

# Experience of Test Automation Using UIVeri5 Framework

Anastasia Nesterova, Figaf  
June 17<sup>th</sup>, 2021



UI5con<sup>ON AIR</sup>

# Agenda

## Our way

- From thoughts to real results

## Advantages

- Pros that we've noticed during the usage

## Disadvantages

- Cons that we've noticed during the usage

# Experience of test automation using UIVeri5

## Our way

### Project

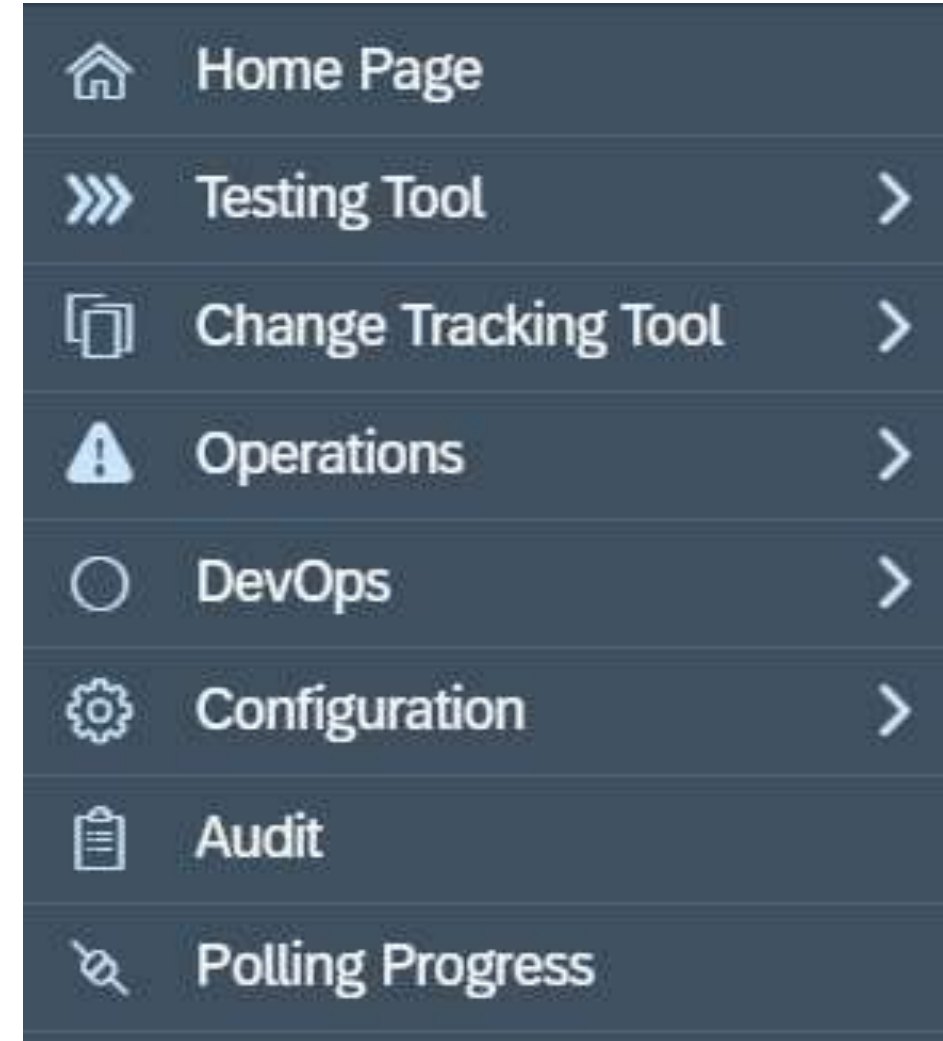
- The app has Java on backend and OpenUI5 on frontend
- The app interacts with SAP systems (PI/PO, CPI, Api Management) and provides features for:
  - Testing
  - Migration
  - Transporting
  - Changes tracking
  - Monitoring
  - ...
- One day we decided to implement automated end-to-end tests...

