



Gemeinsam mehr erreichen.

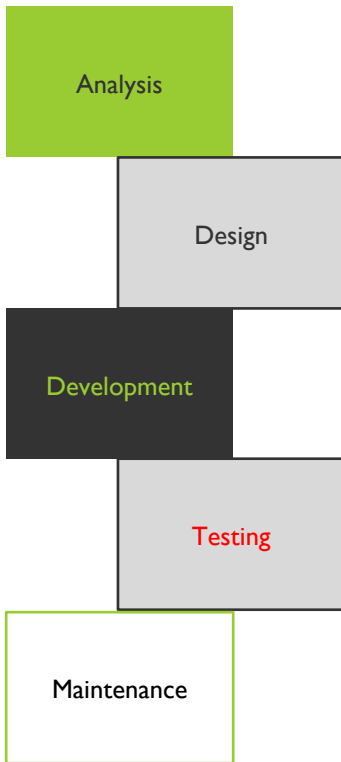
esentri

Java and Selenium: Automating Test, is it always better? – DOAG 2016

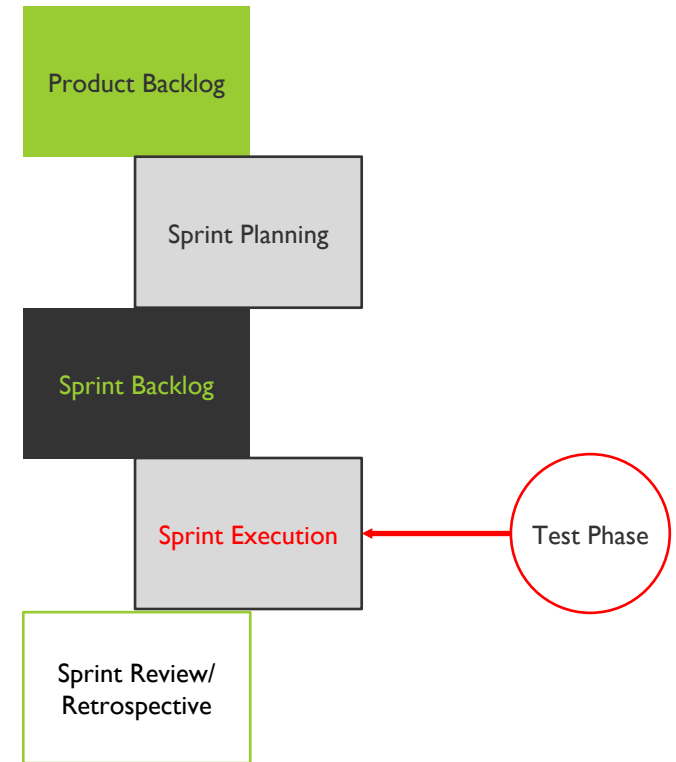
Maria Emanuela Arico – esentri AG

Software Development & Test Phase

- > In the software development the test phase is really important
- > Goals:
 - > Detect bugs and failures in the software in order to fix them
 - > Check the usability of the application



Classic Development



Scrum Iteration

Test Phase

- > 5 Levels of Testing
 - > Unit Testing - Single function/procedure
 - > Integration Testing – Functions/procedures as a group / Detects defects in the integration
 - > System Testing – Complete application
 - > Acceptance Testing – Whole application ready for release
 - > Regression Testing – Modifications



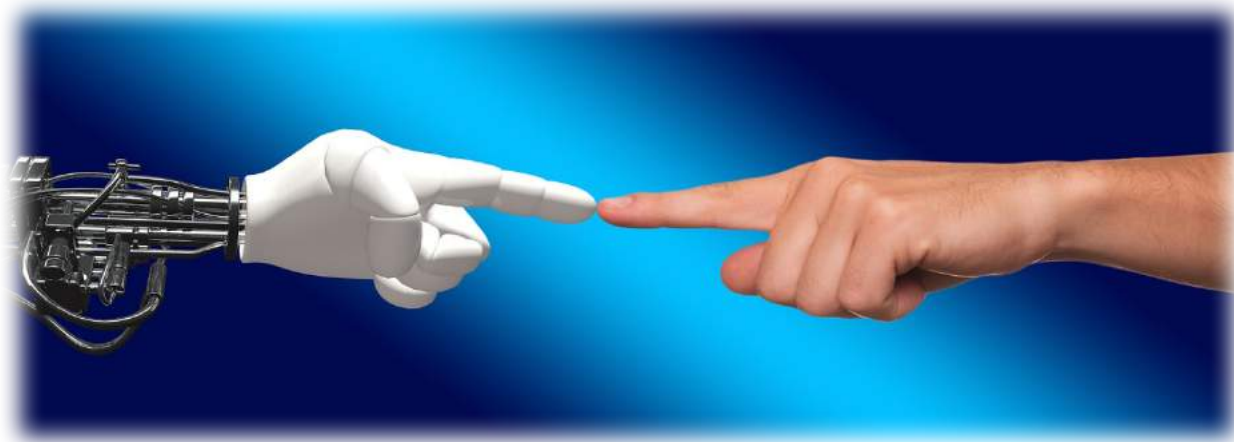
Automated Test vs Manual Test I

Automated Test

- > Automatic execution of pre-defined scripts
- > Needs a developer to write and set them
- > Can be executed fast before each release
- > The execution can be scheduled as we want
- > We can plan the input variable

