Funded by the
European Union

Extending WebDriver: A cloud approach

Boni García

boni.garcia@urjc.es

11th International Conference on the Quality of Information and Communications Technology (QUATIC 2018)
September 6th 2018 | Coimbra, Portugal

http://elastest.io

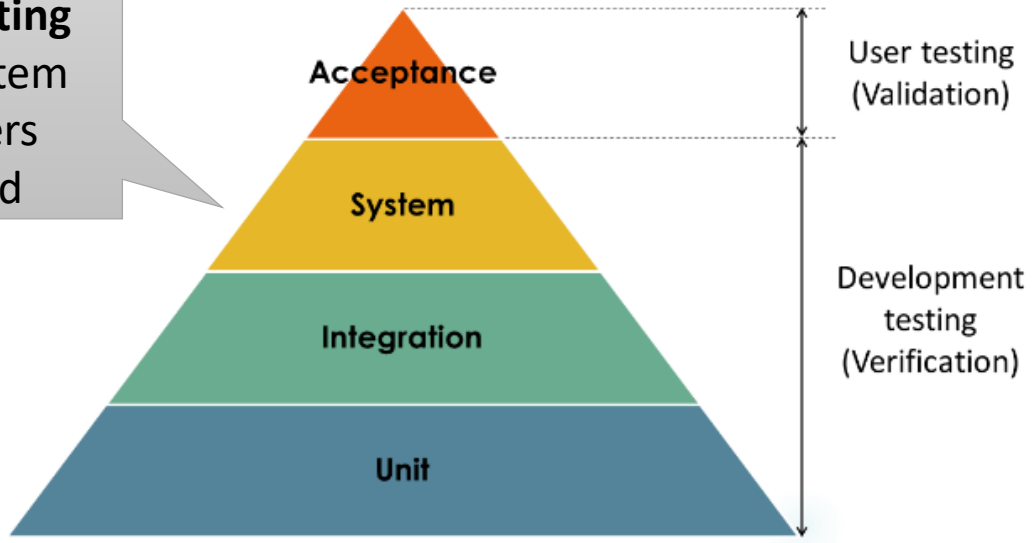
Table of contents

1. Introduction
2. Background
3. ElasTest: platform for end-to-end testing
4. User Impersonation as a Service
5. Demo
6. Conclusions and future work

1. Introduction

- Large distributed heterogenous systems are more and more common (e.g. microservices architectures, cloud native apps, etc.)
- Testing this kind of software is complex, especially to verify the system as a whole

End-to-end testing happens at system level with users impersonated



ElasTest

Table of contents

1. Introduction
2. Background
3. ElasTest: platform for end-to-end testing
4. User Impersonation as a Service
5. Demo
6. Conclusions and future work

2. Background

- The main mechanism used in the current state-of-the-art for the functional testing of web and mobile applications consists on impersonating a user through some kind of GUI automation
- **Selenium** is the most popular solution:

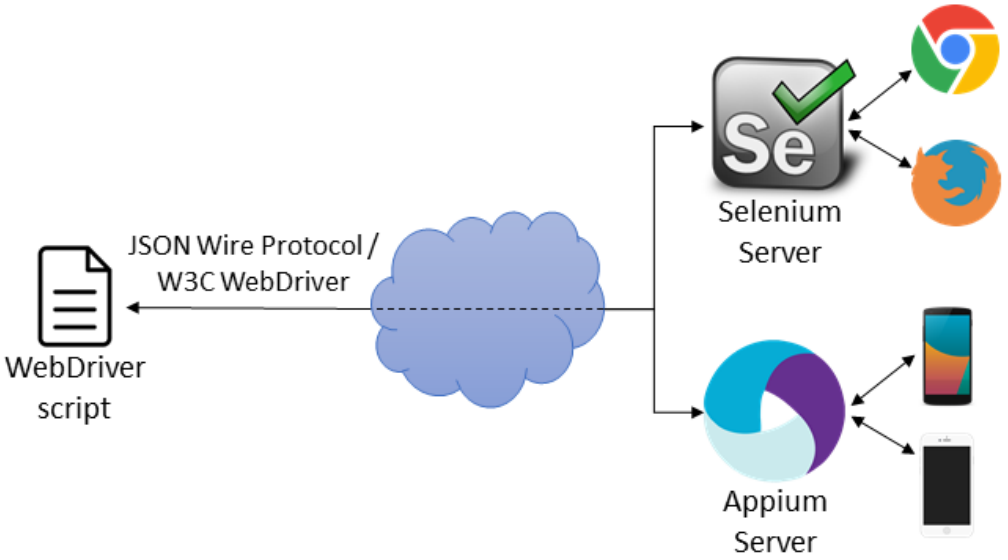


Table of contents

1. Introduction
2. Background
3. ElasTest: platform for end-to-end testing
4. User Impersonation as a Service
5. Demo
6. Conclusions and future work

