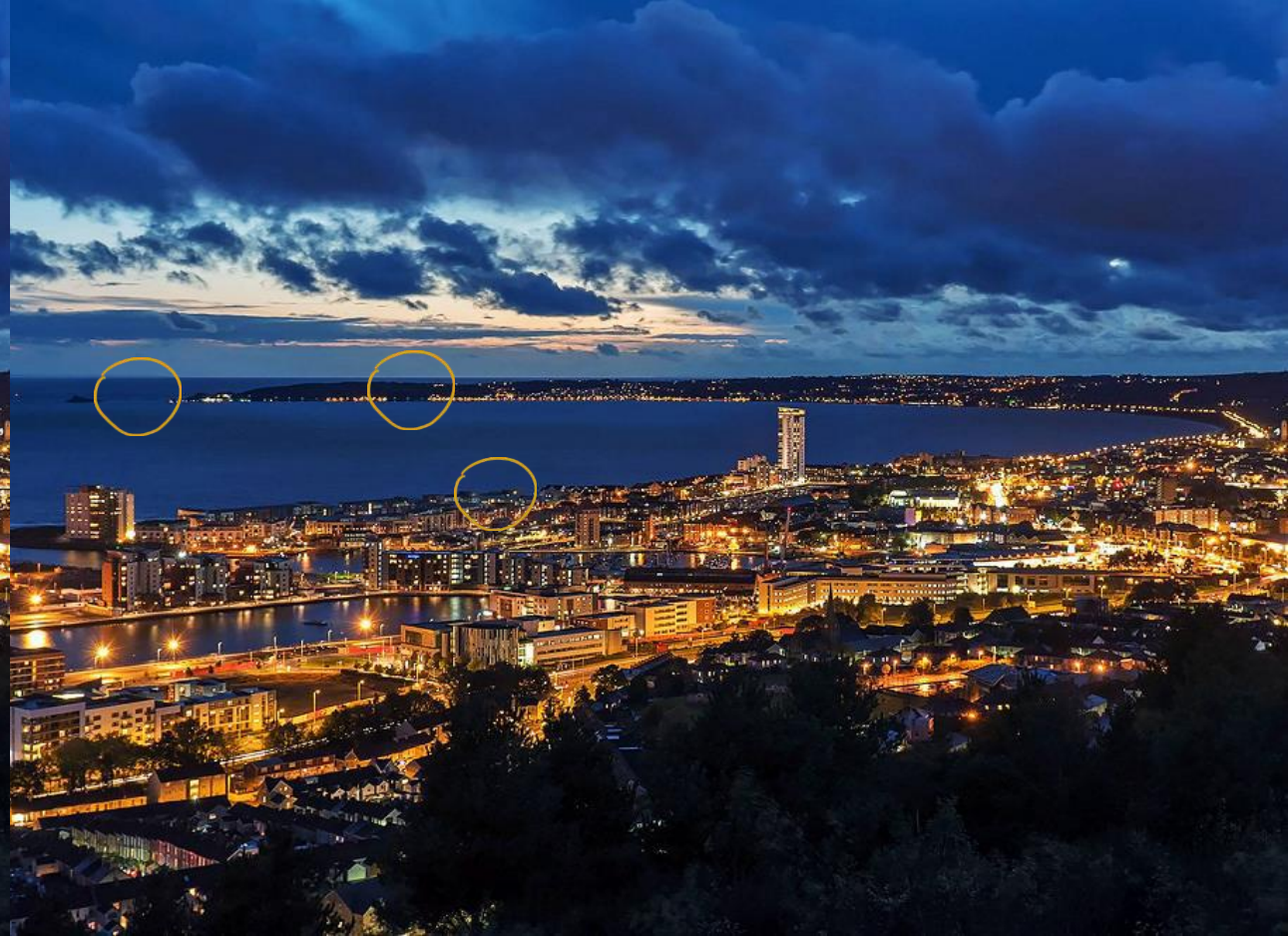
TestCon

Spot the difference

Automating visual regression testing

@11vlr



TestCon

