



WebRobot User Guide: Version 5.2

1	Introduction	3
2	Configuring the Service Tester for Web Robots	3
3	Basics on How a Web Robot Works	4
3.1	Terminology.....	5
3.1.1	Web Robot Status	5
3.1.2	Web Robot Tasks.....	5
3.1.3	Web Robot Test Cases.....	6
3.1.4	Web Robot Servers	7
3.2	Creating a New Web Robot Using Selenium IDE.....	8
3.2.1	Installation of Selenium IDE	8
3.2.2	Recording a Test Case using Selenium IDE.....	8

1 Introduction

OneView can be scheduled to run a predefined test scenario to make sure that for example an application is working. The test is either set up using a script designed in Selenium IDE or directly in the OneView application. The script is run via the OneView Service Tester extender. In order to use the Web Robot functionality a special Web Robot software package must be installed.

This document provides you with details on setting up a test in the Service Tester, an introduction to creating a test script in Selenium IDE, and finally how to test the script using the Service Tester.

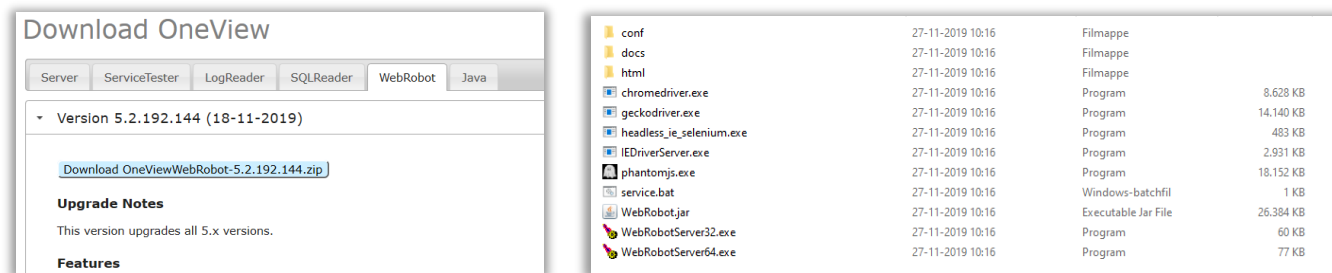
A Web Robot is a good tool of providing information on a workflow in an application and verifying that an application is working also during the night and weekends when there might not be any data in the log files.

Be aware that a Web Robot can be time consuming as it requires work to maintain – it is therefore important to keep it as simple as possible.

2 Configuring the Service Tester for Web Robots

For a Service Tester to be able to schedule robots, a Web Robot directory must already exist. A Web Robot directory is an installation directory containing the executable WebRobot.jar and corresponding configuration files.

The Web Robot software can be downloaded from the OneView Download Page - <https://monsalta.dk/oneview/>. Choose the tab WebRobot and unzip the contents into a folder in your OneView folder – e.g. 'D:\OneView\OneViewWebRobot':



The screenshot shows the 'Download OneView' page with the 'WebRobot' tab selected. Below the tabs, it shows 'Version 5.2.192.144 (18-11-2019)' and a link to 'Download OneViewWebRobot-5.2.192.144.zip'. Under 'Upgrade Notes', it says 'This version upgrades all 5.x versions.' Under 'Features', it lists various components. To the right, a file explorer window shows the contents of the downloaded zip file, including folders like 'conf', 'docs', 'html' and files like 'chromedriver.exe', 'geckodriver.exe', 'headless_ie_selenium.exe', 'IEDriverServer.exe', 'phantomjs.exe', 'service.bat', 'WebRobot.jar', 'WebRobotServer32.exe', and 'WebRobotServer64.exe'.

File/Folder	Date	Type	Size
conf	27-11-2019 10:16	Folder	
docs	27-11-2019 10:16	Folder	
html	27-11-2019 10:16	Folder	
chromedriver.exe	27-11-2019 10:16	Program	8,628 KB
geckodriver.exe	27-11-2019 10:16	Program	14,140 KB
headless_ie_selenium.exe	27-11-2019 10:16	Program	483 KB
IEDriverServer.exe	27-11-2019 10:16	Program	2,931 KB
phantomjs.exe	27-11-2019 10:16	Program	18,152 KB
service.bat	27-11-2019 10:16	Windows-batchfil	1 KB
WebRobot.jar	27-11-2019 10:16	Executable Jar File	26,384 KB
WebRobotServer32.exe	27-11-2019 10:16	Program	60 KB
WebRobotServer64.exe	27-11-2019 10:16	Program	77 KB

A configuration file is installed together with the Web Robot software – in the 'conf' directory. The first time you install the Web Robot software you need to rename the file from 'config.properties.default' to 'config.properties' in order to make the configuration work. Details on the various settings are documented in the file.

Prior to designing your first Web Robot you need to tell your Service Tester where to find the Web Robot software – this is performed in the Service Tester in the Admin module:

OneView ServiceTester Monsalta-BCH/Kursus

Version	5.2.202.2637 (64-bit)	Server	https://bch.oneview.eu	Trans. Last Hour	1774	Started At	2019-11-22 14:34:14
Build Date	2019-11-22 08:35	Last Delivery	17 sekunder siden	Avg. Delivery Time	1 ms	Current Time	2019-11-27 13:53:07

[All](#)
[Enabled](#)
[Disabled](#)
[Running](#)
[Errors](#)

[Admin](#)
[Stop](#)
[Start](#)
[Start Auto Refresh](#)

Search:
Type:
Cred:
[New Test](#)

Fill in the information in the tab 'Web Robot Servers' and point to where your Web Robot software is installed – in this case 'C:\OneView\OneViewWebRobot'. Remember to click 'Save'.

