QA TEST

Taming Dino-Frameworks with NSelene

@yashaka

AUTOMICIAN

Preface…

Taming Dino-Frameworks with NSelene

C# Version

Plan

GIVEN project with legacy selenium-based dinosaur-framework

AND problems with stability, boilerplate code

WHEN believe in evolution :)

THEN let's integrate modern solutions like Selenide/NSelene into legacy code

Coding Session

Summary

- Default "implicit waits for visibility" in a few "code symbols"

- New more stable and concise code with modern tool

    - Full interoperability

- Unstable behaviour of legacy code? => refactor needed parts to new style

Full Interoperability. How?

Full Access to original Driver Procedural Approach

DoNewStuff()

GetWebDriver().DoWhateverYouWant()…

Full Access to original Driver OOP Approach

wrappedDriver = new FirefoxDriver();

driver = new SeleneDriver(wrappedDriver);

driver.DoNewStuff()

wrappedDriver.DoWhateverYouWant()...

Interfaces Compatibility

SeleneElement implements IWebElement

SeleneCollection implements IList<IWebElement>, IReadOnlyCollection<IWebElement>, etc.

SeleneDriver implements IWebDriver

Power. How?

- SeleneElement = IWebElement as Dynamic Proxy + Implicit Smart Waits + Handy Asserts

- SeleneCollection = IList<IWebElement> as Dynamic Proxy + Implicit Smart Waits + Handy Asserts

- SeleneDriver = IWebDriver finding SeleneElement & SeleneCollection

What about ![Java] ?

- SelenideElement = WebElement as Dynamic Proxy + Implicit Smart Waits + Handy Asserts

- ElementsCollection ~ List<WebElement> as Dynamic Proxy + Implicit Smart Waits + Handy Asserts

- SelenideDriver **can be implemented by your own**
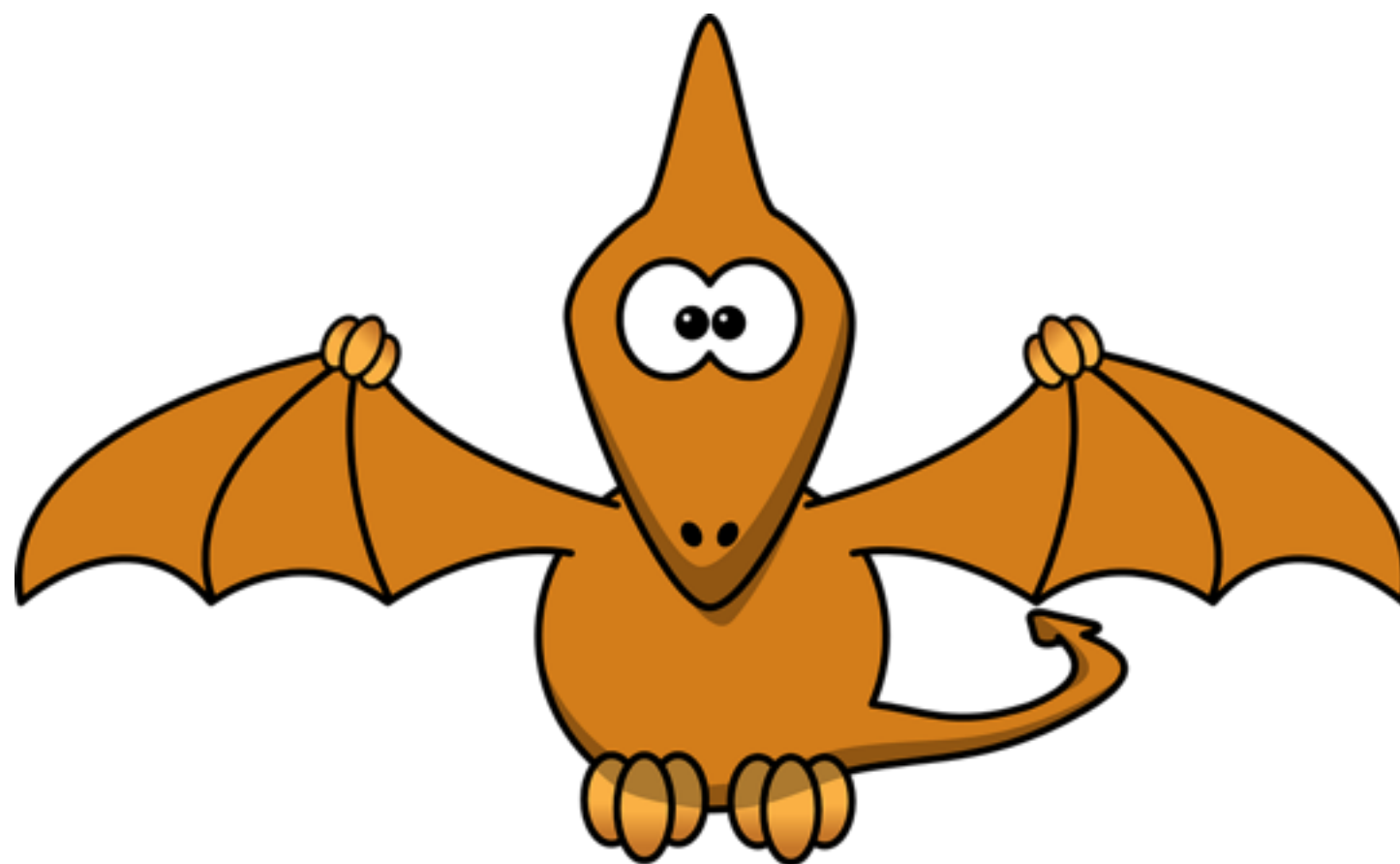
  - **soon will be added to Selenide**

Selenide Examples

- [Default "implicit waits for visibility" in a few "code symbols"](#)

- [New more stable and concise code with modern tool](#)

- [Unstable behaviour of legacy code? => refactor needed parts to new style](#)

CHALLENGE ACCEPTED

Selenide
CONCISE UI TESTS IN JAVA

Q&A

Thank You

github.com/yashaka

github.com/yashaka/nselene

yashaka@gmail.com

facebook.com/yashaka

courses.automician.com

@yashaka

Next

[paid] October 3rd: "Widgets Workshop (Java) (ru)"

[paid] November: "Refactoring selenium waits and PageFactory course (Java) (ru)"

[paid] now: "Refactoring selenium waits and PageFactory videos (c#) (ua)" available at **courses.automician.com**

@yashaka