Manual Test

- > Needs a QA (physical person)
- > It's affected by the possible "human error"
- > It's more flexible for the immediate need of specific tests
- > Gives us a "human" feedback
- > Cannot be scheduled and is linked to the presence of the QA



Automated Test vs Manual Test II

Automated Test

- > Requires the investment to hire a test engineer and developers
- > Better for repetitive and performance test

Manual Test

- > Need to invest money to keep hired a QA (or more)
- > Human "approach"
- > Better for hoc-based script
- > Helps to improve the user-friendliness and usability of the application



Automated Test framework – Selenium

- > “Selenium automates browsers. that's it.” (Cit.)
- > Framework for automated GUI test.
- > Multiple platform supported
- > Open source
- > “Selenese” as own language script
- > APIs to write tests in the most used languages (Java, PHP, Python...)
- > Integration in our project
- > Firefox plugin to create tests without coding



Selenium & Java Development

```
1  import org.openqa.selenium.*;
2  import org.testng.*;
3
4  ▼ public class MyFirstSeleniumTest {
5  ▼      public static void main(String[] args) {
6
7          // Instance of Firefox Driver
8          WebDriver myDriver = new FirefoxDriver();
9          // Going to our test WebSite
10         myDriver.get("http://www.mywebsite.com");
11         //Search for two forms and insert inside a text
12         myDriver.findElement(By.id("username")).sendKeys("user@email.com");
13         myDriver.findElement(By.id("password")).sendKeys("12345");
14         // Pressing one button in the website
15         driver.findElement(By.name("MyButton")).click();
16         // Instance of a new WebDriver wait
17         WebDriverWait driverWait = new WebDriverWait(myDriver, 10);
18         // Waits until a Popup is present
19         Alert myAlert = wait.until(ExpectedConditions.alertIsPresent());
20         // Accepts the popup
21         myAlert.accept();
22         // Check inside the website is present inside an object a specific text
23         Assert.assertEquals(welcomeText, "Welcome User!");
24     }
25 }
```

Selenium & Java Development

- > Some more methods:
 - > `driver.findElement(By.id("myElement")).getText();`
 - > `myElement.submit();`
 - > `driver.quit();`
 - > `driver.getTitle();`
 - > `driver.switchTo().window("myWindow");`
 - > `driver.navigate().forward();`
 - > `driver.navigate().back();`



Selenium & Java Development

- > Annotations:
 - > `@BeforeTest` : Executed before the first `@Test` class.
 - > `@Test` : Test Class
 - > `@AfterTest` : Executed after the last `@Test` class



Is Selenium the future of the software testing process?

- > Growing fast - thanks to open source nature, free to use.
- > Supports all the most famous web browsers (Firefox, Chrome, Safari, IE...)
- > Compatibility and integrations with all the main coding languages
- > Compatible also with the two main mobile OS (iOS and Android)



Can automatic test (and Selenium) replace completely the manual one?

- > Customers want always as much as possible.
- > Can we have automated test for everything?
- > Can we write all the scripts needed to test all the possible functionalities and behaviour of the application and the user that will use it?
- > Can we “plan” all the tests we will need also in the future?
- > Is it really worth it to have a script to test a single bugfix?



Let's try to understand it better with an example!

facebook

Email or Phone Password [Log In](#)

[Forgot account?](#)

Sign Up

It's free and always will be.

First name Last name

Mobile number or email

Re-enter mobile number or email

New password

Birthday

Month Day Year [Why do I need to provide my birthday?](#)

Female Male

By clicking Sign Up, you agree to our [Terms](#) and that you have read our [Data Policy](#), including our [Cookie Use](#).

[Sign Up](#)

[Create a Page](#) for a celebrity, band or business.

English (US) Italiano Deutsch Türkçe Polski Română Français (France) Русский العربية Español Português (Brasil) [+](#)

[Sign Up](#) [Log In](#) [Messenger](#) [Facebook Lite](#) [Mobile](#) [Find Friends](#) [Badges](#) [People](#) [Pages](#) [Places](#) [Games](#)
[Locations](#) [Celebrities](#) [Groups](#) [Moments](#) [About](#) [Create Ad](#) [Create Page](#) [Developers](#) [Careers](#) [Privacy](#) [Cookies](#)
[Ad Choices](#) [Terms](#) [Help](#)

Conclusion : when is better to use an automated tests (and Selenium)?

- > To carry out many repetitions of the test that are executed faster by a tool
- > To test a functionality that has to be tested in every released update of the application
- > In case of “performance tests” and “stress tests”, so when we have to simulate a big amount of users using the application at the same time
- > When functionality of loading or processing of the application tests are required



Conclusion : when is better to use a manual test?

- > To test the “usability” and the “user-friendliness” of the application
- > To test a specific function or behaviour of the application
- > To check a specific test-case, that needs to be tested only a few times, and it does not really need the development of an hoc-script
- > To evaluate new features or functions of the application
- > When “human feedback” about one or more functions is required



Conclusion : a general rule

- > Using a manual test is better when something not strictly related to the “pure” functionality of the application has to be tested, and something needs to be judged by a “human eye”.
- > Using a automated test (and Selenium) is better when we have several repetitions and it is not needed a “human point of view”



About me



Maria Emanuela Arico

Position : Consultant

E-Mail : emanuela.arico@esentri.com

Twitter : @emanuelarico



Vernetzen mit esentri

