

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Projekt 5

**Využití nástrojů pro
testování grafického
uživatelského rozhraní**

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 30. listopadu 2015

Jaroslav Klaus

Abstract

This paper deals with the use of software tools for testing graphical user interface. It compares some of the tools and describes the use of one of them in a way that fits into the subject KIV/OKS.

Abstrakt

Tato práce se zabývá využitím nástrojů k testování grafického uživatelského rozhraní aplikací. Srovnává některé nástroje k tomu určené a popisuje použití jednoho z nich tak, aby svou filosofií zapadal do předmětu KIV/OKS.

Obsah

1	Úvod	1
2	Přehled nástrojů	2
3	Zvolené nástroje	4
3.1	Jubula	4
3.2	SikuliX	4
3.3	Robot Framework	5

1 Úvod

Testování aplikací je nedílnou součástí jejich vývoje a v dnešní době se tomuto oddílu tvorby aplikací věnuje čím dál více pozornosti. Dá se rozdělit do různých skupin, např. podle toho, kdy se testování provádí, nebo podle toho, jakým způsobem se provádí, nebo jak se k testované aplikaci přistupuje, či jaká část aplikace se podrobuje testům.

Jednou z důležitých součástí je testování grafického uživatelského rozhraní¹. Zde se testeré soustředí na to, zda daná aplikace vypadá tak, jak to požaduje návrh a vývojáři, zda grafické prvky správně fungují, nebo zda je aplikace přívětivá k uživateli a práce s ní není příliš komplikovaná.

Při testování grafického uživatelského rozhraní se může spousta testů mnohokrát opakovat, a proto je snaha tyto testy nějak automatizovat. K tomu se může využít některý z nástrojů k tomu určený. Cílem této práce je seznámit se s některými z těchto nástrojů, jeden z nich vybrat a pomocí něj vytvořit sadu ukázkových testů svou filosofií zapadajících do předmětu KIV/OKS².

¹Označované jako GUI - Graphical User Interface

²<https://portal.zcu.cz/portal/studium/courseware/kiv/oks>

2 Přehled nástrojů

V této kapitole následuje přehled nástrojů a některých jejich vlastností. V tabulce 2.1 je uveden název nástroje, jeho licence resp. cena, jazyk, ve kterém se testy píší, platforma, na které nástroj funguje a která GUI je nástroj schopen testovat. Z bezplatných multiplatformních nástrojů bylo mým úkolem vybrat tři a ty podrobněji prozkoumat a porovnat, viz následující kapitola.

Název	Licence/Cena	Skriptovací jazyk	Platforma	Jazykové omezení
AutoIt ¹	Freeware	BASIC-like	Windows	-
AutoHotKey ²	GNU GPLv2	AutoHotKey	Windows	-
AutoKey ³	GNU GPLv3	Python	Linux	-
SikuliX ⁴	MIT License	Python, Ruby	Windows, Linux, Mac	-
Jubula ⁵	EPL 1.0	Drag & Drop, Java	Windows, Linux, Mac	Java, HTML, .NET, iOS
Robot Framework ⁶	Apache License 2.0	Natural-like	Windows, Linux, Mac	Podle!addonů (Java, web, Android, iOS, ...)
Squish ⁷	cca 2400 €/osoba	Python, JavaScript, Ruby, Perl, Tcl	Windows, Linux, Mac	-
eggPlant ⁸	nedostupná, vázaná na stroj	SmartTalk, Drag & Drop, pomocí rozhraní eggDrive např. Java, C#, Ruby	Windows, Linux, Mac	-
UFT ⁹	nedostupná	VBScript, Drag & Drop	Windows, Linux, Mac	-

¹<https://www.autoitscript.com/site/>

²<http://www.autohotkey.com/>

³<https://code.google.com/p/autokey/>

⁴<http://www.sikuli.org/>

⁵<http://www.eclipse.org/jubula/>

⁶<http://robotframework.org/>

⁷<http://www.froglogic.com/squish/gui-testing/index.php>

⁸<http://www.testplant.com/eggplant/testing-tools/eggplant-developer/>

⁹<http://www8.hp.com/cz/cs/software-solutions/unified-functional-automated-testing/>