3. ElasTest: platform for end-to-end testing

- ElasTest is an open source platform aimed to ease the end-to-end testing activities for different types of distributed applications and services
- ElasTest manages the full testing lifecycle, deploying and monitoring the System Under Test (SUT), executing the end-to-end tests and exposing the results to testers

http://elastest.io/

3. ElasTest: platform for end-to-end testing

- ElasTest architecture:

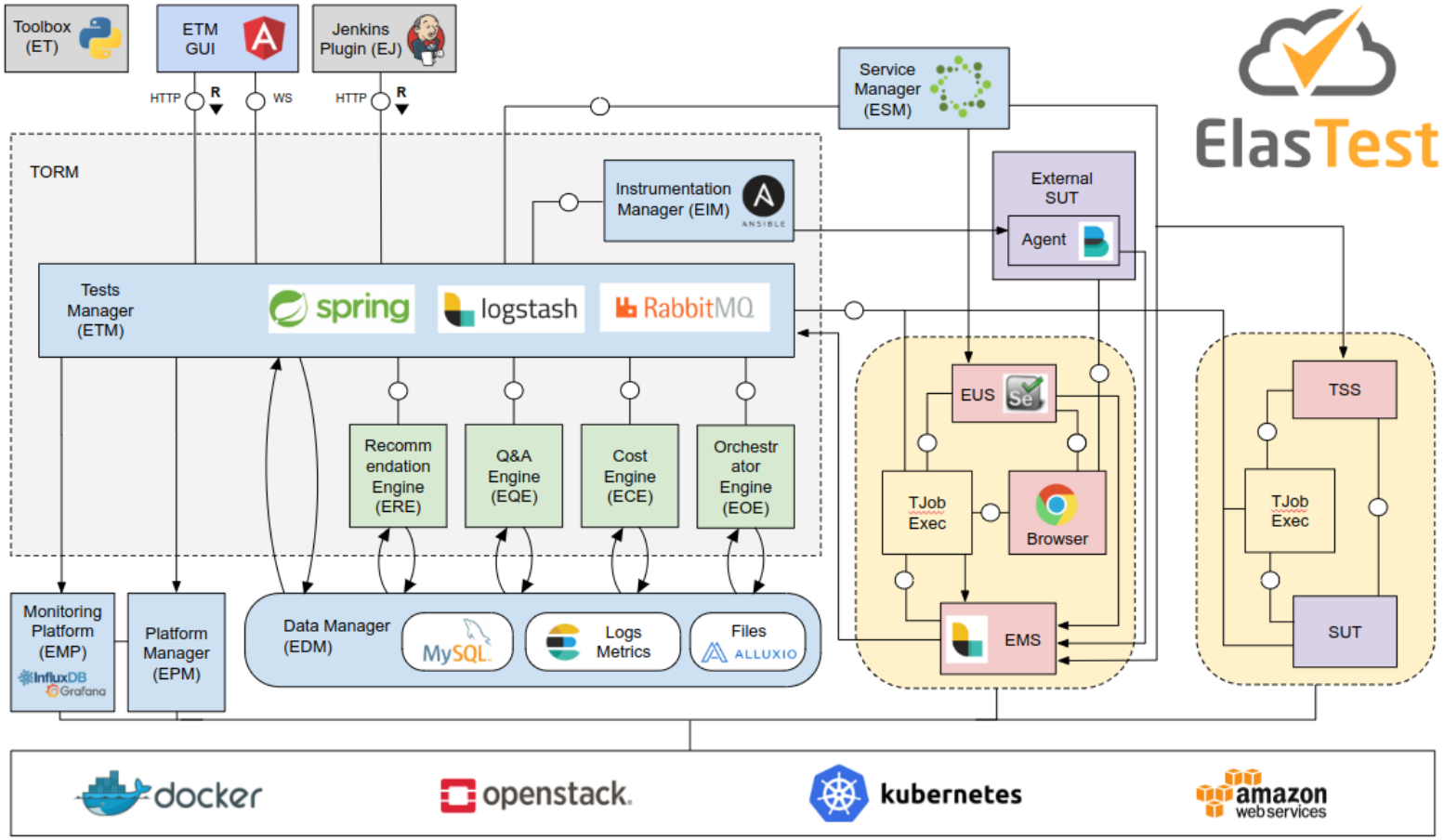


Table of contents

1. Introduction
2. Background
3. ElasTest: platform for end-to-end testing
4. User Impersonation as a Service
5. Demo
6. Conclusions and future work

