

The logo for SORRIR consists of a horizontal line, followed by two diagonal lines forming a peak, and then two vertical lines. A small shield icon is positioned between the two vertical lines.

SORRIR Degradation Configurator

Steffen Knoblauch - Institute of Distributed Systems - Ulm University - 2021

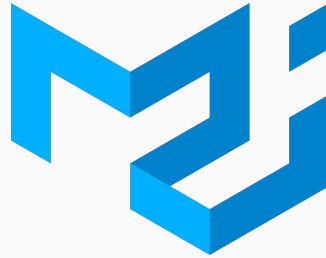
Background

- Composite system
 - Failure of subcomponents
 - Predetermined error handling
- Degradation Levels

Goal: web application for the degradation configuration

Technology Stack

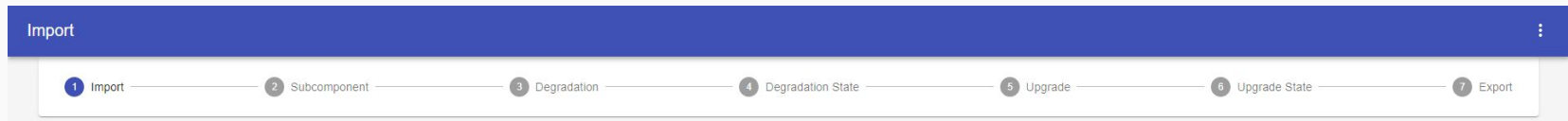
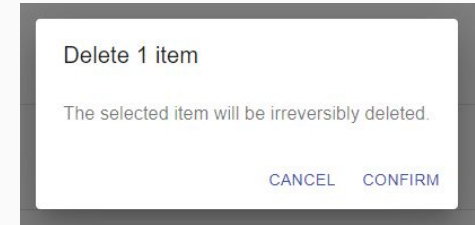
- React application
- Additional libraries



Architecture

- Modularity (components)
- Extensibility (future enhancements)

→ Wizard: step by step configuration



Subcomponent Configuration

- Configuration of subcomponents
- Dialog for creation, update and deletion

Subcomponents			+	
<input type="checkbox"/>	ID	Name ↑	Shadowmodes	
<input type="checkbox"/>	BDS	Barrier Detection Service	On	Off
<input type="checkbox"/>	BU	Barrier Unit	On	Off
<input type="checkbox"/>	CAM	Camera	On	Off
<input type="checkbox"/>	CS	Car Sensor	On	Off
<input type="checkbox"/>	CR	Card Reader	On	Off
Rows per page: 5 ▾ 1-5 of 9 < >				

Degradation Level Management

- Dialog for creation, update and deletion
- Multiple sets of dependencies
- Automatic level calculation

× Edit - Open 📄

Details

Level

2

Label

Open

Shadowmode dependencies

⊖

Barrier Unit	Traffic Light	Camera	Car Sensor	Touchscreen
On	DC	DC	DC	DC
Card Reader	Barrier Detection...	Plate Recognition...	Parking Managemen...	
DC	DC	DC	DC	

+

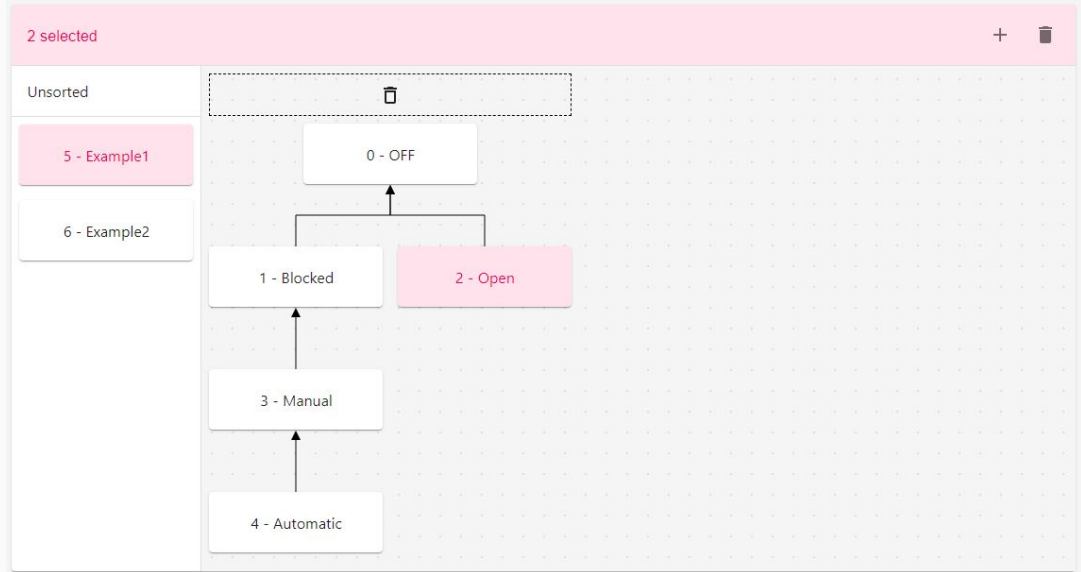
Internal states

States

open ×

Degradation Configuration

- DAG via drag and drop
- Degradations and upgrades as separate steps
- Automatic upgrade DAG calculation



Degradation Level States

- Internal state machine
- Configuration of state changes for degradations/upgrades
- Separate steps in the wizard

State changes for degradation from '1 - Blocked' to '0 - OFF'



State changes for degradation from '2 - Open' to '0 - OFF'



State changes for degradation from '3 - Manual' to '1 - Blocked'



State changes for degradation from '4 - Automatic' to '3 - Manual'



Import validation - JSON schema

- Import of existing configuration
- Validation of the imported JSON file
- Ajv + JSON schema

The screenshot shows a web interface for importing a configuration. At the top is a blue header with the word 'Import'. Below it is a progress bar with two steps: '1 Import' (active) and '2 Subcomponent'. Under the 'Import' step, there is a blue button labeled 'UPLOAD CONFIGURATION' and a text field showing the filename 'configuration2.json' with a '(This ...)' link. Below the filename, a red box contains four validation error messages, each preceded by a red exclamation mark icon:

- Validation: must have required property 'degradations'
- Validation: must have required property 'upgrades'
- Validation: (/subcomponents/0) must have required property 'name'
- Validation: (/degradationLevels/0) must have required property 'label'

DEMO

Logos:

- https://logos-download.com/wp-content/uploads/2016/09/React_logo_wordmark.png
- https://material-ui.com/static/logo_raw.svg
- <https://ajv.js.org/img/ajv.svg>