Administration

[Home](#)

[About](#)
[Registration](#)
[Test Credentials](#)
[Secrets](#)
[Thresholds](#)
[DNS](#)
[Jobs](#)
[Version History](#)
[Test Results](#)
[Web Robot Servers](#)

Configuration

Start web robot servers on demand ☒ [Save](#)

Default Web Robot Server Directory [Save](#)

Max. concurrent robots [Save](#)

3 Basics on How a Web Robot Works

A Web Robot is defined as a selection of test cases – this means that more tasks can be run in the same web robot.

From version OneView version 5.2 the Web Robot is primarily operated from OneView and not from the Service Tester.

In OneView the Web Robot is operated from the tab 'Measurements' and 'Web Robots'.

[Logout](#)
[Main Dashboard](#)
[My Dashboard](#)
[Reports](#)
[Measurements](#)
[Administration](#)
[Extenders](#)
[System](#)
[My Profile](#)

1•VIEW Measurements

[Web Robot Status](#)
[Web Robot Tasks](#)
[Web Robot Test Cases](#)
[Web Robot Servers](#)

List of all web robot instances and their status. ☒ Enabled only

ID	Status	Web Robot Task	Description	Service Tester	Scheduling	Last Run	Last Duration	Message	Remove	Disable	Test	History	
3	OK	Andelsboliginfo (1)		141	Monsalta-BCH	Every 45 seconds	< 1 minute ago	40,560 ms	OK	Remove	Disable	Test	History
5	OK	bch		141	Monsalta-BCH	Every 7 minutes, 30 seconds	5 minutes ago	59,216 ms	OK	Remove	Disable	Test	History
9	OK	bch - chrome - task/no user		141	Monsalta-BCH	Every 1 minutes	< 1 minute ago	21,561 ms	OK	Remove	Disable	Test	History
11	OK	bch - IE - task/no user		141	Monsalta-BCH	Every 15 minutes	12 minutes ago	8,021 ms	OK	Remove	Disable	Test	History
14	OK	bch copy - no task		141	Monsalta-BCH	Every 45 seconds	< 1 minute ago	21,552 ms	OK	Remove	Disable	Test	History
56	OK	Web Robot for test case basis2	Nothing yet.	141	Monsalta-BCH	Every 15 minutes	12 minutes ago	36,674 ms	OK	Remove	Disable	Test	History
57	OK	Web Robot for test case monsalta	Nothing yet.	141	Monsalta-BCH	Every 5 minutes	2 minutes ago	25,132 ms	OK	Remove	Disable	Test	History

3.1 Terminology

Several terms are used when working with the Web Robot. A brief introduction is presented below based on the naming of the tabs in OneView.

3.1.1 Web Robot Status

This tab contains a list of all web robot tasks and their status. From this window you can:

- **Remove** a Web Robot Task – this means that you remove the task from the extender – the task is still in the OneView System. If the task is not deployed to an extender you can use the **Deploy** button – which is visible if a task is not deployed.

44	-	Polycom	141	Monsalta-BCH	Every 10 minutes	-	-	No status available!	Remove
45	-	PrivacyKompasset	-	-	Every 15 minutes	-	-	-	Deploy
46	-	Random	141	Monsalta-BCH	Every 1 minutes	-	-	No status available!	Remove

- **Enable** a Web Robot Task – this means enabling the Web Robot Task to start running and generating data to your OneView system. If a task is already enabled, you can **Disable** the Web Robot Task. Note – if a Web Robot Task is already enabled you are presented with a color in the first column indication that status of the Web Robot Task

3	OK	Andelsboliginfo (1)	141	Monsalta-BCH	Every 45 seconds	< 1 minute ago	40.578 ms	OK	Remove	Disable	Test	History
4	-	Ansejning om våbentilladelse	141	Monsalta-BCH	Every 10 minutes	-	-	No status available!	Remove	Enable	Test	History

- **Test** a Web Robot Task – this enables you to run a test on your Web Robot Task. The test can be performed on both enabled and disabled Web Robot Tasks.

3	OK	Andelsboliginfo (1)	141	Monsalta-BCH	Every 45 seconds	< 1 minute ago	40.578 ms	OK	Remove	Disable	Test	History
4	-	Ansejning om våbentilladelse	141	Monsalta-BCH	Every 10 minutes	-	-	No status available!	Remove	Enable	Test	History

- **History:** The history of a Web Robot Task leads you to the Robot Log from where you can view details on the history of the Web Robot Task. Each run is saved in a separate log file.

