

Evolveum



Testing framework Schrodinger

Kateryna Honchar

June 2021

Agenda



- Introduction
- The history of GUI automated testing in Midpoint
- Schrodinger framework structure and capabilities
- Schrodinger framework usage
- Plans
- Questions and discussion

The history of GUI automated testing in Midpoint

- A need to support UI stability
- Some attempts with selenium and selenide libraries
- Schrodinger framework creation
- Moving Schrodinger to a separate project (<https://github.com/Evolveum/schrodinger>)

Schrodinger project structure

Schrodinger project consists of

- Schrodinger framework

 - Intuitive and understandable code

 - Assertion with screenshots

- Schrodinger tests

 - Rest-api client usage for objects import (addObjectFromFile(File file) method)

 - Configuration properties

 - A possibility to specify the location of extension schema file and post-initial import objects files

Assertion with screenshot

- AssertionWithScreenshot is an implementation of testing Assertion class
- assertion variable is available from every Schrodinger component
- every failure ends up with a screenshot

```
showUser("kirk")
```

```
.selectTabBasic()
```

```
.form()
```

```
.assertPropertyInputValueContainsText("Meal Card Number", "1001")
```

```
.and()
```

```
.and()
```

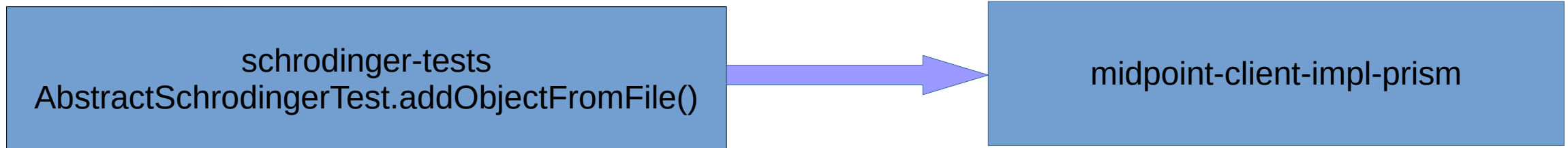
```
.selectTabProjections()
```

Error message: *The value of the input attribute Meal Card Number doesn't contain text '1001'*

Midpoint rest-api client

```
basicPage.importObject()  
  .getObjectsFromFile()  
    .chooseFile("file_path")  
    .checkOverwriteExistingObject()  
    .clickImportFileButton()  
  .feedback()  
  .assertSuccess();
```

The objects are added to the Midpoint using Midpoint rest-api client.



Schrodinger framework configuration

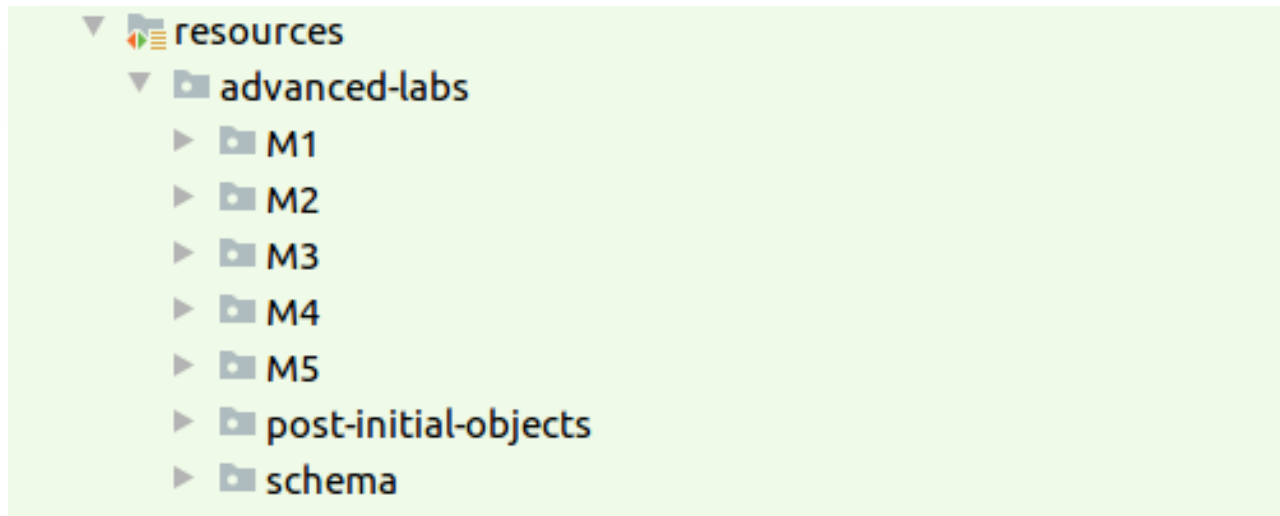
```
schrodinger.properties x
1  username=administrator
2  password=5ecr3t
3  base_url=http://localhost:8180/midpoint
4  base_url_mp_already_started=http://localhost:8080/midpoint
5  webdriver=CHROME
6  webdriverLocation=/opt/chromedriver
7  remoteWebDriverUrl=
8  useRemoteWebDriver=false
9  # By default if no value is specified for headless start then the value is: false
10 headlessStart=true
11 startMidpoint=true
12
```

Extension schema and post-initial objects import

Override the following methods to prepare Midpoint instance before tests running

```
protected File getExtensionSchemaFile() { return null; }  
protected String getPostInitialObjectsFolderPath() { return null; }
```

An example of the resource folder structure for Midpoint advanced training labs



Schrodinger code example

```
ListUsersPage usersPage = basicPage.listUsers();
```

```
usersPage
```

```
    .table()
```

```
        .search()
```

```
            .byName()
```

```
            .inputValue("jdoe")
```

```
            .updateSearch()
```

```
            .and()
```

```
        .clickByName("jdoe")
```

```
            .assertName("jdoe")
```

```
            .assertGivenName("john")
```

```
            .assertFamilyName("doe");
```

Schrodinger framework usage

Dynamic testing environment (Kubernetes, Jenkins) variants:

- Starting the tests starts midpoint instance, we use locale webdriver, no need in Midpoint instance preparation
- Prepare midpoint instance (extension schema file, post-initial objects), run Midpoint, then run the tests over already running Midpoint instance. Use remote webdriver.

Plans

- Further extension of Schrodinger framework
- Further coverage of the Midpoint UI with the tests
- Extend the testing environment variability

Discussion

Questions & Answers

Thank you for your time

See other talks at <https://docs.evolveum.com/talks>

Also **follow us** on our social media for further information!



/Evolveum



/Evolveum



/Evolveum



@Evolveum



/Evolveum

Evolveum