



Software Engineering (Experiments)

Mrs. Yoothika S. Patel, Assistant Professor
CSE, PIET





Experiment-10

Demonstrate the use of different Testing Tools with comparison



Aim: Demonstrate the use of different Testing Tools with comparison

Objectives: To show the working of testing tools and compare the features.



Automated Testing

[According to the recent survey, 48 percent](#) of QA teams admit to relying heavily on manual testing. While technology doesn't yet allow us to stop manual testing altogether, it's still an interesting statistic. Test automation exists to reduce the big-project testing process that can't be possibly done by a human. It will notice details that humans can miss, especially on repeated tests.



Why automated testing?

- Some of the tools on the market offer testing solutions with no coding required, which allow any member of the team to perform tests.
- This is made possible by a tool's ability to translate written text into the programming language, hiding code under the layer of user-friendly UI.
- While this may be a great solution for an organization with a limited number of developers, it's still a pretty restricted option.
- Without access to code, you lose on customization and detailed feedback that's kept in the core.
- It's each company's own decision: to opt for simple and accessible solutions or enable their employees to learn the basics of some simple language like Python.



Comparison of some testing tools

COMPARING QA AUTOMATION TOOLS

	Price	Platform	Supported languages	Tested apps	Coding skills required	Learning curve
Selenium	Free	Windows/Mac/Linux	Java, Python, C#, PHP, JavaScript, Ruby, Perl	Web, mobile (with Appium)	Advanced skills	Steep
TestComplete	\$4400/\$9000	Windows	VB, JavaScript, Jscript, C++, C#, Delphi, Angular, Ruby on Rails, PHP	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Mild
Tricentis Tosca	Custom, high (according to online discussions)	Windows	JavaScript	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Mild
Katalon Studio	Free	Windows/Mac	Java/Groovy	Web, mobile	Minimum skills/Advanced skills for pro scripting	Mild
UFT	\$2500/\$3500	Windows	VBScript	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Moderate
Watir	Free	Windows	Ruby (Java/.Net alternatively)	Web	Minimum skills	Mild
Ranorex	\$2800/\$850	Windows	C#, VB, .Net, Iron Python	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Moderate



Selenium IDE

Selenium IDE (Integrated Development Environment) is the simplest tool in the Selenium Suite. It is a Firefox add-on that creates tests very quickly through its record-and-playback functionality. This feature is similar to that of QTP. It is effortless to install and easy to learn.

Because of its simplicity, Selenium IDE should only be used as a prototyping tool, not an overall solution for developing and maintaining complex test suites.



Selenium IDE: Why it is Good?

- Large community.
- Simultaneous tests.
- Mobile support.



Selenium IDE: Where's the catch?

- Not beginner-friendly.
- No image verification.



TestComplete: Why it is Good?

- Ease of use.
- Customization.
- Timely updates.
- Support of desktop apps.



TestComplete: Where's the catch?

- No Mac support.



Katalon Studio: Why it is Good?

- **Good for both pros and non-techies.**
- **Unified bundle.**
- **Abundance of tutorials.**
- **Visualized reports.**



Katalon Studio: Where's the catch?

- Poor language support.
- Small (although growing) community.



Unified Functional Testing(UFT): Why it is Good?

- Automated tests from manual.
- Collaboration capabilities.



Unified Functional Testing(UFT): Where's the catch?

- Only one language supported.
- Price.
- Only Windows support.



Watir: Why it is Good?

- Choice of languages.



Watir: Where's the catch?

- May be too simple.
- Small community.

Experiment Demonstration