3	OK	Andelsboliginfo (1)	141	Monsalta-BCH	Every 45 seconds	< 1 minute ago	40.578 ms	OK	Remove	Disable	Test	History
4	-	Ansejning om våbentilladelse	141	Monsalta-BCH	Every 10 minutes	-	-	No status available!	Remove	Enable	Test	History

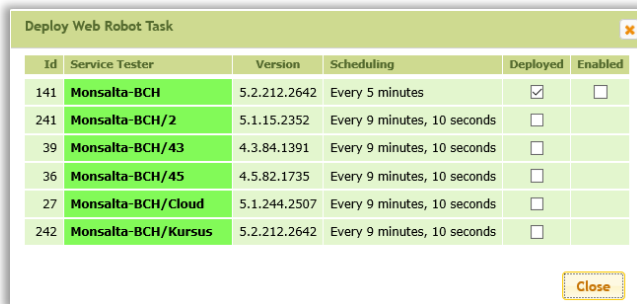
3.1.2 Web Robot Tasks

This tab contains information on all Web Robot Tasks contained in your OneView system. As mentioned earlier more Web Robot Test Cases can be contained in a Web Robot Task. In this section you can view information on the name of the task, which browser type is used for the task and the default scheduling.

From this window you can:

- **Deploy / Remove:** By clicking the **Deploy / Remove** button you can deploy or remove your Web Robot Task to/from a Service Tester. OneView automatically identifies Service Testers registered and shows a list from which you can choose the relevant action. Please note that you can also **Enable** the Web Robot Task in the dialogue box. Enabling means that you enable the task for measurements in the Service Tester. By unclicking

the button, you disable the Web Robot Task.



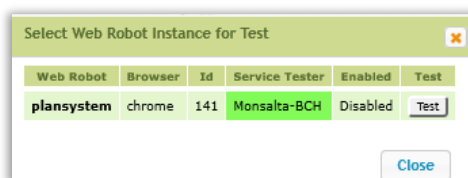
Id	Service Tester	Version	Scheduling	Deployed	Enabled
141	Monsalta-BCH	5.2.212.2642	Every 5 minutes	<input checked="" type="checkbox"/>	<input type="checkbox"/>
241	Monsalta-BCH/2	5.1.15.2352	Every 9 minutes, 10 seconds	<input type="checkbox"/>	<input type="checkbox"/>
39	Monsalta-BCH/43	4.3.84.1391	Every 9 minutes, 10 seconds	<input type="checkbox"/>	<input type="checkbox"/>
36	Monsalta-BCH/45	4.5.82.1735	Every 9 minutes, 10 seconds	<input type="checkbox"/>	<input type="checkbox"/>
27	Monsalta-BCH/Cloud	5.1.244.2507	Every 9 minutes, 10 seconds	<input type="checkbox"/>	<input type="checkbox"/>
242	Monsalta-BCH/Kursus	5.2.212.2642	Every 9 minutes, 10 seconds	<input type="checkbox"/>	<input type="checkbox"/>

- **Test:** By clicking the **Test** button you can test your Web Robot Task. A Web Robot Task can be tested in both states 'Enabled' and 'Disabled'.
- **Delete:** By clicking the **X** you can delete your Web Robot Task. If Web Robot Test Cases are attached to the Web Robot Task you are asked whether you wish to delete. Please note that the Web Robot Test Cases are not deleted from OneView in this step – only the Web Robot Task in question.

3.1.3 Web Robot Test Cases

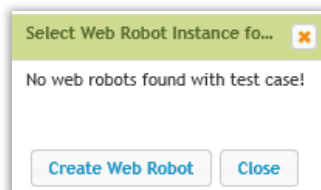
This tab contains information on all Web Robot Test Cases contained in your OneView system. Web Robot Test Cases are step-by-step instructions to a web browser simulating actions made by a real end-user. In this section you can view information all Web Robot Test Cases contained in your OneView installation as well as options to import and create a new Web Robot Test Case.

- **Version:** The **version** indicated which version is currently in use. Note – you can revert to previous versions by entering the Web Robot Test Case and saving this version. Older versions can also be deleted.
- **Test:** By clicking the **Test** button you can test your Web Robot Test Case using a Web Robot. The Web Robot Test Case needs to be connected to a Web Robot prior to testing.
 - If the Web Robot Test Case **is** already connected you are presented with a list of where the Web Robot Test Case is connected, and you click 'Test'. Note – if more Web Robot Test Cases are connected all will be run:



Web Robot	Browser	Id	Service Tester	Enabled	Test
plansystem	chrome	141	Monsalta-BCH	Disabled	<input type="button" value="Test"/>

- If the Web Robot Test Case **is not** connected you are presented with a dialogue box where you can create a new Web Robot Task:



Select Web Robot Instance fo... **X**

No web robots found with test case!

- **Delete:** By clicking the **X** you can delete your Web Robot Test Case.

3.1.4 Web Robot Servers

This tab contains the list of registered Web Robot Servers. A Web Robot Server handles the execution of Web Robot Tasks scheduled by a Service Tester. Starting and stopping Web Robot Servers are done by the Service Tester.

Web Robot Servers									
List of registered web robot servers. A web robot server handles execution of web robot tasks scheduled by a service tester. A web robot handles execution of one or more test cases in a real web browser.									
Hostname	Port	Default Browser	OS	Version	Type	Username	Uptime	Service Tester	Server Console
Monsalta-BCH	1302	Google Chrome	Windows 10	5.2.213.156	AdHoc	bolette	15 hours, 7 minutes	Monsalta-BCH	Server Console
Monsalta-BCH	1305	Google Chrome	Windows 10	5.2.213.156	AdHoc	-	11 hours, 22 minutes	Monsalta-BCH	Server Console

By clicking the **Server Console**, you are led to the log file for the Web Robot Servers. The log file contains information on running and terminated Web Robots and the logs for each run can be opened from here.

