

# The Art and Science of Test Automation

Mandana Nabizadeh

October 2018

# Agenda

- Why Test Automation
- Science of Test Automation
  - Automation Framework
- Art of Test Automation
  - What to Test

# Why Test Automation

- Do you automate your tests?
- Entirely through the user interface?
- Do you still do most of your testing manually because your automated tests can't support your testing objectives?

# Types of Automated Tests

Which types of automated testing are your QA and Development teams primarily performing?  
(Respondents could select more than one answer)

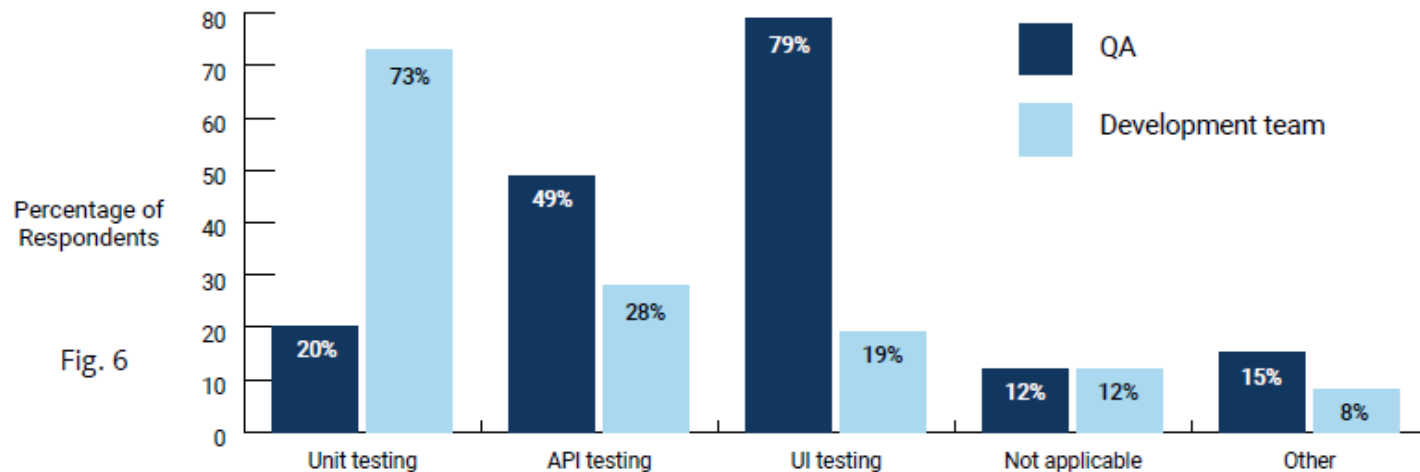


Fig. 6

Source: The evolution of test automation Results from the 2018 QASymphony and TechWell Survey