Spot the difference

Automating visual regression testing

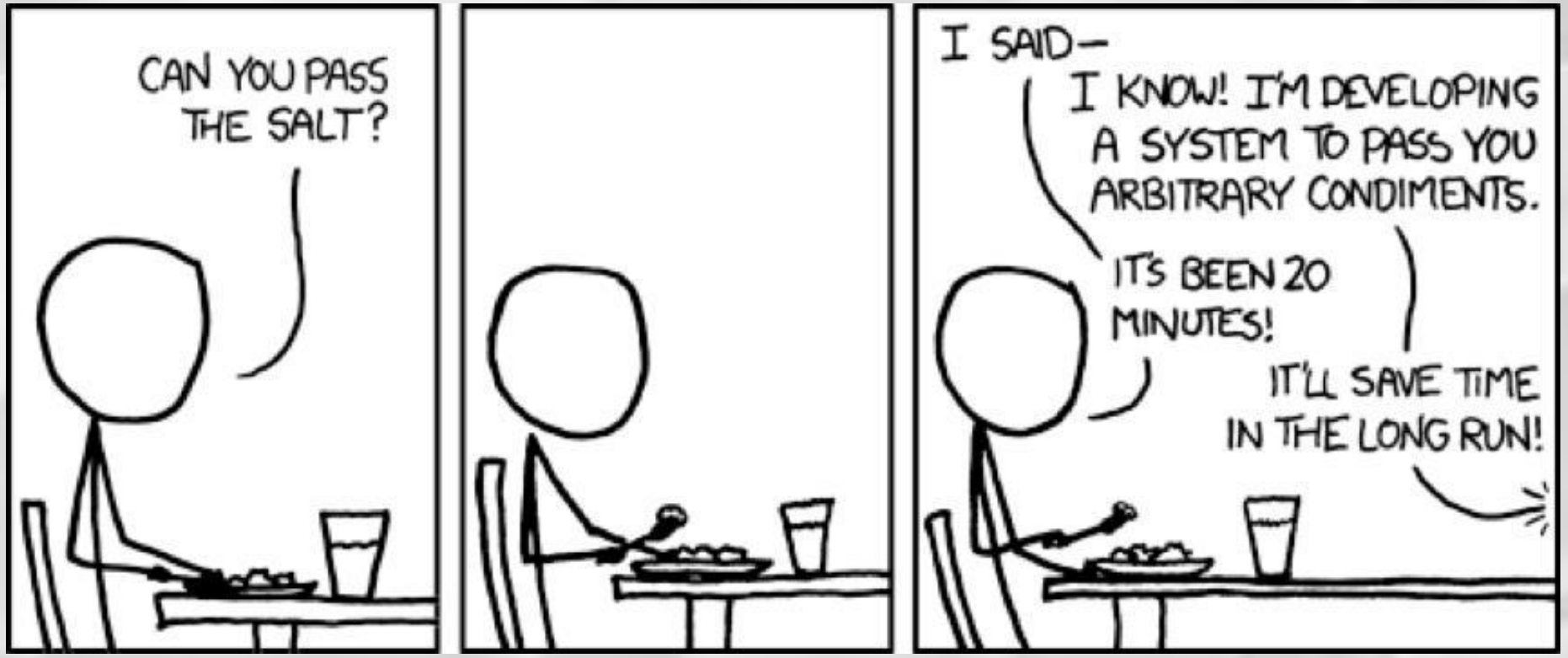
@11vlr

WHY AUTOMATE END TO END UI TESTS?

