Een

Workshop

*Image Recognition* met

**SikuliX**

(in een framework)

# Installatie framework

## Java

De Java Development Kit is de software die een programmeur nodig heeft om programma's in Java te kunnen ontwikkelen. De Java Development Kit bestaat uit de Java Runtime Environment (JRE, alles wat nodig is om Java-programma's op een computer te kunnen draaien) met daarbij een reeks hulpmiddelen (tools) die voor de programmeur van belang zijn.

#### Installatie JDK

1. Ga naar: <https://tinyurl.com/puhhbtr> (<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html> )
2. Vink de Accept License Agreement en download de JAVA SE Development Kit 8u172 door de licentie te accepteren en op de 64 of 86 bits versie te klikken.
3. Open het installatie bestand van JDK. Onthoud de installatie locatie wanneer je de standaard installatie uitvoert.

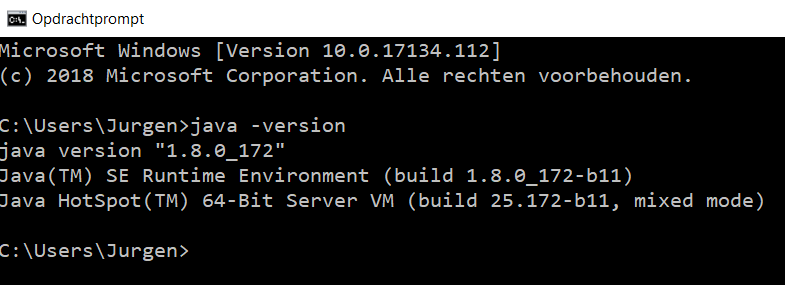
We downloaden hier versie **8** van de JDK. Dit is belangrijk, want versie 9 werkt nog niet in het framework. Houd er rekening mee dat versie 8 in Java jdk1.8 wordt genoemd. In dit geval: < C:\Program Files\Java\jdk1.8.0\_131\ > (de laatste 3 cijfers kunnen anders zijn bij een nieuwere versie, dat is geen probleem) 

#### Nadat de installatie is afgerond dienen de "Environment variables" goed te worden gezet.

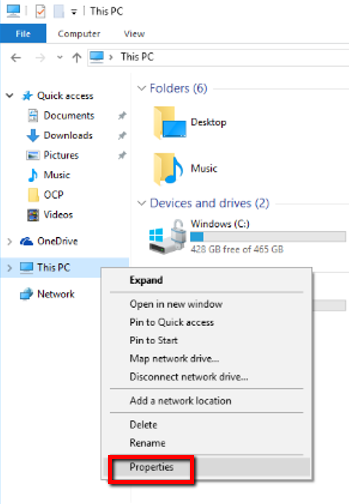
De environment variables ofwel de omgevingsvariabelen bevatten informatie voor de pc om te begrijpen waar bepaalde zaken zijn geïnstalleerd. PATH is de systeemvariabele die door het besturingssysteem wordt gebruikt voor het zoeken van benodigde uitvoerbare bestanden vanaf de opdrachtregel of vanuit een terminalvenster/commandprompt.