# Experience of test automation using UIVeri5

## Our way

### Goals

- Implement tests for:
  - Objects (users, agents) creation
  - Synchronization with real SAP systems
  - Testing Tool
    - Common flow with messages recording and tests running
  - Operations
    - Messages monitoring
    - Alerts polling
  - DevOps
    - Objects configuration and import

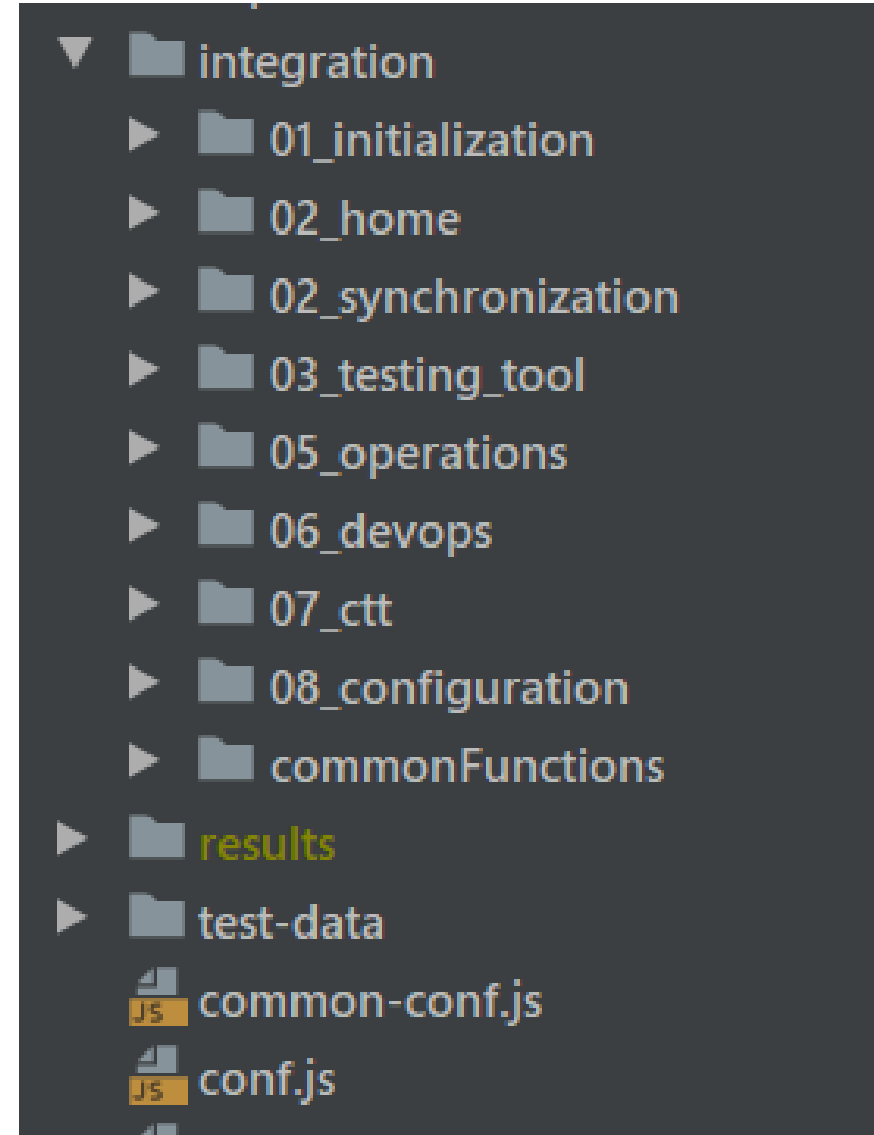


# Experience of test automation using UIVeri5

## Our way

### Results

- Implemented planned E2E flows
- Implemented some API tests
- Added tests running to CI
- Have more plans for further extensions



