Software Architecture Project 2

Runtime Components

There are three grain-coarsed components: Client, Backend and Database.

The Client is a single page web app (SPA) presented by an internet browser. There is no additional installation of software besides that browser required. This was a requirement.

The Client communicates with the Backend via HTTP-Requests (natively supported industry standard) to submit queries that are generated by the Client's user. The Backend translates those domain specific queries into SPARQL queries, that are forwarded to an external graph Database. This is done so that the Database does not need to be visible and accessible directly to the end-user. The query result received as response from the Database is projected in a domain specific model and then redirected to the Client.

The Database is a graph database (neo4j or RDF) and is provided by the customer.

Code Components

The Frontend is made up out of single components that each represent and present different and dedicated aspects of the user interface.

To coordinate the components, they interact with a Store/State-Manager, which is the single source of truth for data in the app. The store communicates with Services that perform requests to the Backend server.

Technology Stack

The Client runs via HTML, CSS and JavaScript in a Browser. React is used as the technology to render the pages.

The Backend is required to be a NodeJS Server. At the current state, it is still open for discussion if it should run in a docker container or not, as well as further backend frameworks.

The Database is either a RDF or a neo4j database (requirement).

Runtime Components

Client

Browser

Backend

QueryHandler

HTTP-Request

SPARQL-Queries

Database

RDF

neo4j

Code Components


```
┌─────────────────────┐   ┌─────────────────────┐   ┌─────────────────────┐
│ UI                  │   │ Data                │   │ Backend             │
│                     │   │                     │   │                     │
│  ┌──────────────┐   │   │   ┌──────────────┐  │   │   ┌──────────────┐  │
│  │ Component A  │───┼───┼──▶│    Store     │  │   │   │    Server    │  │
│  └──────────────┘   │   │   └──────────────┘  │   │   └──────────────┘  │
│                     │   │          │          │   │                     │
│  ┌──────────────┐   │   │          ▼          │   │                     │
│  │ Component B  │   │   │   ┌──────────────┐  │   │                     │
│  └──────────────┘   │   │   │   Service A  │──┼───┼──────────┐          │
│                     │   │   └──────────────┘  │   │          │          │
│                     │   │   ┌──────────────┐  │   │          │          │
│                     │   │   │   Service B  │◀─┼───┼──────────┘          │
│                     │   │   └──────────────┘  │   │                     │
└─────────────────────┘   └─────────────────────┘   └─────────────────────┘
```
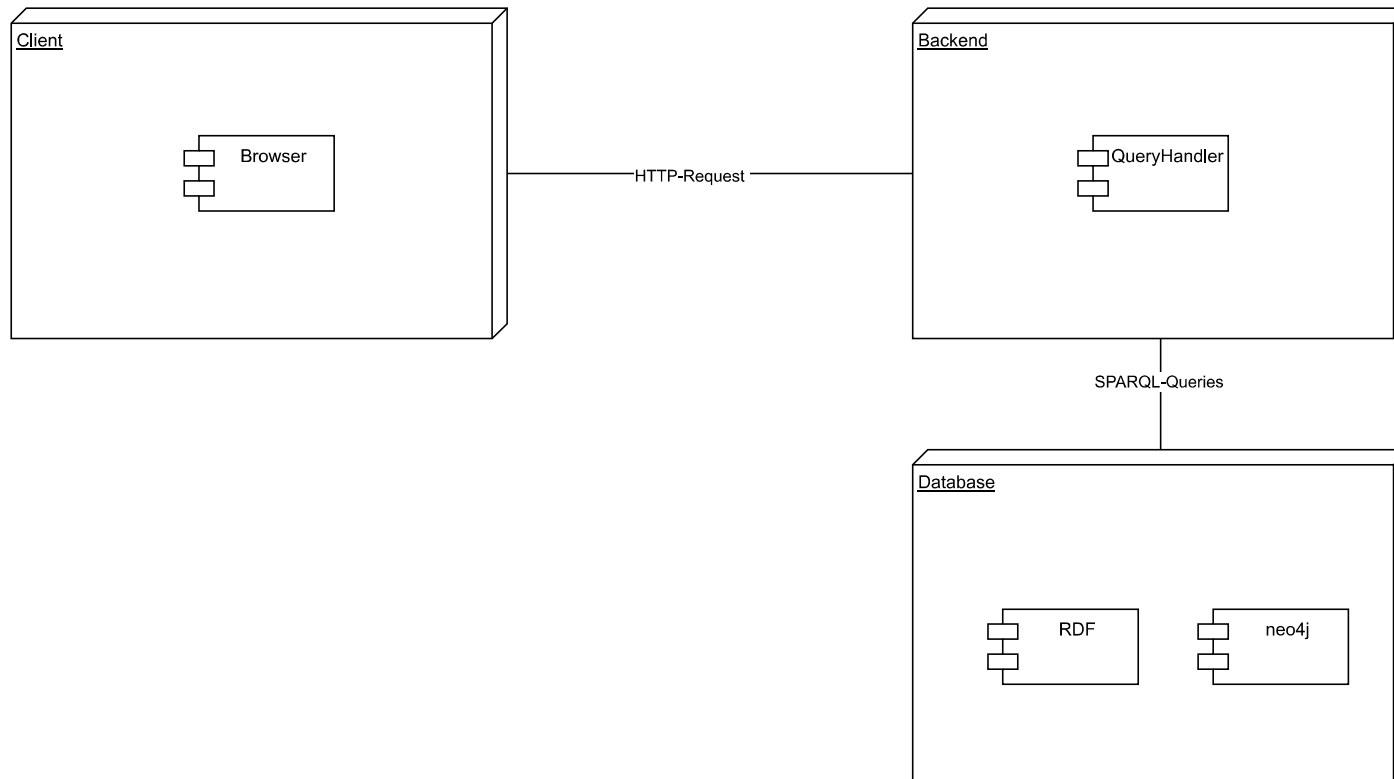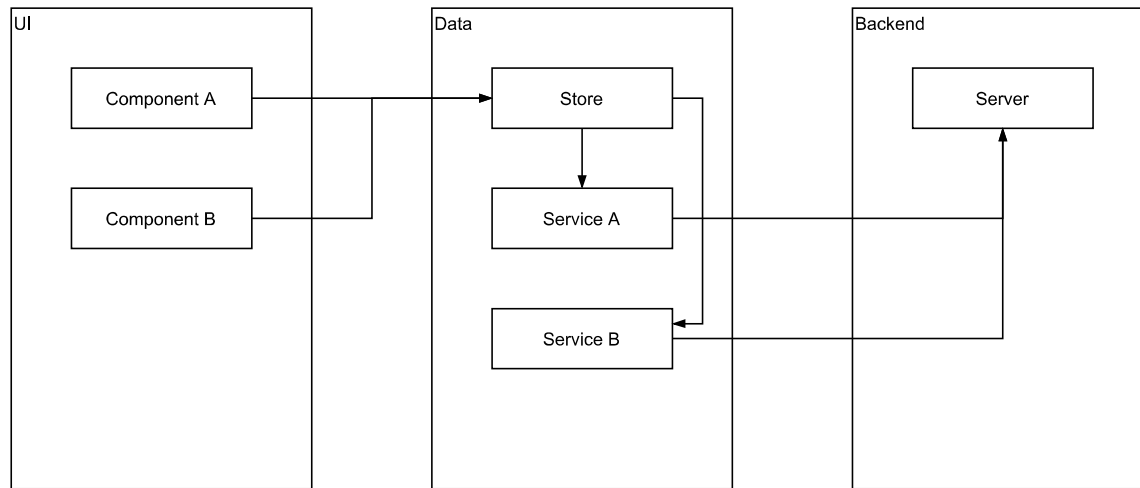
Die Components und Services sind nur Beispiele, da sie bis jetzt nicht weiter spezifiziert werden können.

Technology Stack

Client-Side	Server-Side	Data
React		
JavaScript		
CSS	NodeJS	RDF
HTML	(Docker)	neo4j
Client Browser	AMOS Server	Data Server

Existing

Created During Project