Windows **10**

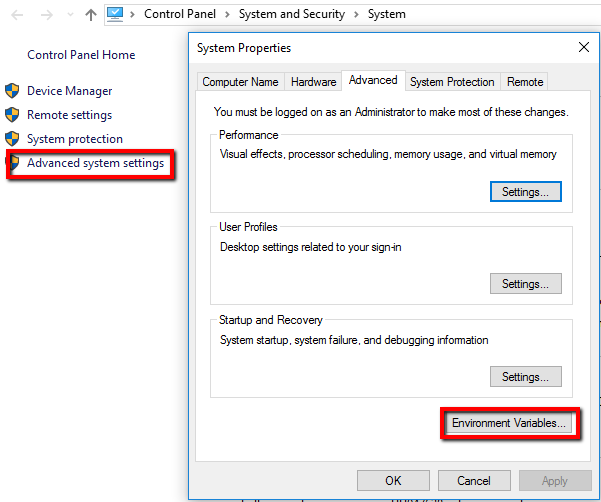
1. Ga naar de Properties van de Computer
2. Klik op Advanced system settings
3. Ga naar de tab Advanced
4. Klik hier op de knop Environment Variables
5. Maak een nieuwe Systeemvariabele aan door op New te klikken (LET OP! Zet hem bij je systeemvariabelen en niet je gebruikersvariabelen)
6. Geef deze als Name JAVA\_HOME en als Value de locatie (in dit voorbeeld dus < C:\Program Files\Java\jdk1.8.0\_131 >) waar JAVA is geïnstalleerd
7. Selecteer vervolgens de variable Path en klik op Edit
8. Klik op nieuw en voeg de volgende waarde toe: %JAVA\_HOME%\bin\



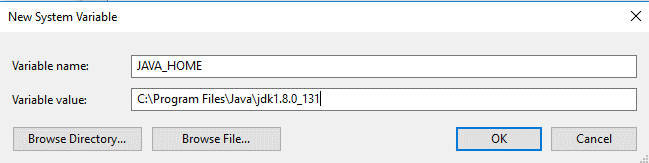
Kijk nu of de installatie van Java is gelukt door in je commandline “java -version” in te tikken. Je commandline/opdrachtprompt kan je benaderen door bij te zoeken naar “cmd”.