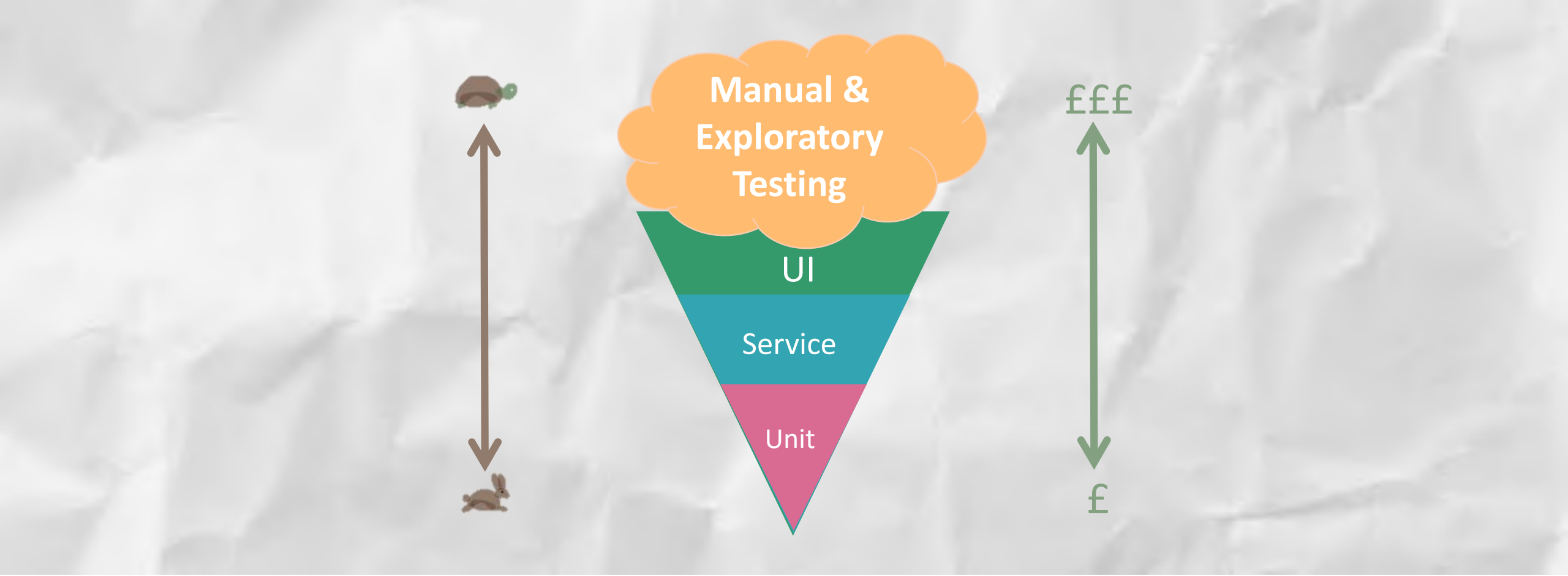
- Automated testing is more reliable, as it is performed by tools and/or scripts.
- Automated testing is executed by software tools, so it is significantly faster than a manual approach.

COMMON UI AUTOMATION ISSUES

- Investment is required
- Automated testing is a practical option when the test cases are run repeatedly over a long time period.
- Automated testing **DOES NOT** entail human observation and cannot guarantee user-friendliness or positive customer experience.