OneView Web Robot Server

Host and Port	Monsalta-BCH:1301	Server Version	5.2.213.156
Server Id	b17d8bce-3d6c-458b-b8f0-55a6d80cb64f	Java Version	13.0.1+9
Service Tester	http://127.0.0.1:12359	Selenium Version	3.141.59
Process	Id:33240 User:system	Chrome Driver Version	78.0.3904.70
Started At	2019-12-04 10:14:30		
Memory Used	80 / 4054 Mb		
OS Name	Windows 10		
Directory	C:\OneView\OneView\WebRobot		
Java Runtime	OpenJDK Runtime Environment (13.0.1+9, 64-bit, c:\OneView\openjdk-13.0.1\jdk-13.0.1)		

Running Robots

Robot Id	Started At	Driver	Browser	Source	Elapsed	Log
6	2019-12-04 10:16:24	chrome	No browser started!	ServiceTester@Monsalta-BCH bch - chrome - task:no user	7 seconds	Log

Terminated Robots

Robot Id	Started At	Driver	Browser	Source	Duration	Log	Exit Code	Message
3	2019-12-04 10:15:56	chrome	chrome 78.0.3904.108	ServiceTester@Monsalta-BCH bch - chrome - task:no user	27 seconds	Log	0	

3.2 Creating a New Web Robot Using Selenium IDE

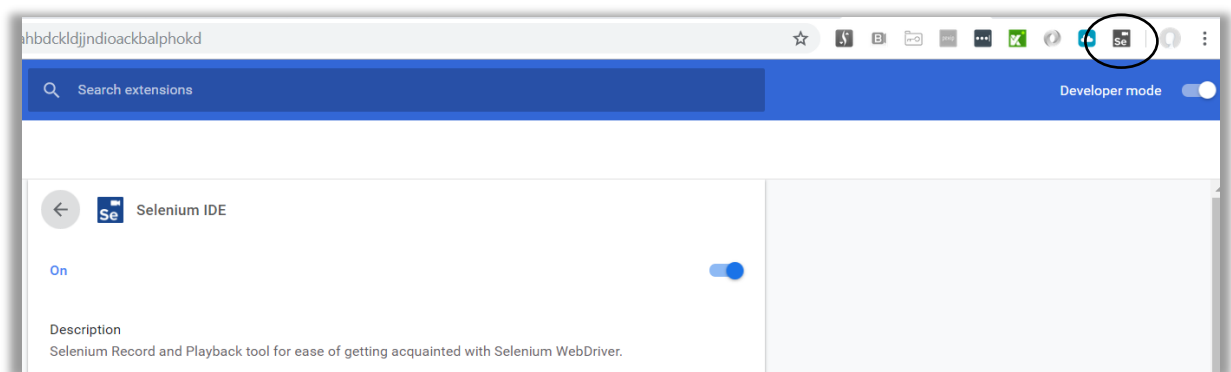
An example of creating a completely new Web Robot Task containing a Web Robot Test Case is described below:

3.2.1 Installation of Selenium IDE


Selenium IDE is the tool used for recording a new Web Robot Test Case to be used in OneView.

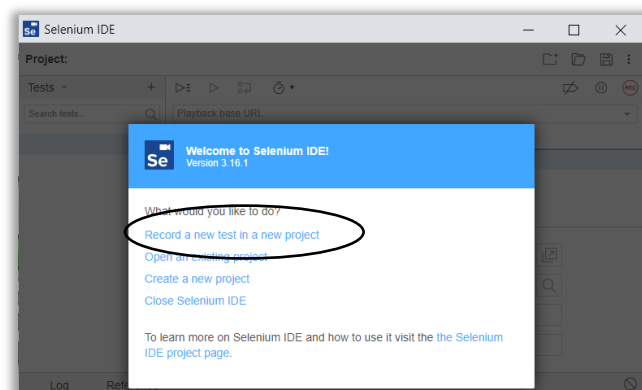
- Install Selenium IDE for Chrome using the following link:
<https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd>

It is added as an extension in Chrome upon installation and a Selenium icon is installed in the topbar of your Chrome browser:

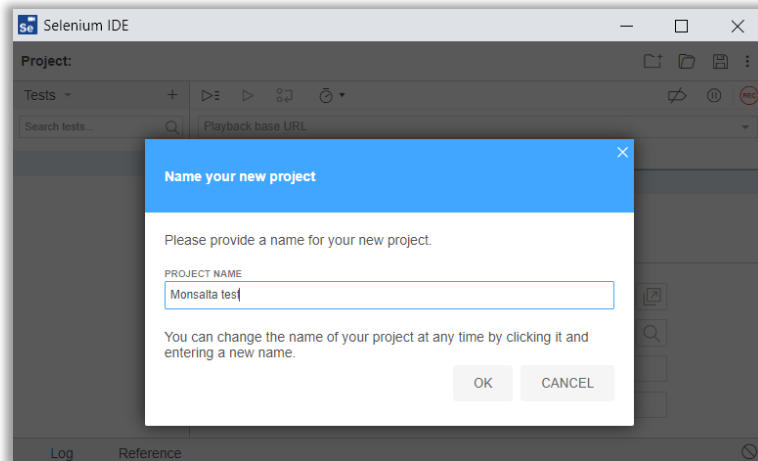


3.2.2 Recording a Test Case using Selenium IDE

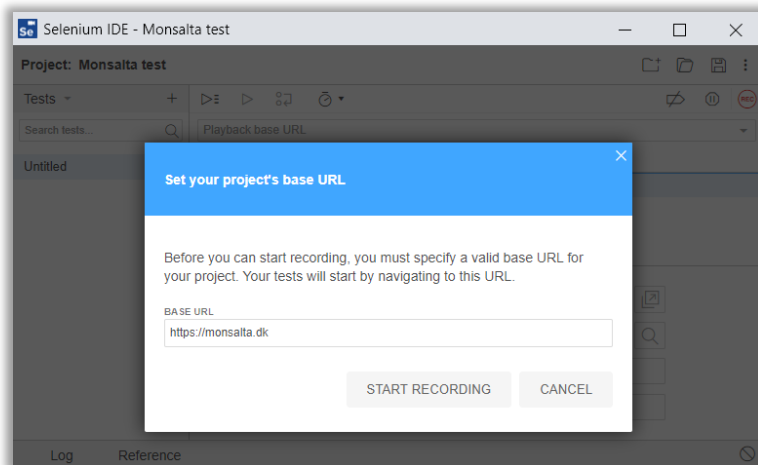
1. Open Selenium IDE in Chrome by clicking the icon in Chrome - . Pick the option 'Record a new test in a new project':



2. Provide a name to the project

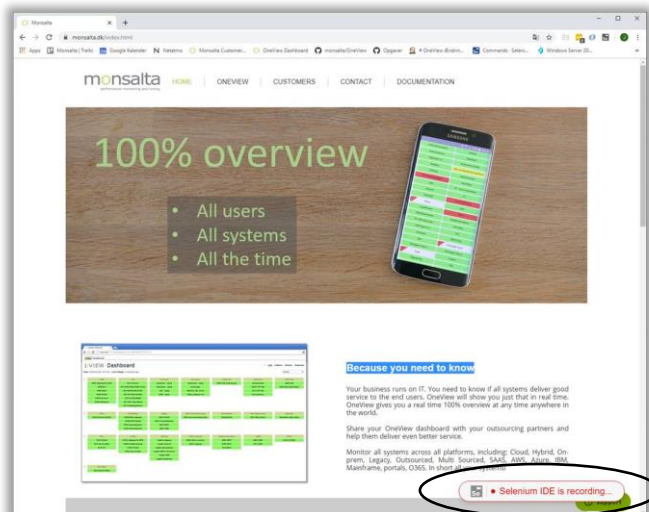


3. Set the base URL for the new test and start recording. The base URL is the URL where the test starts

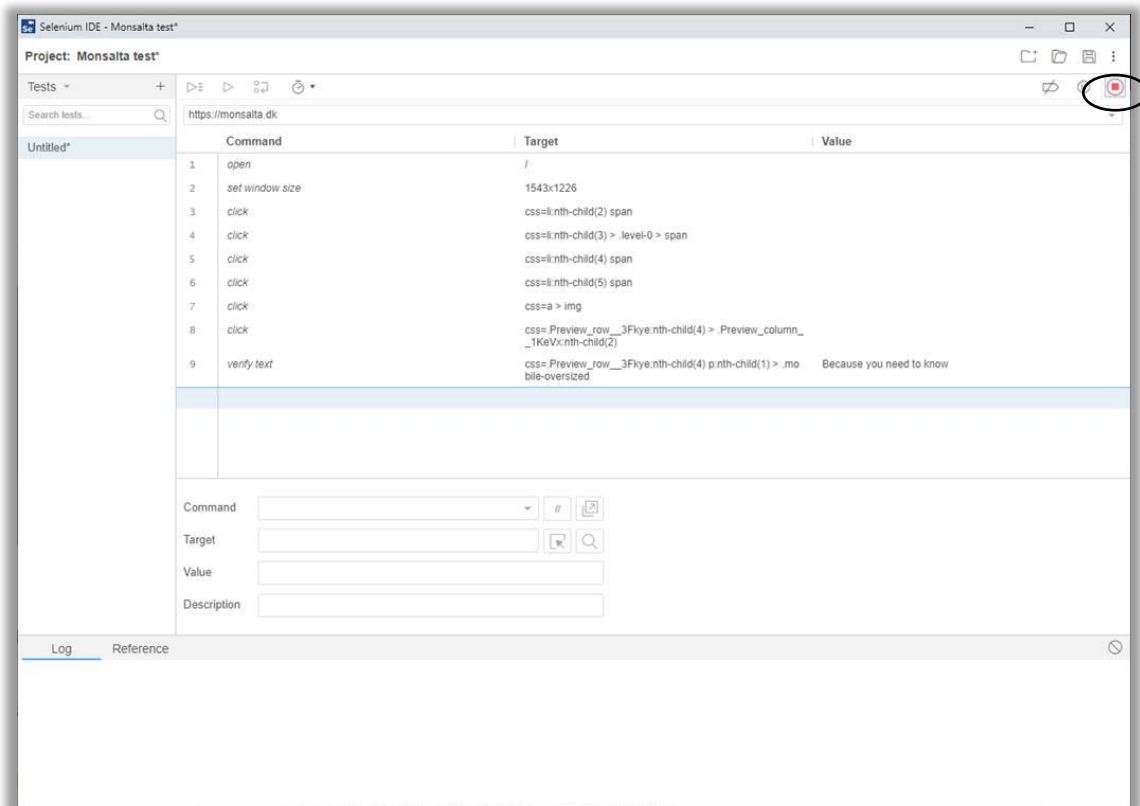


4. The webpage stated opens and from here all clicks, and other actions are recording in the background. Note the recording information at the bottom of the screen.