Název	Licence/Cena	Skriptovací jazyk	Platforma	Jazykové omezení
Rational Functional Tester ¹⁰	3300 \$/osoba	Nahrávání akcí	Windows, Linux	-
Ranorex ¹¹	690 €	C#, VisualBasic, nahrávání akcí	Windows	-
SilkTest ¹²	nedostupná	C#, VisualBasic, Java	Windows	-
TestComplete ¹³	889 €/stroj	Python, VBScript, JScript, C#Script, DelphiScript, C++Script, nahrávání akcí	Windows	-

Tabulka 2.1: Přehled nástrojů

¹⁰<http://www-03.ibm.com/software/products/cs/functional>

¹¹www.ranorex.com

¹²<http://www.borland.com/Products/Software-Testing/Automated-Testing/Silk-Test>

¹³<http://smartbear.com/product/testcomplete/overview/>

3 Zvolené nástroje

Vzhledem k požadavkům na nástroje jako bezplatnost, schopnost fungování nezávisle na OS¹ nebo podpora testování programů vytvořených technologií Java² a webových aplikací jsem z výše zmíněných vybral nástroje Jubula, SikuliX a Robot Framework. Každý z nástrojů bude podrobněji popsán v následujících sekcích.

3.1 Jubula

Jubula je nástroj, který vznikl a je vyvíjen v rámci IDE³ Eclipse⁴ a do projektu přispívá také firma BREDEX GmbH⁵, která vytváří také tzv. standalone verzi, což je program, který je možné používat samostatně bez IDE Eclipse, obsahuje navíc některé nespecifikované funkce a nemusí být licencována pod EPL 1.0⁶ jako je tomu u verze pro IDE Eclipse.

Pro tvorbu testovacích skriptů byla používána metoda Drag & Drop, popř. se akce určovaly klikáním na různé nabídky, avšak v jedné z posledních verzí bylo vydáno Java API⁷ a skripty je tak možné psát pomocí jazyka Java. Mezi podporovaná testovaná rozhraní patří Java Swing, SWT, JavaFX, HTML a iOS. Výhodou této aplikace je také možnost její integrace pro ostatních programů pro organizaci testování.

3.2 SikuliX

Sikuli (nověji SikuliX) je nástroj, který vznikl jako projekt skupiny User Interface Design Group na MIT⁸, což odpovídá i jeho licenci - MIT License⁹

¹operační systém

²<https://www.oracle.com/java/index.html>

³Integrated Development Environment - integrované vývojové prostředí

⁴<http://www.eclipse.org/jubula/>

⁵http://www.bredex.de/guidancer_jubula_en.html

⁶<https://www.eclipse.org/legal/epl-v10.html>

⁷Application Programming Interface - rozhraní pro programování

⁸Massachusetts Institute of Technology

⁹<https://opensource.org/licenses/MIT>

a nyní jeho vývoj převzal Raimund Hock (aka RaiMan) společně s open-source komunitou¹⁰.

Pro tvorbu skriptů je možné využít pro SikuliX vlastní jazyk podobný přirozené angličtině, nebo některý ze zavedených, jako je Python, Ruby, Java, Jython, JRuby, Scala, Groovy, Clojure a dalších. Nástroj není omezený na určitá testovaná rozhraní, protože k identifikaci GUI používá rozpoznání obrazu podle vzoru¹¹, dokáže simulovat ovládání myši a klávesnice nebo rozpoznávat text v obrázcích¹². Výhodou této aplikace je tak její nezávislost vůči testovanému rozhraní.

3.3 Robot Framework

Robot Framework je nástroj založený na pluginech¹³, vývoj je podporován společností Nokia Networks¹⁴ a projekt je open-source.

Základ nástroje, tzv. core framework, je vytvořený v jazyce Python a knihovny je možné psát také v jazyce Python nebo Java a samotné skripty pak v jazyce podobném přirozené angličtině, které dodržují jisté formátování, a tím jsou pro člověka velmi přehledné. Mezi podporovaná testovaná rozhraní patří např. Android, iOS, Java Swing, webové aplikace, databáze a aplikace vytvořené pro OS Windows. Výhodou této aplikace je přehledná zpráva v HTML o průběhu testů, která je automaticky generována po skončení.

¹⁰skupina lidí, která sdílí záměr a vytváří aplikace s open-source licencí

¹¹pomocí OpenCV, <http://opencv.org/>

¹²pomocí Tesseract OCR, <https://github.com/tesseract-ocr>

¹³knihovny, resp. volitelné části, které je možné přidat

¹⁴<https://networks.nokia.com/>

Literatura