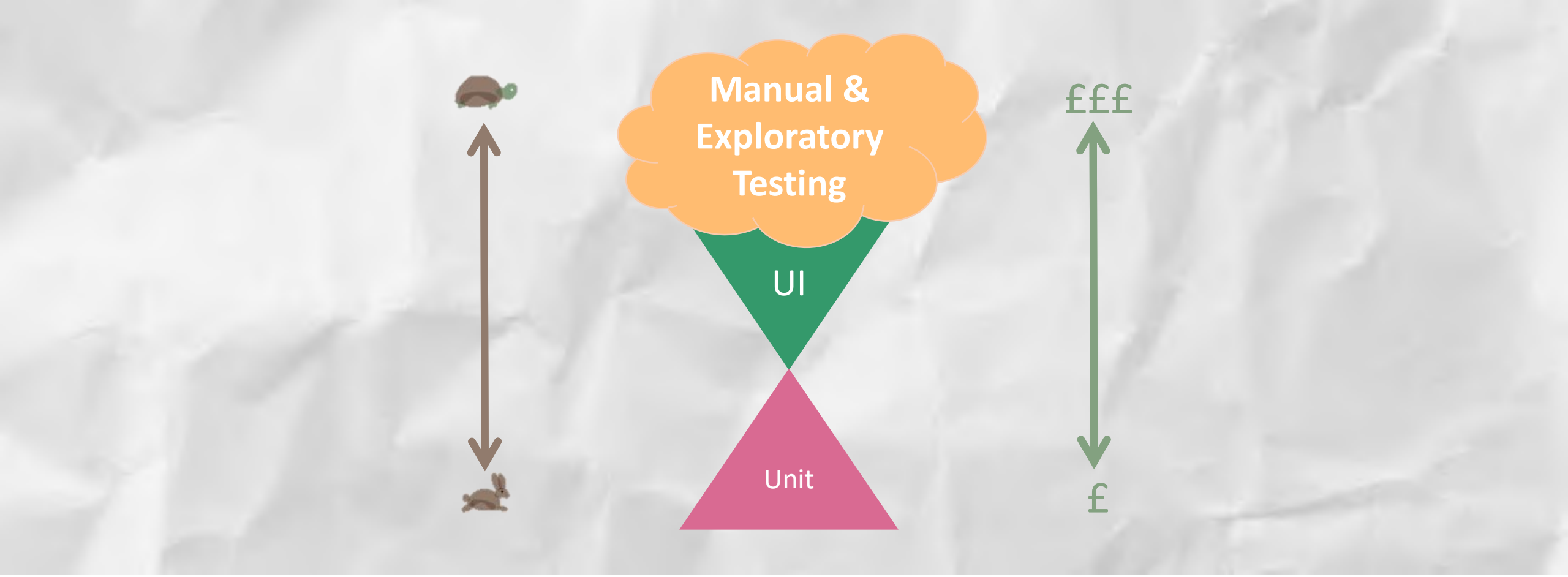


IT CAN BE A SLIPPERY SLOPE



ICE-CREAM CONE ANTI-PATTERN

Adaption of https://watirmelon.blog/testing-pyramids/



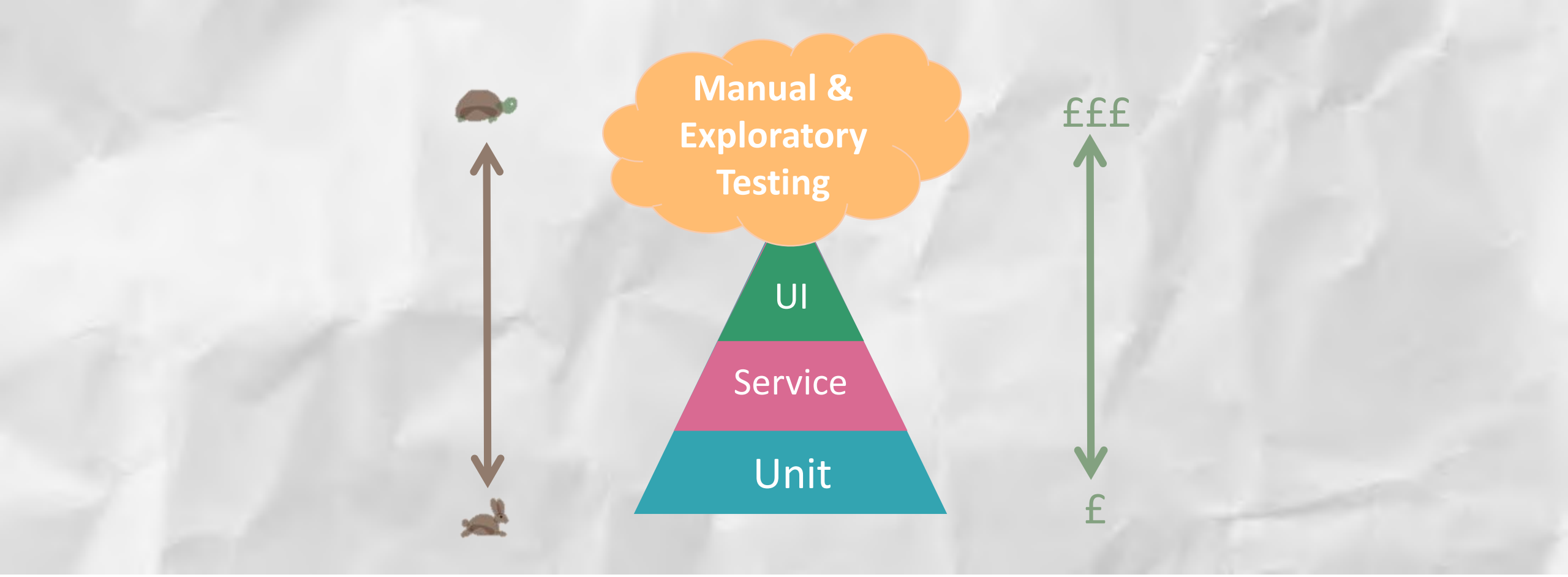
HOUR GLASS ANTI-PATTERN

Adaption of: https://www.getautoma.com/blog/the-test-hourglass

WHEN YOU HAVE UNIT BUT NO INTEGRATION TESTS

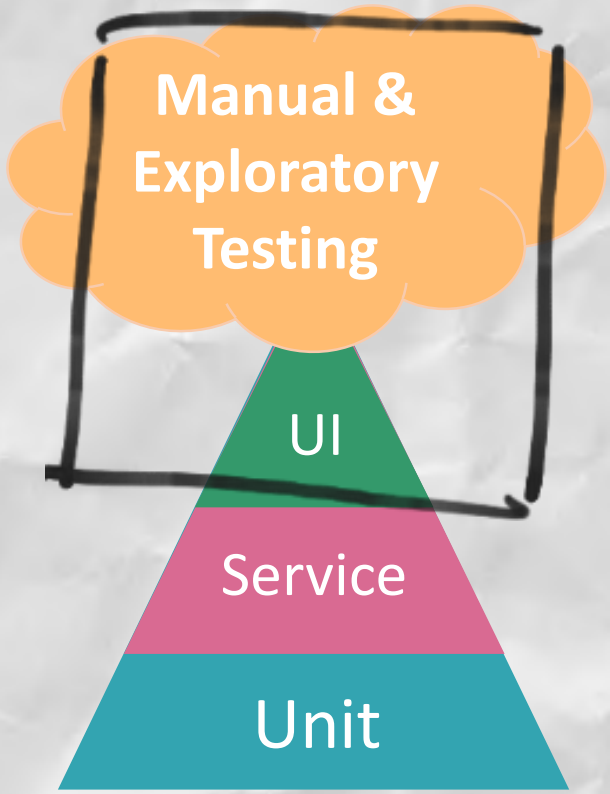
