Technical Design Documentation

m4xrdk edited this page 6 minutes ago · 12 revisions

Edit New Page

Technical Design Documentation

This technical design documentation provides information on the architectural design decisions of the product. It will illustrate the components of the system and their relation with each other. Users of the software will gain an understanding of how the product works as well as understand the front- and backend architecture decisions.

Software Architecture

Overview

-TODO: Introduction Text for our Software Architecture

Architecture & Design Principles

Backend

The Backend is built in a Microservice Architecture style with Spring Boot, which are listed as follows:

- ProductArea Service (Handles the product areas. You can get listed all the product areas or create new product areas here)
- ProductRating Service (Handles the ratings for products. You can create product ratings, update product ratings or find the rating for a specific product here.)
- Product Service (Handles the products. You can for example create & delete products or update & find products by the ID here.)
- Project Service (Handles the projects. You can create projects, find & update projects or get shown all projects here.)
- Rating Service (Handles the ratings. You can get all ratings here or get a rating for a specific rating area.)
- User Service (Handles the user management, like create or delete a user, finding a user or update the users email address and password)

Implementation

TODO: Auf die genaue Implementation eingehen

Frontend

TODO:

Javascript CSS HTML

Framework -> React

Testing

TODO: Ausformulieren Unit Tests (J-Unit) Integration Tests Automatisierung durch CI/CD?

+ Add a custom sidebar

Clone this wiki locally

https://github.com/amosproj/amos2021ws0

Software Architecture Diagrams

Runtime Components

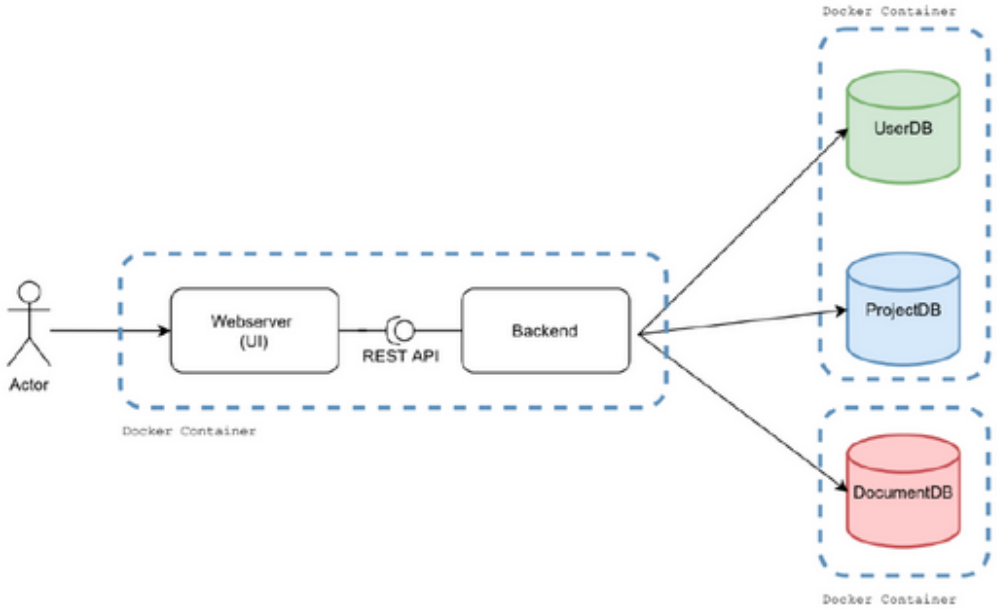

Figure 1: Architecture Model of Runtime Components

UML Diagram of Code Components

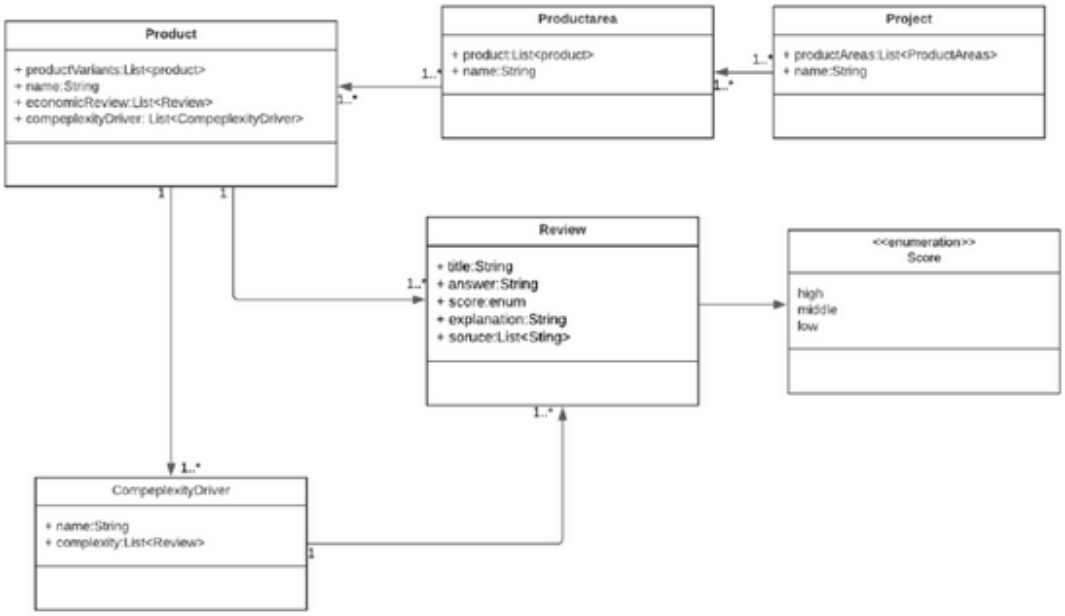

Figure 2: UML Diagram of Code Components

Technology Stack

In the following list you can see the projects underlying technology stack. It is a combination of the technologies, which we have used to build and run the application. To be more accurate it displays our used programming languages, front- and backend frameworks, software applications and our database decision.

1. Programming Languages

- *Java (Backend)*: Java is an object-oriented programming language which consists of a development tool for creating code and a runtime environment to run code.
- *JavaScript (Frontend)*: JavaScript is a scripting & just-in-time compiled programming language for web pages. It supports object orientation and is imperativ.
- *HTML (Frontend)*: HTML is a markup language for documents which can be displayed on web pages. It is used for structuring web pages and the content on the web page.
- *CSS (Frontend)*: CSS is a style language which describes the presentation of a web site written in HTML.

2. Frameworks

- *React (Frontend)*: Is an open-source frontend framework that is based on JavaScript.
- *Spring Boot (Backend)*: Spring Boot is an open-source java framework used for a microservice architecture.

3. Version Control Software

- *Git*: Git is a distributed version control system with the aim to help our group to develop the software together.
- *GitHub*: GitHub is the place where our repository lays. It is a provider for hosting software development and version control using Git.

4. Database System

- *HyperSQL*: Is a relational database system written in Java.

5. CI/CD-Tools

- *GitHub Actions*: Is a continuous integration and continuous delivery platform, which automates building, testing and the development.

6. Other development tools

- *Docker Engine*: Docker is for isolation of an application using containers.
- *Docker CLI*: It is the Command Line Interace using Docker.