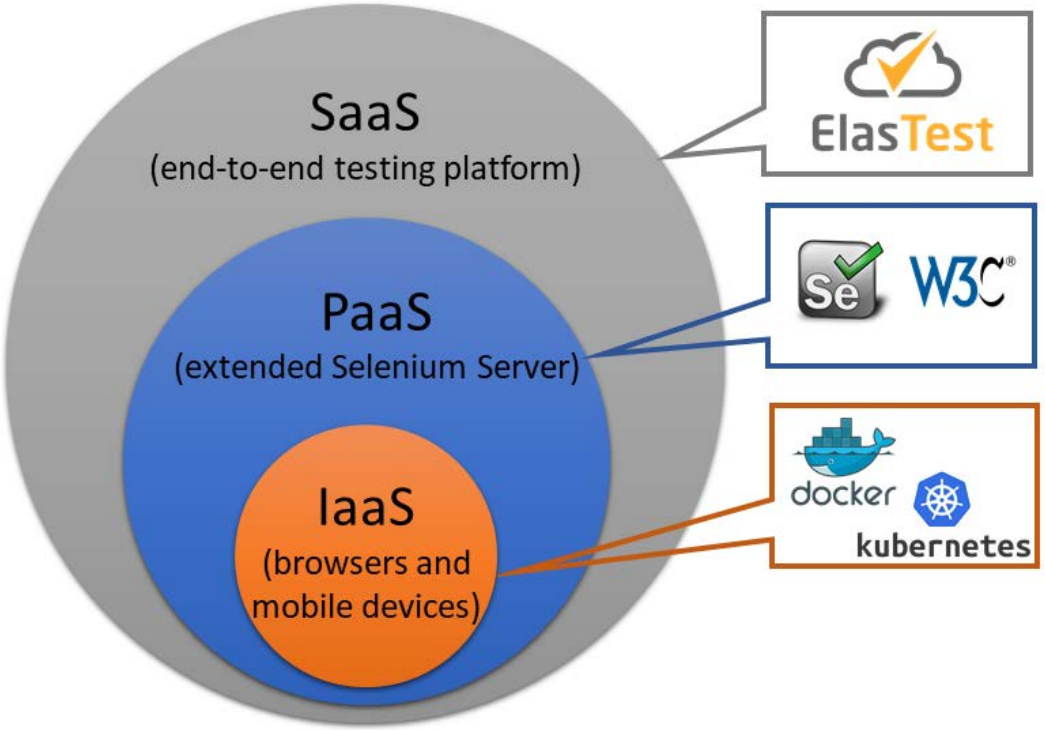
4. User Impersonation as a Service

- ElasTest implements an User Impersonation as a service capability (UIaaS) that provides Software as a Service (SaaS) extending the W3C WebDriver with advanced capabilities:

1. Evaluation of the perceived Quality of Experience (QoE) of users on real-time multimedia applications (WebRTC)
2. Equivalent automation capability for sensors and smart devices for Internet of Things (IoT)

4. User Impersonation as a Service

- Our concept of UIaaS has been implemented as a layered approach following the NIST definition of cloud computing (SaaS, PaaS, and IaaS)