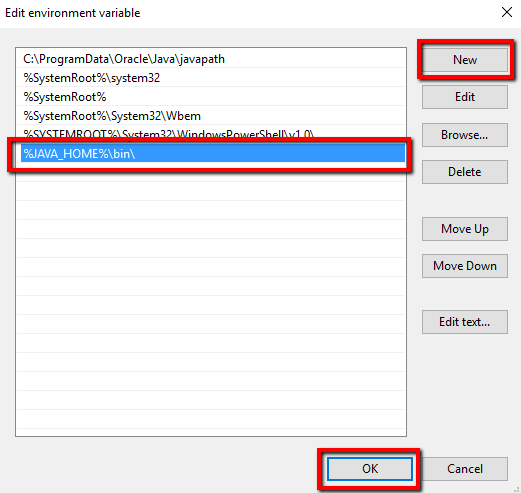
Stap 1&2



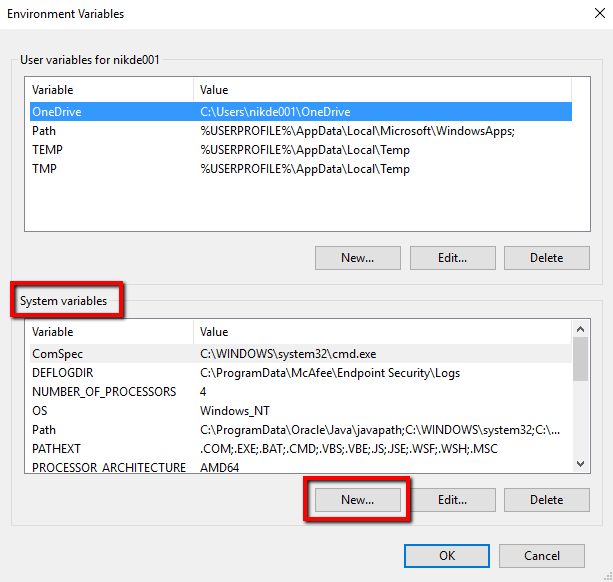
Stap 3&4



Stap 6

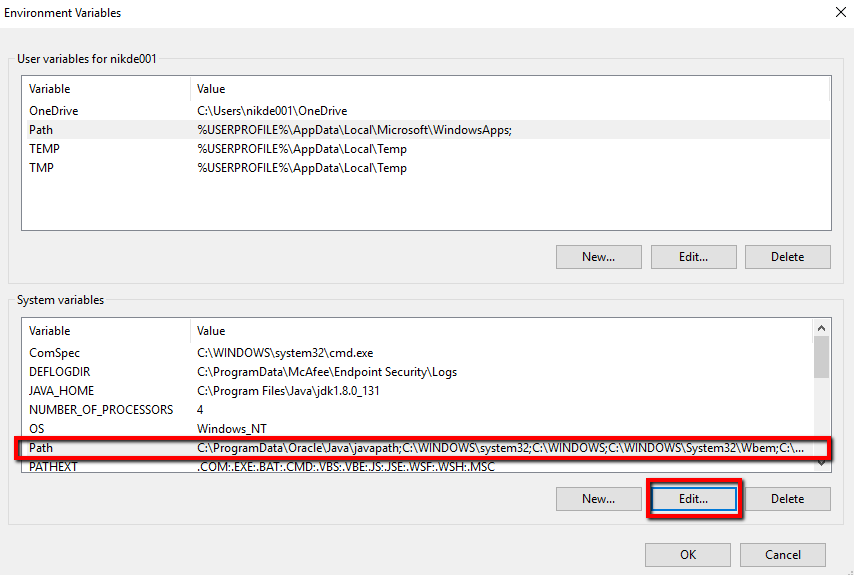


Stap 8



Stap 5

*Stap 7*



## SikuliX standalone

Sikuli is image recognition software. Waar Selenium het gebrek heeft dat het alleen in een browser gebruikt kan worden, kan je sikuli overal gebruiken als het een GUI heeft. Het wordt dus niet gebruikt voor op systemen die geen grafische schil hebben. Bijvoorbeeld als je een server of een database wilt testen. Ook is het niet geschikt voor performance en load testen. Aan de andere kant is het wel zeer geschikt als je een applicatie wilt testen waar je niet bij de broncode kan komen of als je de gebruikershandelingen preciezer wilt simuleren.

Als je bijvoorbeeld kijkt naar een selenium dan zal deze door een pagina heen klikken ondanks dat een element wellicht helemaal niet meer op de goede plek op een website staat. Dit maakt dat sikuli in dat opzicht nauwkeuriger is, maar ook weer meer onderhoud vergt.

We beginnen met de standalone tool om hier een testgeval mee te maken.

1. Download de sikulixsetup.jar op de volgende website: <https://tinyurl.com/yclvyuxk> (<https://launchpad.net/sikuli/sikulix/1.1.2>)

2. Maak een folder aan op je c-schijf waar je het .jar bestand naar toe verplaatst of waar je het in download

3. Dubbel klik op het jar bestand, werkt dit niet voer dan het volgende commando in cmd in: “java -jar sikulixsetup\*.jar” waarbij het sterretje voor de juiste versie staat.

4. Kies optie 1 waarbij je Jython gebruikt en klik verder door het menu heen

5. Als sikuli geinstalleerd is kan je het runsikulix.cmd bestand aftrappen door te dubbelklikken, of door het vanaf cmd als letterlijke waarde in te voeren vanaf de plek waar je het geïnstalleerd hebt.

Voor meer informatie kan je naar <http://www.sikulix.com/quickstart/> gaan

#### Opdracht

Scenario voor SikuliX:

1) Start IE op via de Windows knop  (Tip: je kan het standaard knipprogramma gebruiken om afbeeldingen te maken)

2) Navigeer naar internet opties (klik op tandwiel)

3) Ga daar naar internet options

4) Klik op delete

5) Klik alle vinkjes uit (door middel van een soort loop zie: <https://tinyurl.com/ycb3rukc>

(<http://sikulix-2014.readthedocs.io/en/latest/tutorials/uncheckall/uncheckall.html> )

(Tip: Als je op een afbeelding klikt in de standalone app dan kan je in de popup naar het tweede tabblad [Overeenkomen Voorbeeld]. Probeer nu eens met het schuifje heen en weer te bewegen.)

6) Valideer resultaat

7) Sluit de browser weer af

Gaat dit je makkelijk af? Misschien kan je al een scenario bedenken wat voor jou nuttig kan zijn?

