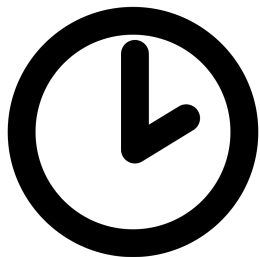

A new frontend for openHAB

Hochschule Darmstadt, 25 May 2019
Smart Home Day
Yannick Schaus, openHAB Foundation
Twitter: @yschaus · GitHub: @ghys



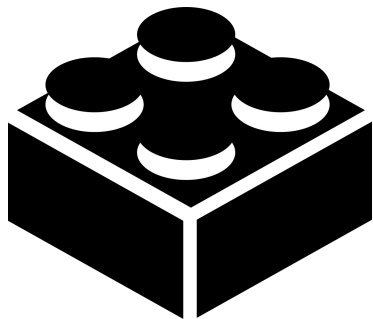
Timeline

- October-December 2018 (Smart Home Day, openHAB Foundation board meetings):
early discussions about replacing Paper UI
 - January 2019: early prototype
 - February 2019: approved by the Architecture Council
 - April 2019: Technology & vision approved by the UI team
 - ...
 - Planned to ship with the next major openHAB version
-



Design goals & vision

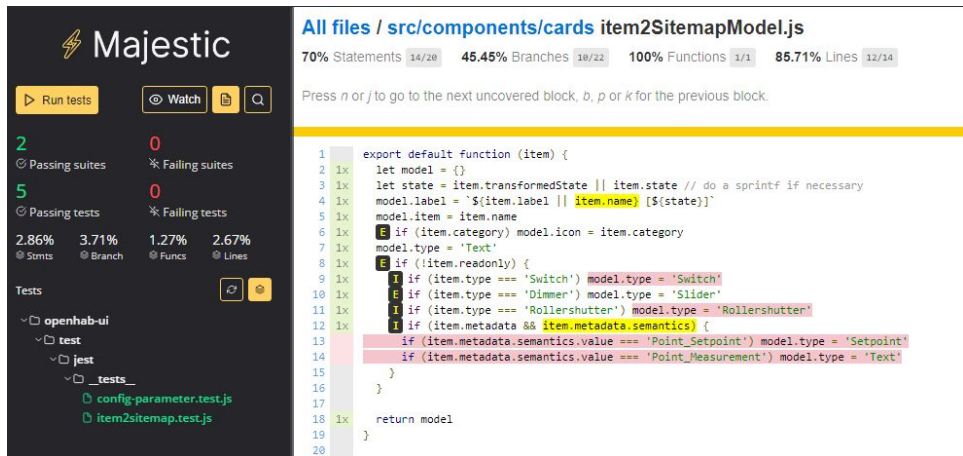
- Provide openHAB with a best-in-class “default” UI
 - Differentiation factor & appeal (Material Design + iOS look & feel)
 - Reduce the fragmentation & overlapping of functionality spread among several (web) interfaces:
 - 2 admin UIs (Paper UI, HABmin)
 - Home page (Dashboard)
 - Sitemap viewer (Basic UI)
 - Mobile (phones/tablets) as a first-class citizen
 - Add to home screen & Cordova apps
 - “Native app” feeling - but supplementing, not replacing native apps!
 - Not compromising on the desktop experience
-



Architectural principles

- Capitalize on well-known, battle-tested frameworks & techniques:
 - VueJS (HABot, website...)
 - Single-file components
 - Framework7 as components framework (<https://framework7.io>)
 - ECharts...
 - Avoid over-engineering and another maintainer-less situation: reduce the barrier to entry for contributors
 - Clear folder structure from the start, avoid unnecessary layers, easy to use testing infrastructure
-

Unit (Jest) & E2E (Cypress) tests



Majestic

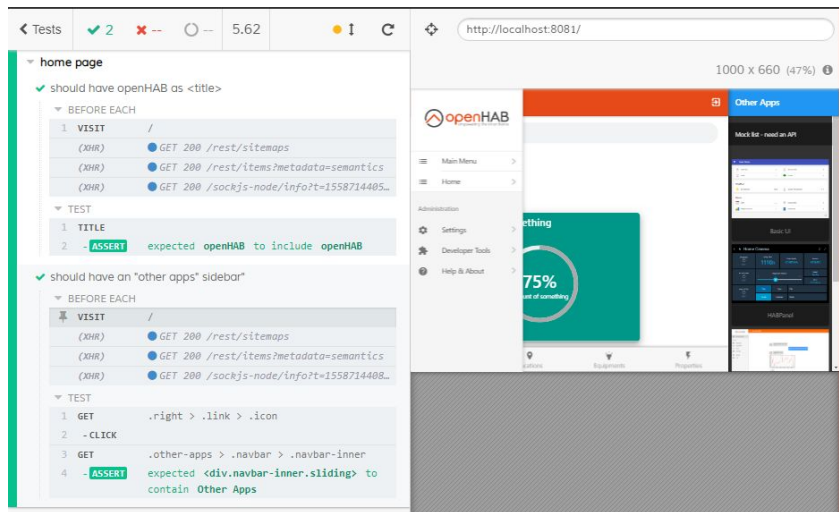
70% Statements 14/28 45.45% Branches 38/22 100% Functions 1/1 85.71% Lines 32/14

Press n or j to go to the next uncovered block, b, p or k for the previous block.

```
1 export default function (item) {
2   let model = {}
3   let state = item.transformedState || item.state // do a sprintf if necessary
4   model.label = `${item.label || item.name} [${state}]`
5   model.item = item.name
6   if (item.category) model.icon = item.category
7   model.type = 'Text'
8   if (!item.readonly) {
9     if (item.type === 'Switch') model.type = 'Switch'
10    if (item.type === 'Dimmer') model.type = 'Slider'
11    if (item.type === 'Rollershutter') model.type = 'Rollershutter'
12    if (item.metadata && item.metadata.semantics) {
13      if (item.metadata.semantics.value === 'Point_Setpoint') model.type = 'Setpoint'
14      if (item.metadata.semantics.value === 'Point_Measurement') model.type = 'Text'
15    }
16  }
17 }
18 return model
19 }
20 }
```

```
npm run test:unit
npm run test:unit:watch
```

```
npm run test:e2e
npm run test:e2e:gui
```



Cypress Test Runner

Tests: 2 passed, 0 failed, 5.62s

home page

- should have openHAB as <title>
 - BEFORE EACH
 - VISIT /
 - (XHR) GET 200 /rest/sitemaps
 - (XHR) GET 200 /rest/items?metadata=semantics
 - (XHR) GET 200 /sockjs-node/info?t=1558714408...
 - TEST
 - TITLE
 - ASSERT expected openHAB to include openHAB
- should have an "other apps" sidebar
 - BEFORE EACH
 - VISIT /
 - (XHR) GET 200 /rest/sitemaps
 - (XHR) GET 200 /rest/items?metadata=semantics
 - (XHR) GET 200 /sockjs-node/info?t=1558714408...
 - TEST
 - GET .right > .link > .icon
 - CLICK
 - GET .other-apps > .navbar > .navbar-inner
 - ASSERT expected <div.navbar-inner.sliding> to contain Other Apps

Application Screenshot: openHAB interface showing navigation menu and sidebar.

Try it!

Temporary repo with current code:

<https://github.com/ghys/openhab-ui-f7>

(PR to *openhab/openhab-webui* will follow when feature complete & stabilized)

Regularly deployed to:

<https://tiny.cc/ohdemo>

(be gentle! real demo openHAB instance behind - writing restricted w/o authentication)

—

DEMO!

See it in action:

<https://vimeo.com/342616345>