# Experience of test automation using UIVeri5

## Our way

Total statistics

Specs Total: 618

Passed: 567

Failed: 51

\_initLicense

\_initOauthClients

\_initUsers

initAgents

initConsumersAndConfigureRules

initLandscapes

initMessageMonitors

initScenarioMappings

syncAgents

initComparisonConfigurations

setIntObjectConfigurations

testing

initEdiMessageMonitors

ediMessageMonitoring

messageMonitor

sapSystemMonitoring

supportTool

\_initTransportConfigurations

createReleases

createTicketsAndAttachObjects

releases

tickets

Specs

Total: 4

Passed: 4

Failed: 0

Pending: 0

Disabled: 0

Duration: 28.3 sec

Authentication

Load invalid type

Load invalid uuid

Load license

Authentication

1. Execute 'sendKeys' on element with ID '\_\_xmlview1--idEmailInput-inner'

2. Execute 'sendKeys' on element with ID '\_\_xmlview1--idPasswordInput-inner'

3. Execute 'click' on element with ID '\_\_xmlview1--idLoginButton'

1. Execute 'sendKeys' on element with ID '\_\_xmlview1--idEmailInput-inner'
2. Execute 'sendKeys' on element with ID '\_\_xmlview1--idPasswordInput-inner'
3. Execute 'click' on element with ID '\_\_xmlview1--idLoginButton'

# Agenda

## Our way

- From thoughts to real results

## Advantages

- Interacts with UI5 controls
  - Get controls by binding properties
  - Provides possibility to get controls properties
- Can be used for different test levels:
  - E2E testing
  - API testing
- Can be easily configured
- Easy to start

## Disadvantages

- Cons that we've noticed during the usage

# Experience of test automation using UIVeri5

## Pros

### Interacts with UI5 controls

```
/** get button with property text */
const changeVersionButton_byProperty = element(by.control({
  controlType: "sap.m.Button",
  properties: [{text: "Change Version"}]
}));

/** get button by bindingPath */
const changeVersionButton_byBindingPath = element(by.control({
  controlType: "sap.m.Button",
  bindingPath: {
    modelName: "appView",
    propertyPath: "/bShowVersionSwitchInHeader"
  }
}));

/** get button by id */
const changeVersionButton_byId = element(by.id("sdk---app--changeVersionButton"));
```



