

UX Fault

- User documentation
- Specification
 - Current Problem List
 - Clear stuck alarms
 - Export
 - Alarm Notifications
 - Alarm Log
 - Fault status summary view
- Data Provider

User documentation

To operate a network, it is important to get an overview about the currently raised alarms. The application offers basic fault management of devices supporting ONF-TR-532. The alarms are classified according to the severity level (warning, minor, major, critical).

Overview Fault management:

CURRENT PROBLEM LIST ALARM NOTIFICATIONS (0) ALARM LOG						
icon	Timestamp ↓	Node Name	Count	Object Id	Alarm Type	Severity
▲	2021-02-10T07:22:36.5Z	sim2	0	2	critical	NonAlarmed

Specification

“Fault” User Interface should be implemented to show the Alarm notifications of the devices.

Below are the UI elements to be implemented.

The graphical user interface is separated into three tabs:

1. Current Alarm Notifications
2. Alarm Notifications
3. Alarm Log


The table view implements the common ux-framework-table supporting sorting, filtering and pagination on the server.


The following tabs should show a table with its respective data.

CURRENT PROBLEM LIST	ALARM NOTIFICATIONS (0)	ALARM LOG
----------------------	-------------------------	-----------

Current Problem List

It lists all current active faults in the network. In addition, it also lists alarms sent by the SDN controller itself, which detects connection losses to the NetConf server (connectionLossOAM) or which detects connection loss to a device via a mediator to a device (connectionLossNeOAM).


Reload Button  : Reloads/Refresh the Current Problems List

Sync button  : Clears stuck alarms

The Current Alarm Notification panel shows the table with the below columns.

- 1. icon
- 2. Time Stamp
- 3. Node Name
- 4. Count
- 5. Object Id
- 6. Alarm Type
- 7. Severity

CURRENT PROBLEM LIST ALARM NOTIFICATIONS (0) ALARM LOG

icon	Timestamp ↓	Node Name	Count	Object Id	Alarm Type	Severity
	2021-02-10T07:22:36.5Z	sim2	0	2	critical	NonAlarmed

Clear stuck alarms

The sync button allows refreshing the current alarms for the visual NEs.

If the alarms are not indicated as active anymore, they will be removed from the list.

Clear stuck Alarm icon will only be available when there are Faults available else it wont be visible.

Export

The "export to CSV" button will export the presented current problem list to a CSV File.

Alarm Notifications

As long as the page is not fully refreshed (the browser reloaded), it lists all alarm notifications reached by the SDN Controller. Previous alarm notification can be viewed in the alarm log.

Alarm Notifications should be collected in real-time via websockets.

The Alarm Notification panel shows the table with the below columns.


- 1. icon
- 2. Time Stamp
- 3. Node Name
- 4. Count
- 5. Object Id
- 6. Alarm Type
- 7. Severity

Alarm Log

Next to the currently active alarms, an alarm log lists all alarm notifications of the past.

The Alarm Log panel shows the table with the below columns.

- 1. icon
- 2. Time Stamp
- 3. Node Name
- 4. Count
- 5. Object Id
- 6. Alarm Type
- 7. Severity

 **Reload Button** : Reloads/Refresh the Alarm log

CURRENT PROBLEM LIST ALARM NOTIFICATIONS (0) <u>ALARM LOG</u>							
<div> </div>							
icon	Timestamp ↓	Node Name	Count	Object Id	Alarm Type	Severity	Source
	2021-02-10T07:22:36.5Z	test84	0	2	minor	NonAlarmed	undefined

Fault status summary view

Elements like Screenshot below:

- Elements are
 - Node status, showing the actually connected NEs number
 - Alarm Summary view
- If Fault is enabled, status window is visible all the time, independent of the actually selected app.
- If possible location is in the Title area at the right but left of Menu, Help or Logout
- Order of Status element is the same, that is provided by order of left side menus entries
- A click
 - onto the Alarm status area link to alarm app and alarm view
 - onto the Number of nodes links to the connect app
- The layout needs to be adapted to React/Material layout.
- Design discussion should be done: Is Node status part of Connect App and Fault part of Fault app or is both provided by Fault app?
- Help Button in status bar is provided by help app. Framework provides information about actual selected app.

Screenshot with Summary view

Alarm status: 6 | 2 | 1 | 2 |

Data Provider

The data is stored in an ElasticSearch database. The [Data-Provider](#) provides a REST-API to read the data.

- Current Problem List
 - get the current problem list: **read-faultcurrent-list**
- Alarm Log
 - get the alarm log: **read-faultlog-list**
- Summary View
 - get the fault status: **read-status**

Note: The alarm notifications are only updated via websockets. No REST-API is provided.