

Customer: Rentschler & Holder Rotebühlplatz 41 70178 Stuttgart

Supplier: Team 2 (Paul Brenner, Jonas Alexander Graubner, Mohaddeseh Tibashi, Selvana Dwi Ayunda, Luka Dominik Pavic) Rotebühlplatz 41 70178 Stuttgart

Version History

Version	Date	Author	Comment
0.1	30.10.2022	Jonas Graubner	Created
0.2	09.11.2022	Jonas Graubner	Added Subsystem Specification
0.3	03.04.2023	Jonas Graubner	Changed AASX Server implementation
0.4	28.04.2023	Jonas Graubner	Changed Module implementation
1.0	12.05.2023	Jonas Graubner	Finalization

Table of content

- 1 Introduction
- 2 System Overview
 - 2.1 System Environment
 - 2.2 Software Environment
- 3 Architecture
- 4 System Design
- 5 Subsystem Specification
 - 5.1 MOD01 MongoDB
 - 5.2 MOD02 AASX Server
 - 5.3 MOD03 Web Asset View
 - 5.4 MOD04 Web User Administration
- 6 Technical Concepts

1 Introduction

The goal of this project is to develop a web application that acts as a management system for the "Asset Administration Shell" (AAS). This specific web application shall have an identity and access management as well as a user administration with persistent data storage in MongoDB. The user administration enables a role distribution of the users in the user groups "Admin", "Advanced" and "Basic", whereby the role distribution is carried out manually via the Admin. Each role is equipped with different access rights and read permissions ("Advanced" gets full read access to all AAS and their sub models and "Basic" gets read access only to the basic sub models to all AAS), with the admin also having functions for managing AAS content and user management. This uses the specification of the concept as a REST API in openapi.

2 System Overview

2.1 System Environment

The AAS-Management shall be implemented as a full stack solution with a web application as the user frontend. The Browser hereby acts as the code execution platform for the user frontend by running JavaScript. The Data is served by the Hypertext Transfer Protocol Secure (HTTPS) to the frontend. The backend consists of a MongoDB Database and an Rest-API which communicate with the frontend. The Frontend fetches all Data via the REST-API.

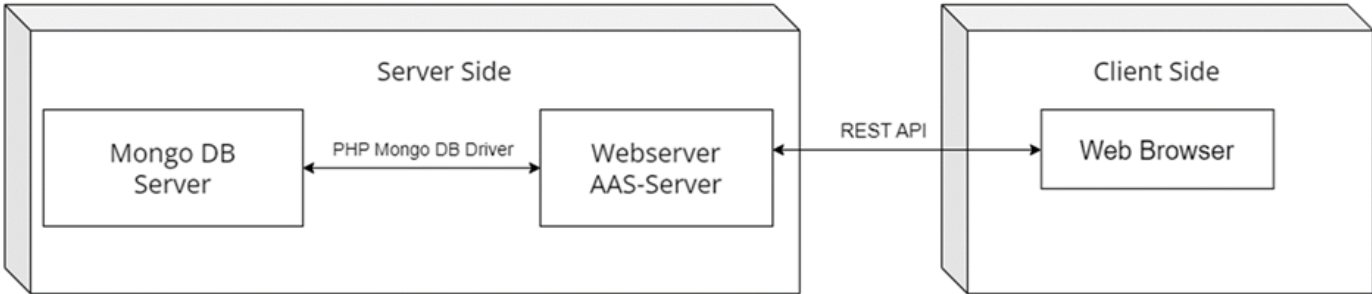
2.2 Software Environment

The Frontend is Build using the React framework, a JavaScript library for building user interfaces (UI). For broad compatibility the proprietary code is compiled into HTML, CSS and JavaScript Code after the development process. A node.js Server is used during local

development.

The AASX-Server is based on the AASX Server of the IDTA (Branch "MasterV3" - Commit [c5091b4b53d6788bff7865de3a39cb084a734981](#)). This Server was modified so that it communicates with an MongoDB for persistent storage. The Server is started with the security active.

3 Architecture



The Application is split into the Server Side and the Client Side. The Server Side contains an MongoDB Instance and the AASX Server. The Client Side contains the Web frontend.