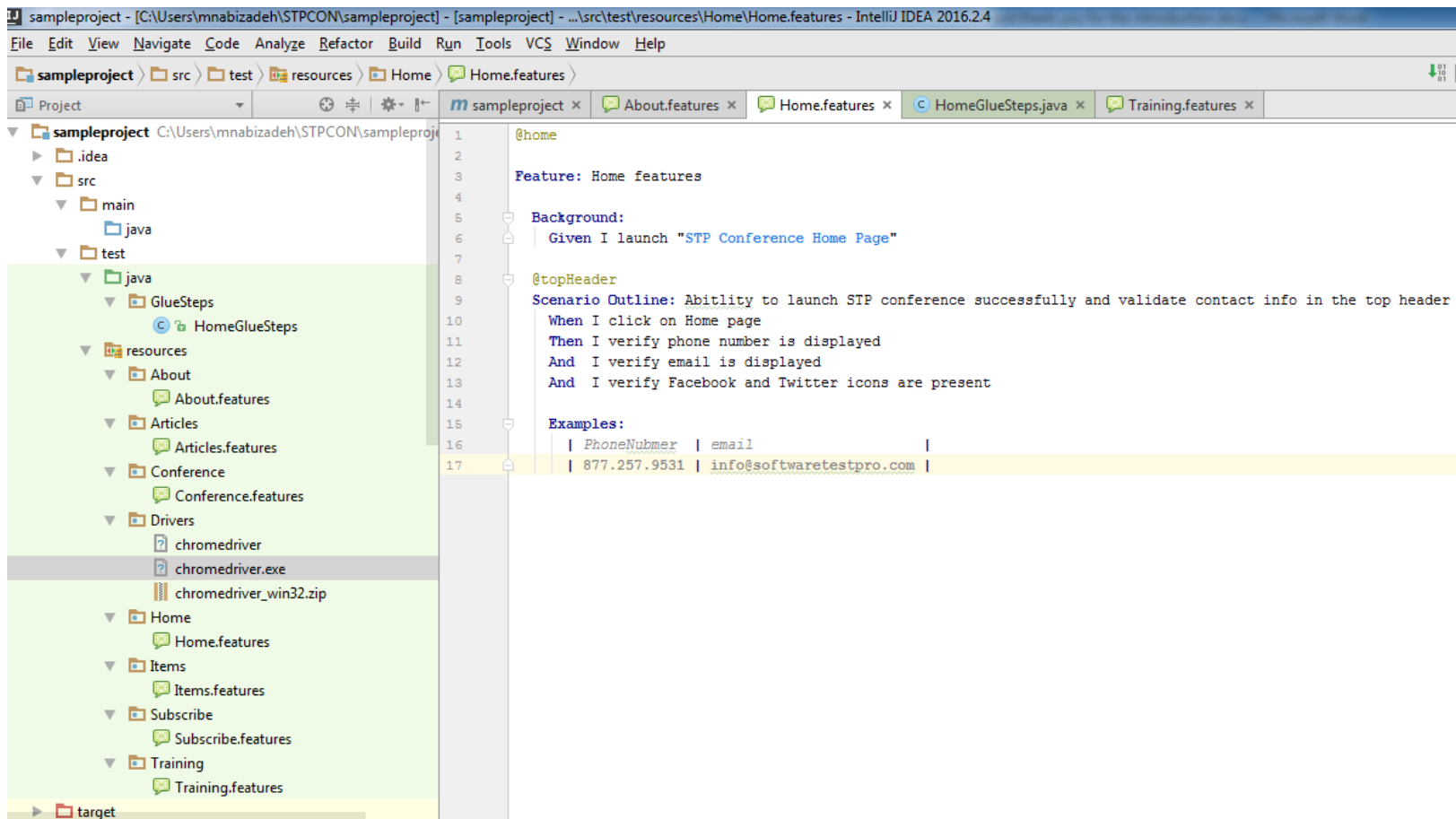
# The Science of Test Automation

- **Science:**
  - a systematically organized body of knowledge on a particular subject.
- **Test Automation**
  - Use of software to control the execution of tests
  - Comparison of actual outcomes with predicted outcomes

# Automation Framework

- Framework Selection
  - Type of application under test
    - Mobile
    - Web-based
    - Desktop
  
- Technology used

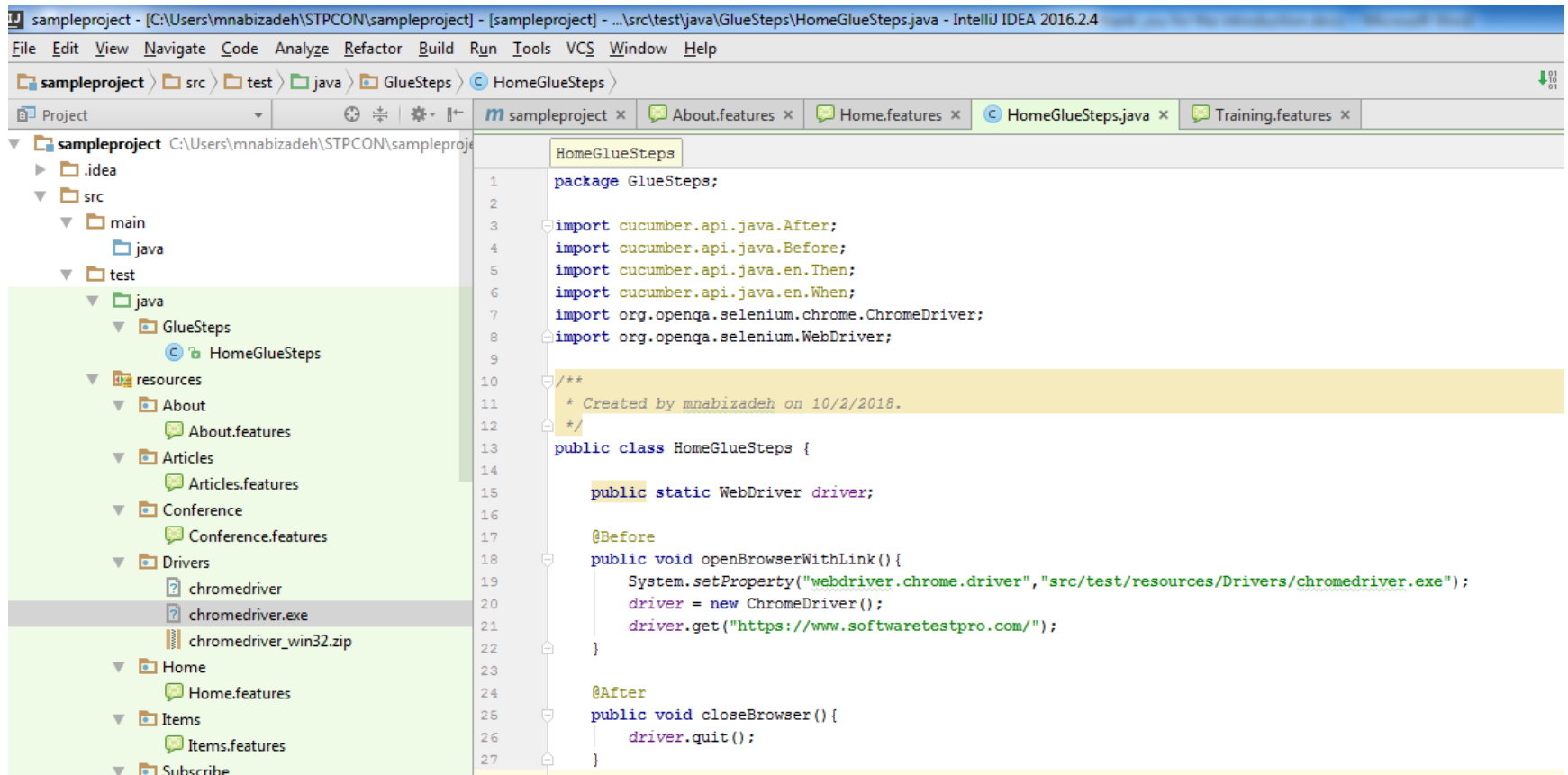
# Behavioral Driven Development (BDD)