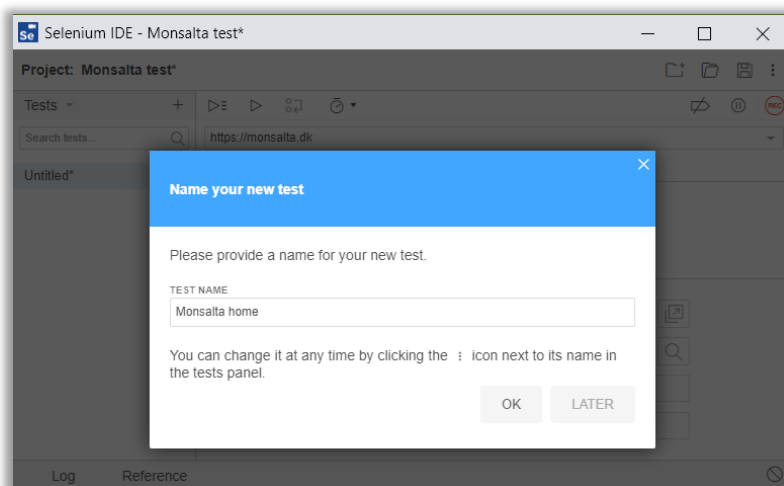
The test case that we are recording in this example opens the Monsalta web site, clicks the tabs at the top and verifies some text.




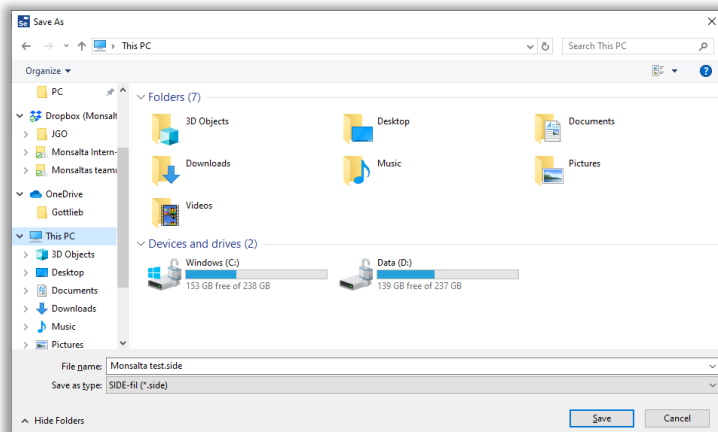
- When you have finished your test in the Chrome browser you need to stop the recording in the Selenium IDE. This is done by clicking the 'Stop Recording' button (the red square at upper right corner)




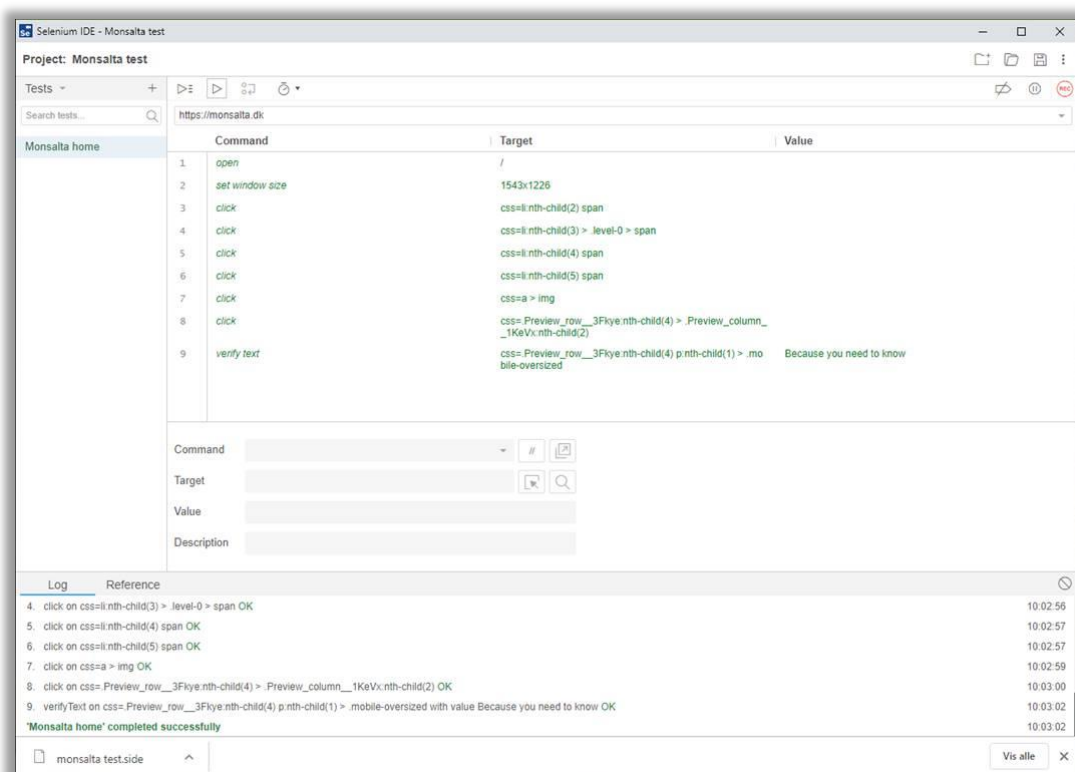
- Provide a name to the new test case and click 'OK'