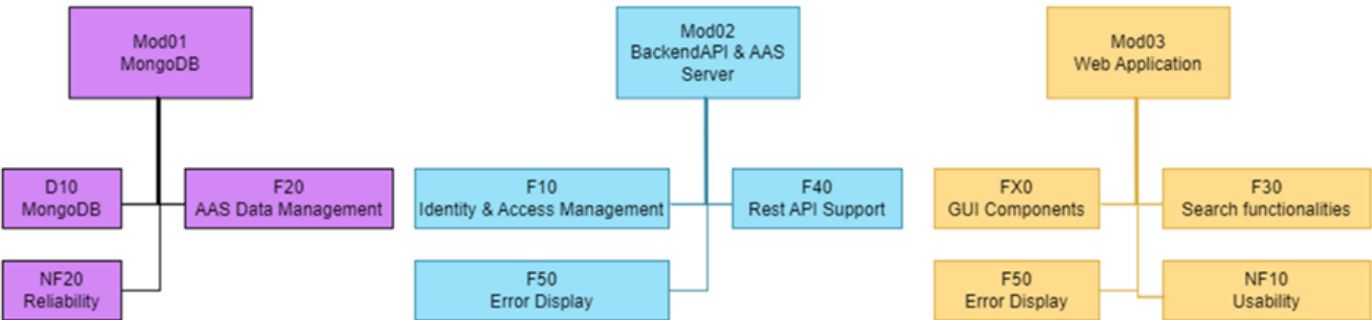
The MongoDB handles all Storgae Operations.

The AASX Server provides thee REST-API for the frontend and the Connection to the MongoDB. Furthermore is an Authentification Server provided by the AASX Server to handle the different user roles.

The Web frontend Calls the API and provides the user with an Interface for the AAS Management and the user controll.

4 System Design

5 Subsystem Specification



5.1 MOD01 MongoDB

Subsystem specification ID	MOD01
System requirements covered	<ul style="list-style-type: none">AASM-REQ7AASM-NF20 Reliability
Service	The MongoDB Server provides a reliable place to store all AAS Assets.
Interfaces	MongoDB C# Driver
External data	none
Storage location	https://github.com/JoTec2002/TINF21C_AAS_Management/blob/main/SOURCE/Server/aasx-server/src/AasxServerStandardBib/MongoDBInterface.cs
Module documentation	https://github.com/JoTec2002/TINF21C_AAS_Management/wiki/MOD01-MongoDB

5.2 MOD02 AASX Server

Subsystem specification ID	MOD02
System requirements covered	<ul style="list-style-type: none"> • AASM-REQ2 Identity & Access Management • AASM-REQ3 AAS content data management • AASM-REQ5 Rest-API Support • AASM-REQ6 Error Display • AASM-REQ7 MongoDB Integration • AASM-NF20/ Reliability • AASM-NF30 Performance
Service	The connection between the Backend and the Frontend is realised via the REST-API. All Data that is shown in the Frontend is provided via the REST-API. The API is sending the Data according to the user role.
Interfaces	software, REST-API
External data	none
Storage location	https://github.com/JoTec2002/TINF21C_AAS_Management/tree/main/SOURCE/Server/aasx-server/src/IO.Swagger.V1RC03/Services
Module documentation	https://github.com/JoTec2002/TINF21C_AAS_Management/wiki/MOD02-AASX-Server

5.3 MOD03 Web Asset View

Subsystem specification ID	MOD03
System requirements covered	<ul style="list-style-type: none"> • AASM-REQ1 GUI Components • AASM-REQ3 AAS content data management • AASM-REQ4 Search functionalities • AASM-REQ5 Rest-API Support • AASM-REQ6 Error Display • AASM-NF10 Usability • AASM-NF30 Performance
Service	The Web Asset View allows the user to View the Assets provided via the Server API. The user can login and perform actions according to his user role. Any unsolvable error is presented to the user.
Interfaces	Software, REST-API, Web View
External data	none
Storage location	https://github.com/JoTec2002/TINF21C_AAS_Management/blob/main/SOURCE/frontend/src/Pages/Guest.js
Module documentation	https://github.com/JoTec2002/TINF21C_AAS_Management/wiki/MOD03-Web-Asset-View

5.4 MOD04 Web User Administration

Subsystem specification ID	MOD04
----------------------------	-------

**Subsystem
specification
ID** **MOD04**

System requirements covered	<ul style="list-style-type: none">• AASM-REQ2 Identity & Access Management• AASM-REQ6 Error Display• AASM-NF10 Usability
Service	Via the Web Administration the user can add, edit and delete users. The user can login and perform actions according to his user role. Any unsolvable error is presented to the user.
Interfaces	Software, REST-API
External data	none
Storage location	https://github.com/JoTec2002/TINF21C_AAS_Management/blob/main/SOURCE/frontend/src/Pages/AdminDashboard.js
Module documentation	https://github.com/JoTec2002/TINF21C_AAS_Management/wiki/MOD04-Web-User-Administration

6 Technical Concepts

Release / Approval

Approval is made by the customer and the suppliers

Date:

Signature Customer:

Signature Suppliers:

Document author	Jonas Graubner
Created on	30.10.2022

Duale Hochschule Baden-Württemberg