The screenshot shows the IntelliJ IDEA interface with a project named 'sampleproject'. The left sidebar displays the project structure, including folders for 'src', 'test', and 'resources'. The 'test/resources/Home' folder is expanded, showing 'Home.features'. The main editor window displays the content of 'Home.features', which is a Gherkin-style BDD scenario. The scenario is titled '@home' and describes a feature for 'Home features'. It includes a background step, a scenario outline, and an examples table.

```
1 @home
2
3 Feature: Home features
4
5 Background:
6   Given I launch "STP Conference Home Page"
7
8 @topHeader
9 Scenario Outline: Ability to launch STP conference successfully and validate contact info in the top header
10  When I click on Home page
11  Then I verify phone number is displayed
12  And I verify email is displayed
13  And I verify Facebook and Twitter icons are present
14
15 Examples:
16   | PhoneNubmer | email |
17   | 877.257.9531 | info@softwaretestpro.com |
```

# Step Definition (Glue)



The screenshot displays the IntelliJ IDEA IDE interface. The top menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The breadcrumb navigation shows the path: sampleproject > src > test > java > GlueSteps > HomeGlueSteps. The Project tool window on the left shows a tree view of the project structure, with the 'test' directory expanded to show the 'HomeGlueSteps' class. The main editor window shows the source code for 'HomeGlueSteps.java'.

```
1 package GlueSteps;
2
3 import cucumber.api.java.After;
4 import cucumber.api.java.Before;
5 import cucumber.api.java.en.Then;
6 import cucumber.api.java.en.When;
7 import org.openqa.selenium.chrome.ChromeDriver;
8 import org.openqa.selenium.WebDriver;
9
10 /**
11  * Created by mnabizadeh on 10/2/2018.
12  */
13 public class HomeGlueSteps {
14
15     public static WebDriver driver;
16
17     @Before
18     public void openBrowserWithLink(){
19         System.setProperty("webdriver.chrome.driver", "src/test/resources/Drivers/chromedriver.exe");
20         driver = new ChromeDriver();
21         driver.get("https://www.softwaretestpro.com/");
22     }
23
24     @After
25     public void closeBrowser(){
26         driver.quit();
27     }
28 }
```



# POM.xml

The screenshot displays the IntelliJ IDEA interface with a Maven POM.xml file open. The left sidebar shows the project structure, including the 'target' directory and 'pom.xml' file. The main editor shows the XML content of the POM file, which defines the project's metadata and dependencies. The bottom panel shows the test results for a scenario, indicating that all tests passed.