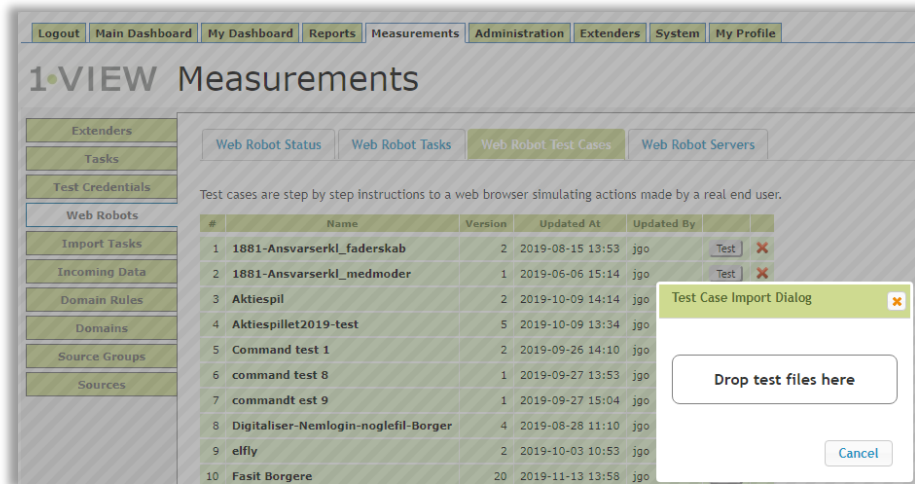
7. Save test case as a .SIDE file which can be used in OneView. Saving is done by clicking the 'Save' button in the upper right corner 



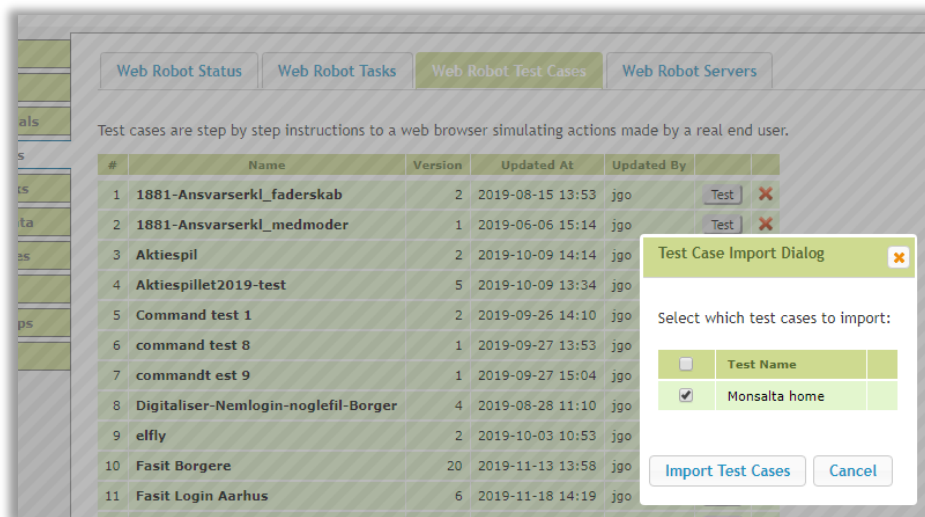
8. Perform a test run of the recorded test case by clicking the 'Run Current Test Case' icon in the top bar . The lines turn green to indicate that the test run was successful.



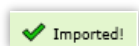
- Import the test case to OneView – using the path: Measurements -> Web Robots -> Web Robot Test Cases -> Import Test Case. Simply drag and drop the .SIDE file into the box.



- Select the test case to import to OneView by selecting the test case and clicking 'Import Test Cases'.



Upon a successful import the following icon is shown:



11. To view the imported test case simply click the name in OneView.

Edit Test Case

Name: Monsalta home
Id: 58fed74-382c-4f70-b5f9-7b00330af731
Version: Version 1 updated on 2019-11-26 10:16:30 by jgo

#	Command	Target	Value		
1	open	https://monsalta.dk/		✗	+
2	setWindowSize	1543x1226		✗	+
3	click	css=li:nth-child(2) span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
4	click	css=li:nth-child(3) > .level-0 > span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
5	click	css=li:nth-child(4) span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
6	click	css=li:nth-child(5) span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
7	click	css=a > img		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
8	click	css=.Preview_row_3Fkye:nth-child(4) > .Preview_column_		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
9	verifyText	css=.Preview_row_3Fkye:nth-child(4) p:nth-child(1) > .mob	Because you need to know	✗	+

Export (OneView) Export (Selenium IDE) Copy Save Close

From here it is possible to edit Web Robot commands and insert Oneview commands. OneView commands are identified by @OneView and are used by OneView to start and stop transactions used in OneView.

To add a new command click the '+' and choose your command from the drop-down box. Click 'Save' to save the test case with your changes.