4. User Impersonation as a Service

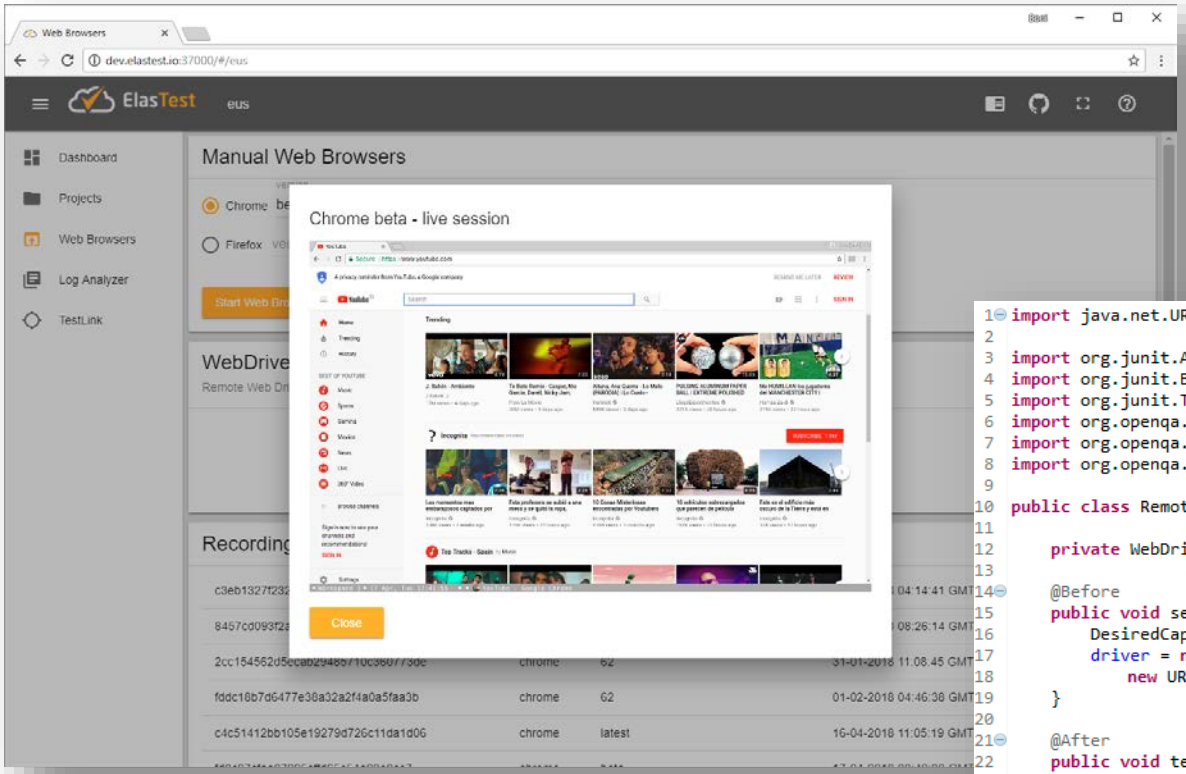
- Extension to W3C WebDriver recommendation by ElasTest User Impersonation Service:

Method	Path	Description
Event subscription		
POST	/session/{sessionId}/element/{elementId}/event	Subscribe to a given event within an element
GET	/session/{sessionId}/event/{subscriptionId}	Read the value of event for a given subscription
DELETE	/session/{sessionId}/event/{subscriptionId}	Remove a subscription
Recordings		
GET	/session/{sessionId}/vnc	Get remote session
DELETE	/session/{sessionId}/vnc	Delete remote session
Media capabilities		
POST	/session/{sessionId}/usermedia	Set user media for WebRTC
GET	/session/{sessionId}/stats	Read the WebRTC stats
POST	/session/{sessionId}/element/{elementId}/latency	Measure end-to-end latency of a WebRTC session
POST	/session/{sessionId}/element/{elementId}/quality	Measure quality of a WebRTC session

Table of contents

1. Introduction
2. Background
3. ElasTest: platform for end-to-end testing
4. User Impersonation as a Service
5. Demo
6. Conclusions and future work

5. Demo




```java
import java.net.URL;

import org.junit.After;
import org.junit.Before;
import org.junit.Test;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
import org.openqa.selenium.remote.RemoteWebDriver;

public class RemoteTest {

    private WebDriver driver;

    @Before
    public void setup() throws Exception {
        DesiredCapabilities capabilities = DesiredCapabilities.chrome();
        driver = new RemoteWebDriver(
            new URL("http://dev.elastest.io:34362/eus/v1/"), capabilities);
    }

    @After
    public void teardown() {
        if (driver != null) {
            driver.quit();
        }
    }

    @Test
    public void test() {
        // My test logic
    }

}
```

Table of contents

1. Introduction
2. Background
3. ElasTest: platform for end-to-end testing
4. User Impersonation as a Service
5. Demo
6. Conclusions and future work

6. Conclusions and future work

- ElasTest is an open source platform aimed to ease end-to-end tests for heterogenous large distributed systems
- ElasTest implements a User Impersonation as a Service (UIaaS) extending the W3C WebDriver recommendation
- Existing test codebases with Selenium and Appium are completely compatible with ElasTest
- Some features are still under development, such as measurement of the end-users' perceived QoE or support for IoT devices