. [Built With](#)
10. [Versioning](#)
11. [Authors](#)

Getting Started

Prerequisites

- [node.js](#)
- [express.js](#)
- [axios](#)

Cloning

Get the source code by cloning its repository via <https://github.com/ziemers/beuthbot-registry>

Installing

After cloning the repository, you will need to make sure that you have node and npm installed on your working system. To check if you already have node installed, try

```
node --version
```

Same for checking if npm is installed, just with npm instead of the node command

```
npm --version
```

If you don't have node or npm installed, download the Software via the links provided in [Prerequisites](#) or search for them via your preferred search engine.

After that install all necessary dependencies

```
npm install
```

Now you can start the local development server to play around with the API and make your calls

```
npm run dev
```

This will fire up a development server that listens on port 3000.

Overview

The registry was built as a regular machine for the chatbot. Here it is decided which service is responsible for delivering the answer to the user. The decision is made on the Intent of the NLU unit.

Structure

The registry consists of two files of a JavaScript file which reflects the service. And a Json file which contains the configuration of the available services. Here each intent is assigned a corresponding service interface.

Functionalities

The registry forwards the incoming requests as a request to the various services and waits for their response before forwarding them to the requester.

Further Development

To be able to offer further services the corresponding intent only needs to be included in the json file with an interface. Later it would be possible to extend this service with a better self-learning rule engine.

Further Reading

-[Drools](#)

Built With

- [Node.js](#)
- [Express.js](#)

- [Axios](#)
- [Body-Parser](#)

Versioning

We use [GitHub](#) for versioning.

Authors

Timo Bruns - *Initial work* - [GitHub](#)

Christopher Lehmann - *Initial work* - [GitHub](#)

See also the list of [contributors](#) who participated in this project.

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.