# Experience of test automation using UIVeri5

## Pros

### Interacts with UI5 controls

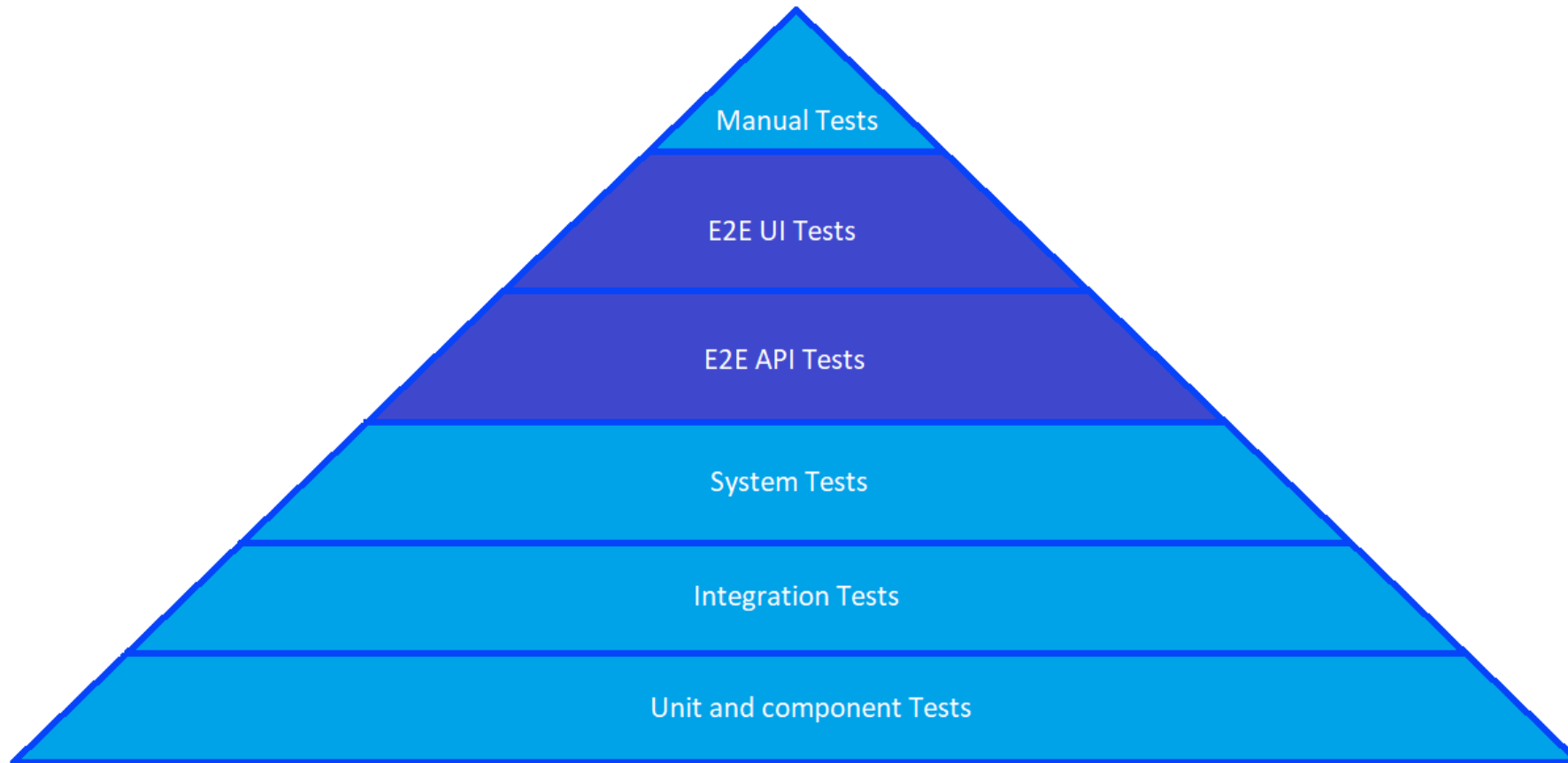
```
const changeVersionDialog = element(by.control({
  controlType: "sap.m.Dialog",
  properties: {title: "Change Version"}
}));
const versionStandardListItems = changeVersionDialog.all(by.control({
  controlType: "sap.m.StandardListItem",
  id: /-versionList-/
}));

// it isn't efficient but I want to show that we can work with lists of objects
// and use asControl().getProperty(<property>) to do something with property value (e.g. check it)
versionStandardListItems.then(standardListItems => {
  standardListItems.forEach(standardListItem => {
    standardListItem.asControl().getProperty( property: "title")
      .then(version => {
        if (version === "1.89.0") {
          standardListItem.click()
        }
      })
  })
});
```

# Experience of test automation using UIVeri5

## Pros

Can be used for different test levels



# Experience of test automation using UIVeri5

## Pros

### Can be easily configured

- `conf.js` file:
  - profile
  - baseUrl
  - specs
  - reports
  - timeouts
  - ...

```
const path = require('path');

exports.config = {
  profile: 'integration',
  baseUrl: 'https://openui5.hana.ondemand.com/api/',
  specs: [
    path.relative(process.cwd(), path.join(__dirname, '*', '*.spec.js'))
  ],
  reporters: {...},
  takeScreenshot: {...},

  timeouts: {getPageTimeout: '100000'...},

  connectionConfigs: {...},

  browsers: [{browserName: 'chrome'...}]
};
```