Edit Test Case

Name: Monsalta home
Id: 58fed74-382c-4f70-b5f9-7b00330af731
Version: Version 3 updated on 2019-11-26 11:35:33 by jgo

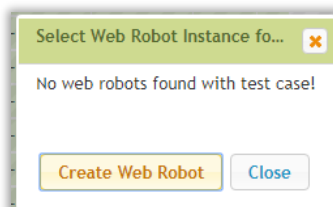
#	Command	Target	Value		
1	open	https://monsalta.dk/		✗	+
2	setWindowSize	1543x1226		✗	+
3	@OneView	TransactionBegin	Open oneview page	✗	+
4	click	css=li:nth-child(2) span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
5	@OneView	TransactionEnd		✗	+
6	@OneView	TransactionBegin	Open customers page	✗	+
7	click	css=li:nth-child(3) > .level-0 > span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
8	@OneView	TransactionEnd		✗	+
9	click	css=li:nth-child(4) span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
10	click	css=li:nth-child(5) span		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
11	click	css=a > img		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		
12	click	css=.Preview_row_3Fkye:nth-child(4) > .Preview_column_		✗	+
	Opens New Window	New Window Handle Name	Window Timeout		

Export (OneView) Export (Selenium IDE) Copy Save Close

12. The next step is to test the test case in a Web Robot. Simply click the 'Test' button next to your newly created test case

19	Monsalta home	1	2019-11-26 10:16	jgo	Test	X
----	---------------	---	------------------	-----	------	---

Since the test case is not connected to an existing Web Robot a new Web Robot has to be created – click 'Create Web Robot'.



Set up the new Web Robot Task. You are presented with a lot of parameters but in this case simply use the default parameters including the name for the Web Robot Task. Save the task by clicking 'Save'.

Web Robot Task

Task Id

webrobot_20191126_1019_1

Saved At

2019-11-26 10:19:57

Name

Web Robot for test case Monsalta home

Description

Nothing yet.

Browser

Google Chrome

Width

0 px

Height

0 px

Headless

☒

Robot Options

Timeout

600 seconds

Take Screenshots

☒

Seed Test

☐

iterations

1

Test Case

Timeout

600 seconds

Transaction Prefix

Test Case Step

Timeout

40 seconds

Delay

0 milliseconds

Windows Options

Run As Task

☐

Run When Logged In Only

☐

Windows Credentials

Default Schedule

Interval

every

900 seconds.

Test Case Name

Version

Updated By

Monsalta home

1

jgo

X

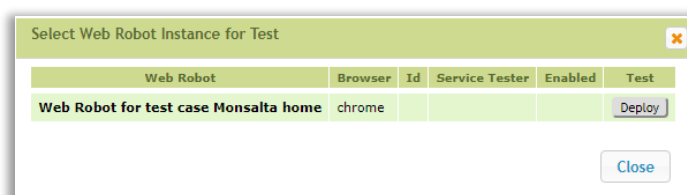
Add Test Case

Copy

Save

Close

13. The next step is to deploy the task – to deploy a task means that you connect it to a Service Tester.



Click 'Deploy' and choose the Service Tester you wish to deploy the Web Robot Task to.

In this step we recommend that you **do not enable** the Web Robot Task since you need to verify it before enabling it to run. Click 'Close'

Deploy Web Robot Task

Id	Service Tester	Version	Scheduling	Deployed	Enabled
1248	JGO-X1	5.2.207.2639	Every 15 minutes	<input type="checkbox"/>	
1249	JGO-X1/2	5.1.194.2466	Every 15 minutes	<input type="checkbox"/>	
1250	JGO-X1/3	5.1.113.2431	Every 15 minutes	<input type="checkbox"/>	
1252	JGO-X1/Cloud	5.2.195.2636	Every 15 minutes	<input type="checkbox"/>	
1253	JGO-X1/monitis	5.1.17.3	Every 15 minutes	<input type="checkbox"/>	

Close

	Id	Service Tester	Version	Scheduling	Deployed	Enabled
1248	JGO-X1	5.2.207.2639	Every 15 minutes	<input checked="" type="checkbox"/>		<input type="checkbox"/>
1249	JGO-X1/2	5.1.194.2466	Every 15 minutes	<input type="checkbox"/>		
1250	JGO-X1/3	5.1.113.2431	Every 15 minutes	<input type="checkbox"/>		
1252	JGO-X1/Cloud	5.2.195.2636	Every 15 minutes	<input type="checkbox"/>		
1253	JGO-X1/monitis	5.1.17.3	Every 15 minutes	<input type="checkbox"/>		

Close

14. Now the Web Robot Test Case is connected to a Web Robot Task and you can test it. Click 'Test' and the Web Robot Task including the test case will run.

Web Robot	Browser	Id	Service Tester	Enabled	Test
Web Robot for test case Monsalta home	chrome	1248	JGO-X1	Disabled	<input type="button" value="Test"/>

Close

The Web Robot Log is opened, and it is possible to view the steps performed in the Web Robot. Screenshots are shown telling you what is tested. Once the Web Robot is finalized and tested OK you can enable it and add the relevant transactions in your OneView installation.

11:26:03.709	0 ms	Monsalta home -	Info	Step 2: Status=CONTINUE	
11:26:03.463	246 ms	Monsalta home -	Info	Step 2: command=setWindowSize, target=1543x1226	
11:26:03.462	1 ms	Monsalta home -	Info		
11:26:02.285	1.177 ms	Screenshot	Info		