## Maven

Apache Maven is een uitbreiding voor Java. Het zorgt er voor dat je afhankelijkheden van je project makkelijk op een centrale plek binnen kan halen. Maven gebruikt hiervoor het "Project Object Model" (pom.xml). In het pom.xml bestand staan deze afhankelijkheden (broncode van andere projecten die wij gebruiken). Daarnaast kan je hier nog meer informatie opslaan. Dit bestand is dan ook het centrale document voor maven om vanuit te opereren. Met maven kan nog meer, maar daar gaan we nu niet op in.

#### Installatie

Kijk eerst of je Maven al geïnstalleerd hebt door in je commandline “mvn --version” in te tikken.

Mocht dit het geval zijn dan kan je dit onderdeel overslaan.

1. Ga naar: <https://maven.apache.org/download.cgi>
2. Selecteer de Link van de Binary zip archive
3. Pak het zojuist gedownloade ZIP-bestand uit in de java folder zodat het pad als volgt wordt: C:\Program Files\Java\apache-maven-3.5.0\
4. Nu dient (net zoals bij de JDK) de Environment Variable te worden gezet. Voeg de Maven omgevingsvariabelen met de naam MAVEN\_HOME en de waarde gevuld met de folder waar Maven staat.
5. Doe hetzelfde met de variabele Path: %MAVEN\_HOME%\bin\  (voor Windows 10)

## Eclipse IDE

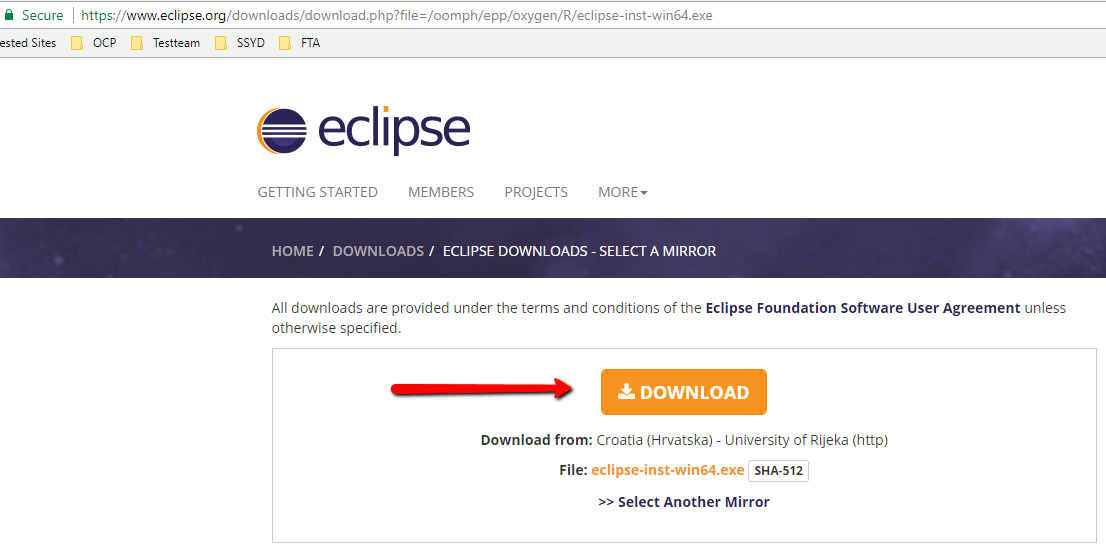
Eclipse IDE (integrated development environment) is een open source ontwikkeltool voor het bouwen van Java applicaties. In Eclipse schrijf je dus de code van je applicatie, zowel de broncode als ook de testcode. Alternatieven zijn Eclipse zijn IDE’s zoals NetBeans, IntelliJ, etc. Een paar kenmerken:

• Open source

• Veel plugins (gratis en betaald)

• Door plugins te gebruiken kan je in verschillende talen programmeren

#### Installatie

1. Ga naar: [http://www.eclipse.org](http://www.eclipse.org/) en download Eclipse installer
   1. Klik op de download button (2 keer achter elkaar)
2. Zodra deze gedownload is, voer het bestand uit, en klik tijdens de installatie op Eclipse IDE for Java Developers
3. Volg het standaard installatie proces
4. NOTE: Houd er rekening mee dat je de installatie van de JDK al afgerond hebt voordat je met Eclipse IDE gaat werken. Zonder de JDK kan je Eclipse IDE niet opstarten.

