Development of a web-based application for the configuration of IoT components

Project Proposal

Steffen Knoblauch

April 2021

1 Motivation

The connection of more and more devices to the internet not only enables the combination of various functionalities but also creates an increasingly dense network of different end devices and sensors. Consequently, more and more IoT applications are emerging that use a wide variety of combinations to enable new applications.
An example of such an IoT application is an intelligent car park barrier. For example, it can use a camera to detect a car entering the car park and automatically open the barrier. A wide variety of use cases and error szenarios arise for such an application, since errors can occur in practice, for example the malfunction of a sensor. The application must be able to react to these scenarios appropriately. In order to protect the IoT and the applications based on it against failures and attacks, the SORRIR research project was launched.
The goal of SORRIR is to develop a self-organising, resilient execution platform for IoT services.To enable this platform, different operation levels are introduced. These levels describe the functionality depending on the status of the component and its subcomponents. If, for example, a subcomponent fails, a lower operation level can be used after a so-called degragation. In the context of the car park, this would be the case, for example, if the automatic recognition of the number plate fails. This would then only allow limited use of the barrier, as the car would not be identified automatically. In this case, the driver would have to authenticate himself by other means of authentication.

2 Problem Statement

The IoT and its devices allow the use of a wide variety of combinations to create applications. In the context of the SORRIR platform this means there are also a variety of different applications that require a configuration for their behaviour.
The configuration not only requires a descriptions of the normal operation mode, but also the fallbacks when for example a subcomponent is not working anymore. Or in other words a description of the different degradation levels that will guarantee the appropriate handling of possible errors and problems. In order to be able to configure the application, a configurator is required. This configurator should allow the configuration of the involved components, their states and the different modes of operations and their degraggation levels.

3 Approach

The main goal of this project is the creation of a web application that will act as configurator for applications based on the SORRIR patform.
This web application has to fullfil several requirements to be able to configure such an application:

- Import/definition of subcomponents and their corresponding states

- (Graphical) Configuration of the different degradation levels

- (Graphical) Configuration of the state machines for the different levels

- Validation of the configuration

- Export of the configuration to documented JSON format

4 Planning

4.1 Own Background

I already have some working experience with web technologies and in addition some general experience with the IoT. My RNSEC-I project was the development of a context-aware Android application for the LoRaPark Ulm. This app used the IoT sensors and their data from the LoRaPark to interfere a context and execute actions on the smartphone of a user.

4.2 Required Resources

In general, no special resources are required. The development will be done on my personal computer and the code versioning is already setup in gitlab (`https://gitlab-vs.informatik.uni-ulm.de/rtx66/sorrir-configurator`)

4.3 Work packages

This project will be a 8 ECTS Project, starting from now until the end of the current semester.
The work will be splitted into several parts(chronological order):

- Determination of the technology stack that will be used for the development. This part consits of several desicions:
  - Usage of JavaScript or TypeScript
  - Usage of a framework: Angular/React/Vue/...
  - Evaluation of additional libraries, for example for the drag and drop handling

- Planning of the architecture (UML,...) and further exploration of the selected technology stack: learning of the requried fundamentals

- Implementation of the inital setup of a set of subcomponents (to be able to define the different subcomponents and their possible states) and the implementation of the first configuration part - the DAG that represents the configuration of the degradation levels

- Implementation of an export (of the already implemented configuration) to a JSON document

- Implementation of the state machine configuration for each degradation level and the extension of the configuration to be able to export the added configurations

- Development of the configuration validation

4.4 Contingency plan

The goal for this project is to complete it by the end of the semester. If my current subjects take up too much time or in the case of other time problems, an extension to the lecture-free period is possible.