# Experience of test automation using UIVeri5

## Pros

### Easy to start

- Go to <https://github.com/SAP/ui5-uiveri5> and follow the installation guide
- Install UI5 Test Recorder (for Chrome) <https://chrome.google.com/webstore/detail/ui5-test-recorder/hcpkckcanianjcibiigbklddcpfiljmhj>
  - <https://www.youtube.com/watch?v=hNQ6E6eX1Dg> - UI5con@SAP 2019: UI5 Test-Recorder - A Point&Click Adventure for your UI5 Tests
- Record your first tests
- Select UIVeri5 language
- Download archive
- <https://www.youtube.com/watch?v=jlOUkRdNxO0> - UI5con@SAP 2019: E2E Testing with UIVeri5

# Agenda

## Our way

- From thoughts to real results

## Advantages

- Pros that we've noticed during the usage

## Disadvantages

- Doesn't support parallel tests running
- Sometimes tests can fail with `no UI5 on this page` error message
- It's a little bit tricky to work with localStorage items

# Experience of test automation using UIVeri5

## Cons

### Doesn't support parallel tests running

- <https://github.com/SAP/ui5-uiveri5/issues/250>



Open

Support parallel executions #250

adrianna-u opened this issue on Oct 14, 2020 · 1 comment

Thanks  
Adrianna



tsaleksandrova added the **question** label on Dec 14, 2020



maximnaidenov commented on Dec 15, 2020

Contributor



Hi Adriana,

Our initial scope is to support the parallelisation of the whole suite over a homogeneous set of browser instances. We have not defined the syntax yet but we will try to stick to protractor style: <https://github.com/angular/protractor/blob/master/lib/config.ts>  
With saucelabs, this will look like several parallel sessions started and specs will run on some of them. And of course, you can assign those sessions with build, job, tags and so achieve a good overview in the saucelabs dashboard.

We also have the "browser restart" feature in our backlog. It means we will have a flag that instructs the runner to start a fresh browser for every spec and close it when finished. With saucelabs, this would mean exactly a session per spec. Of course, the parallelisation will also work in this case.

# Experience of test automation using UIVeri5

## Cons

### Sometimes tests can fail with `no UI5 on this page` error message

- Error while waiting for Protractor to sync with the page: "waitForUI5: no UI5 on this page." happens from time to time
- It seems it stopped occurring for us after adding `browsers.capabilities` configuration in `conf.js` file

# Experience of test automation using UIVeri5

## Cons

It's a little bit tricky to work with localStorage items

```
/** Doesn't work */  
// localStorage.getItem("item");  
  
/** Works */  
browser.executeScript( script: `return window.localStorage.getItem("item");`)  
  .then( opt_callback: itemValue => {  
    // work with itemValue  
  })
```



# Agenda

## Our way

- From thoughts to real results

## Advantages

- Pros that we've noticed during the usage

## Disadvantages

- Cons that we've noticed during the usage

# Experience of test automation using UIVeri5

## Useful links

### Useful links

- Awesome tools to help you with test implementation:
  - UI5 Inspector (for Chrome) <https://chrome.google.com/webstore/detail/ui5-inspector/bebecogbafbigghaildooiibipcnbngo>
  - UI5 Test Recorder (for Chrome) <https://chrome.google.com/webstore/detail/ui5-test-recorder/hcpkckcanianjcibiigbklddcpfiljmhj>
- UIVeri5 tests configuration:
  - Authentication <https://github.com/SAP/ui5-uiveri5/blob/master/docs/config/authentication.md>
  - Reports <https://github.com/SAP/ui5-uiveri5/blob/master/docs/config/reporters.md>
- UIVeri5 locators <https://github.com/SAP/ui5-uiveri5/blob/master/docs/usage/locators.md>
- UIVeri5 API cheatsheet <https://gist.github.com/cvakiitho/55cae3cd31d20c85a1fc6f4c0ba277dc>
- UI5con@SAP 2019: E2E Testing with UIVeri5 <https://www.youtube.com/watch?v=jlOUkRdNxO0>
- UIVeri5 examples <https://github.com/AnastasyaNN/uiveri5-example>

# Thank you!

**Anastasia Nesterova**

Experience of test automation using UIVeri5 framework  
[ane@figaf.com](mailto:ane@figaf.com)



UI5con<sup>ON AIR</sup>