Nu je alle randvoorwaarden hebt geinstalleerd kan je Eclipse opstarten.

Bij de popup met de vraag waar je je workspace op wilt slaan kies je een willekeurige directory en klik op OK.

Voor het volgende onderdeel van de workshop gaan we sikuli in Eclipse gebruiken. Daarvoor gebruiken we een maven project. Klik het startscherm weg door de tab te sluiten in de linkerbovenhoek. Klik daarna op New -> Other -> Maven project.

- Gebruik de default workspace location en klik nogmaals op Next.

- Selecteer maven-archetype-quickstart (deze is meestal al voorgeselecteerd) en klik op Next.

- Geef de group id en artifact id op. Voor nu is dat niet heel belangrijk, je kan hier bijvoorbeeld kiezen voor sikulitest en sikuli.

- Als je op finish klikt wordt er een project voor je aangemaakt.

- Wat belangrijk is, is dat je JRE System Library op dezelfde versie staat als de JDK die jij geinstalleerd hebt. Is dat al het geval dan hoef je niets te doen.

Anders doe je een rechtermuisklik op de tekst “JRE System Library” en ga je naar de properties om daar de workspace default te kiezen.

Sikuli embedded in het framework

Om Sikuli te gebruiken in het framework is er niet heel veel nodig. Zoals je eerder gelezen hebt is Maven een buildtool om code van buitenaf te downloaden en in je framework te kunnen gebruiken. Het enige wat je dan ook hoeft te doen om sikuli te gebruiken is het toevoegen van de juiste dependency in je pom.xml. Open de pom.xml in je project en klik in het nieuwe tabblad wederom op pom.xml. Daarna kan je de dependency toevoegen. De dependency die je moet gebruiken is de onderstaande, waarbij de versie anders kan zijn. Om te kijken of er een nieuwere versie is kijk je op [www.mvnrepository.com](http://www.mvnrepository.com).

<dependency>

<groupId>com.sikulix</groupId>

<artifactId>sikulixapi</artifactId>

<version>1.1.1</version>

</dependency>

In je project in eclipse kan je nu op het mapje klikken “src/test/java” en kan je door door te klikken het java bestand AppTest.java openen. Laat regel 1 staan (package …) voor de rest kan je dit bestand leegmaken.

Zet er de volgende tekst in:

**public** **class** AppTest {

**public** **static** **void** main(String[] args) {

[HIER KOMT JE CODE]

}

}

Vervolgens kan je in je code een padnaam opgeven (ImagePath.add(“…”) waar de afbeeldingen te vinden zijn en kan je er direct mee aan de slag.

De volgende code kan je bijvoorbeeld rechtstreeks gebruiken in je test stappen. Als je programma bij de desbetreffende regels is zal je muis het overnemen. Deze gaat dan naar de afbeelding in je scherm die overeenkomt met de screenshot waar je op wilt klikken.

**public** **class** AppTest {

**public** **static** **void** main(String[] args) {

Screen sikuli = **new** Screen();

ImagePath.*add*("src/test/java/images/");

sikuli.click("windowsLogo.png");

sikuli.wait("shutDownButton.png");

}

}

Je zult zien dat bepaalde woorden rood onderstreept zijn. Dit zijn functies die nog geïmporteerd dienen te worden met een quick fix. Dit doe je door met je muis op het woord te staan en vervolgens klik je op import "betreffende functie". (deze imports worden mogelijk gemaakt door je pom.xml file die je eerder hebt geconfigureerd). Je kan nu zien dat bovenaan je code is aangevuld met de imports. Deze quick fix dien je in het vervolg ook uit te voeren voor de code die we uitgevoerd wordt. Als je een rood onderstreept woord ziet ga er dan met je muis op staan zodat een popup komt waarbij je de functie kan importeren.