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5     <modelVersion>4.0.0</modelVersion>
6
7     <groupId>com.stpcon.sampleproject</groupId>
8     <artifactId>sampleproject</artifactId>
9     <version>1.0-SNAPSHOT</version>
10
11     <dependencies>
12         <dependency>
13             <groupId>info.cukes</groupId>
14             <artifactId>cucumber-testng</artifactId>
15         </dependency>
16         <dependency>
17             <groupId>info.cukes</groupId>
18             <artifactId>cucumber-java</artifactId>
19             <version>1.2.5</version>
20         </dependency>
21         <dependency>
22             <groupId>info.cukes</groupId>
23             <artifactId>cucumber-core</artifactId>
24             <version>1.2.5</version>
25         </dependency>
26         <dependency>
27             <groupId>info.cukes</groupId>
28             <artifactId>cucumber-jvm-deps</artifactId>
29             <version>1.0.5</version>
30         </dependency>
31         <dependency>
32             <groupId>info.cukes</groupId>

```

Run Scenario: Ability to launch STP conference successfully and validate contact info in the top header Done: Scenarios 1 of 1 (11s 211ms)

```

All Tests Passed 95ms
"C:\Program Files\Java\jdk1.8.0_121\bin\java" ...
Testing started at 8:56 AM ...
Starting ChromeDriver 2.42.591088 (7b2b2dca23cca0862f674758c9a3933e685c27d5) on port 41901
Only local connections are allowed.
Oct 08, 2018 8:56:15 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS

1 Scenarios (1 passed)
5 Steps (5 passed)
0m9.982s

Process finished with exit code 0

```

# Art of Test Automation

- Scalable, maintainable framework
- Valuable tests
- Testing is a scientific practice

# Learn the fundamentals

- Testing is observing and analyzing
- Framework gives you Scaffolding



# Learning to draw

- Developing successful automation suites are highly dependent on efficient test.
  - Apply D.R.Y concept to test case creation
  - Consider the value of the tests
  - Clear, concise, only one objective
  - Use test case design techniques

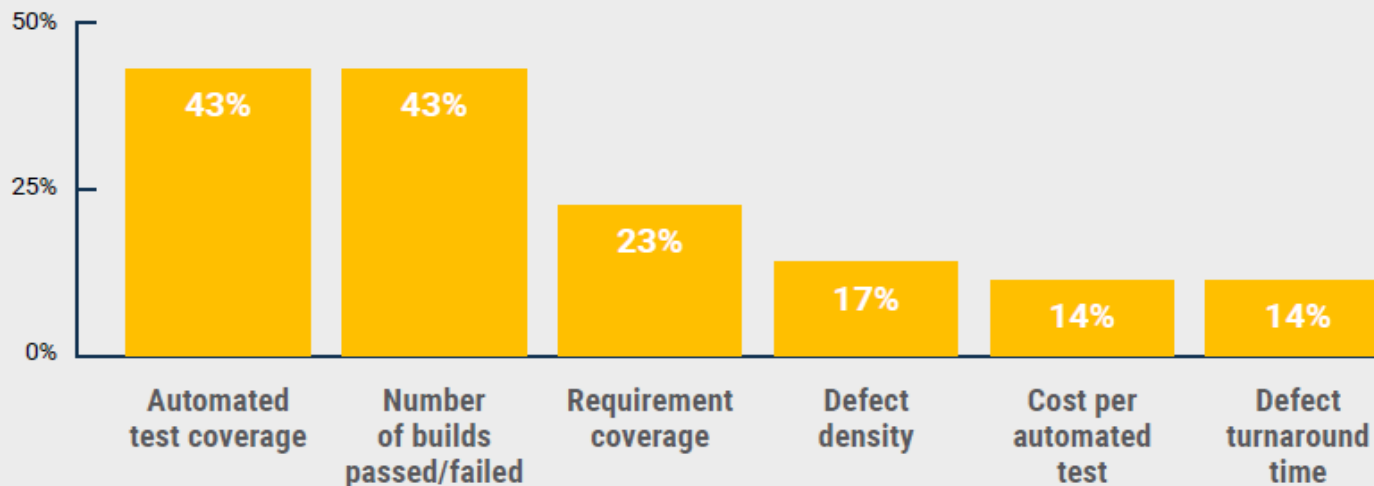
# Stabilize Your Test Suites

- Automated suites are living artifacts
- Re-visit your suite, adjust if necessary
- Remove test with little or no value
- Add high impact tests
- **Avoid Old Oak Tree Syndrome** (Reference: Lesson learned from Software Testing – A context driven approach)

# Quantify & Measure

## How Do You Measure Success?

Top 6 Key Performance Indicators



Source: The evolution of test automation Results from the 2018 QASymphony and TechWell Surrvey

# Further reading

- Lessons Learned in Software Testing by Cem Kraner, James Bach, Brett Pettichord
- Thinking in Systems by Donella H. Meadows
- Testing Computer Software by Cem Kaner, Jack Falk and Hung Quoc Nguyen
- Selenium Design Patterns and Best Practices by Dima Kovalenko
- Continuous Testing For Devops Professionals by Eran Kinsbruner

# Contact

Mandana.nabizadeh@discounttire.com

Twitter: Mon\_Donna

Linkedin: [linkedin.com/in/mandananabizadeh](https://www.linkedin.com/in/mandananabizadeh)



# Q & A