Hier heb je een flink aantal commando’s een aantal mogelijkheden zijn:

* Aanklikken
* Wachten op verschijnen
* Wachten op verdwijnen.
* Dubbelklikken
* Rechtermuisklik
* Slepen
* etc

#### Opdracht

Probeer door gebruik te maken van de voorbeeldcode hierboven, het scenario van eerder nu in Eclipse te zetten. Kom je er niet uit? Kijk dan op de laatste pagina voor een voorbeeld. Voornamelijk de loop die we eerder hebben gemaakt is wat ingewikkelder. Kom je er nog steeds niet uit? Ga dan weer verder met de standalone tool om je eigen scenario te bedenken!

## 

## Virtualbox

Om dit geheel te draaien in je eigen virtuele machine (vm) kan je virtualbox downloaden. Hiermee kan je verschillende besturingssystemen via je eigen pc draaien. Zo kan je een VM draaien met Windows 10 er op waardoor je je testen met sikuli kan draaien zonder dat je er last van hebt.

1. Ga naar <https://www.virtualbox.org>

2. Download virtualbox

3. Download een image met als platform virtualbox met bijvoorbeeld Windows 10: <https://developer.microsoft.com/en-us/microsoft-edge/tools/vms/>

4. Pak het bestand uit met 7-zip of een andere tool

5. Dubbelklik op het uitgepakte bestand

6. Open het bestand met virtualbox en accepteer de popups binnen de applicatie

Wanneer dit allemaal is gedownload en geïnstalleerd dan kan je inloggen als IEUser met als wachtwoord: Passw0rd!

Als je hierna de voorgaande stappen herhaald (java, maven, eclipse en sikulix installeren en initialiseren) dan kan je met image recognition testen. Houdt er rekening mee dat een VM een geldigheid heeft van 90 dagen. Het slimste is dus om een zogenoemde snapshot te maken op het moment dat je het framework hebt draaien. Dan kan je als de licentie bijna verloopt terug naar deze snapshot. Je doet dit door in virtualbox bij machinegereedschap, op het pijltje naar onder te klikken en daar snapshots te selecteren.

## Sikuli testgeval

**Voorbeeldcode voor de opdracht**

import org.sikuli.script.FindFailed;

import org.sikuli.script.ImagePath;

import org.sikuli.script.Pattern;

import org.sikuli.script.Screen;

**public** **class** AppTest {

**public** **static** **void** main(String[] args) throws FindFailed, InterruptedException {

Screen sikuli = new Screen();

ImagePath.add("C:\\Users\\Jurgen\\Desktop\\sikuliscript.sikuli");

sikuli.click("windowsLogo.png");

sikuli.wait("shutDownButton.png");

sikuli.type("Internet Explorer");

sikuli.click("internetexplorer.PNG");

sikuli.click("IEsettings.png");

sikuli.click("Internetopties.png");

sikuli.click("verwijderen...Button.png");

sikuli.wait("legeCheckBox.png");

// Opmerking: vanaf hier begint het deel voor de loop, eerst een object maken,

// omdat de checkbox anders te breed geïnterpreteerd wordt en daarna de

// daadwerkelijke loop.

Object p = new Pattern("legeCheckBox.png").similar((float) 0.87);

while (sikuli.exists(p) != null) {

sikuli.click("legeCheckBox.png");

}

sikuli.click("verwijderenButton.png");

sikuli.wait("geselecteerdeGeschiedenisVerwijderd.png");

sikuli.click("okButton.png");

sikuli.waitVanish("okButton.png");

sikuli.click("afsluitenButton.png");

}

}