frAgile TESTING

@11vlr



TESTING PYRAMID

Adaption of https://martinfowler.com/bliki/TestPyramid.html



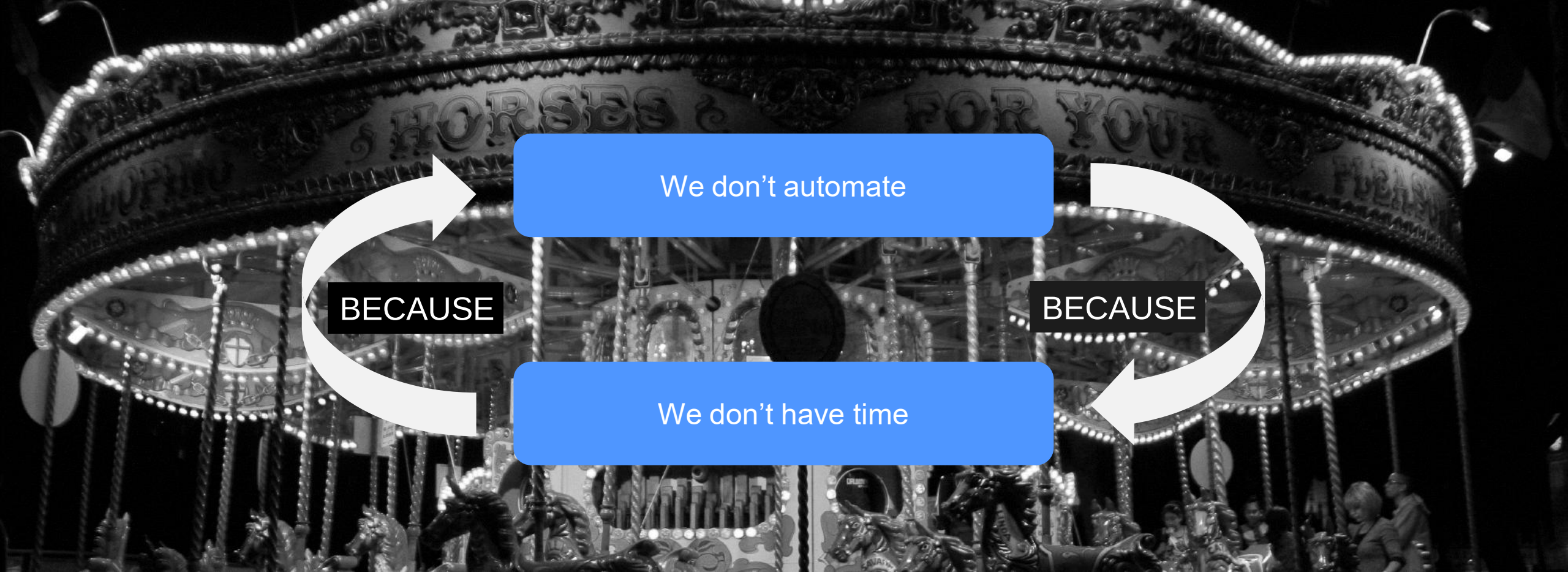
WHY NOT JUST STICK WITH MANUAL TESTING THEN?

- Manual testing allows for human observation, which may be more useful if the goal is user-friendliness or improved customer experience.

MANUAL TESTING

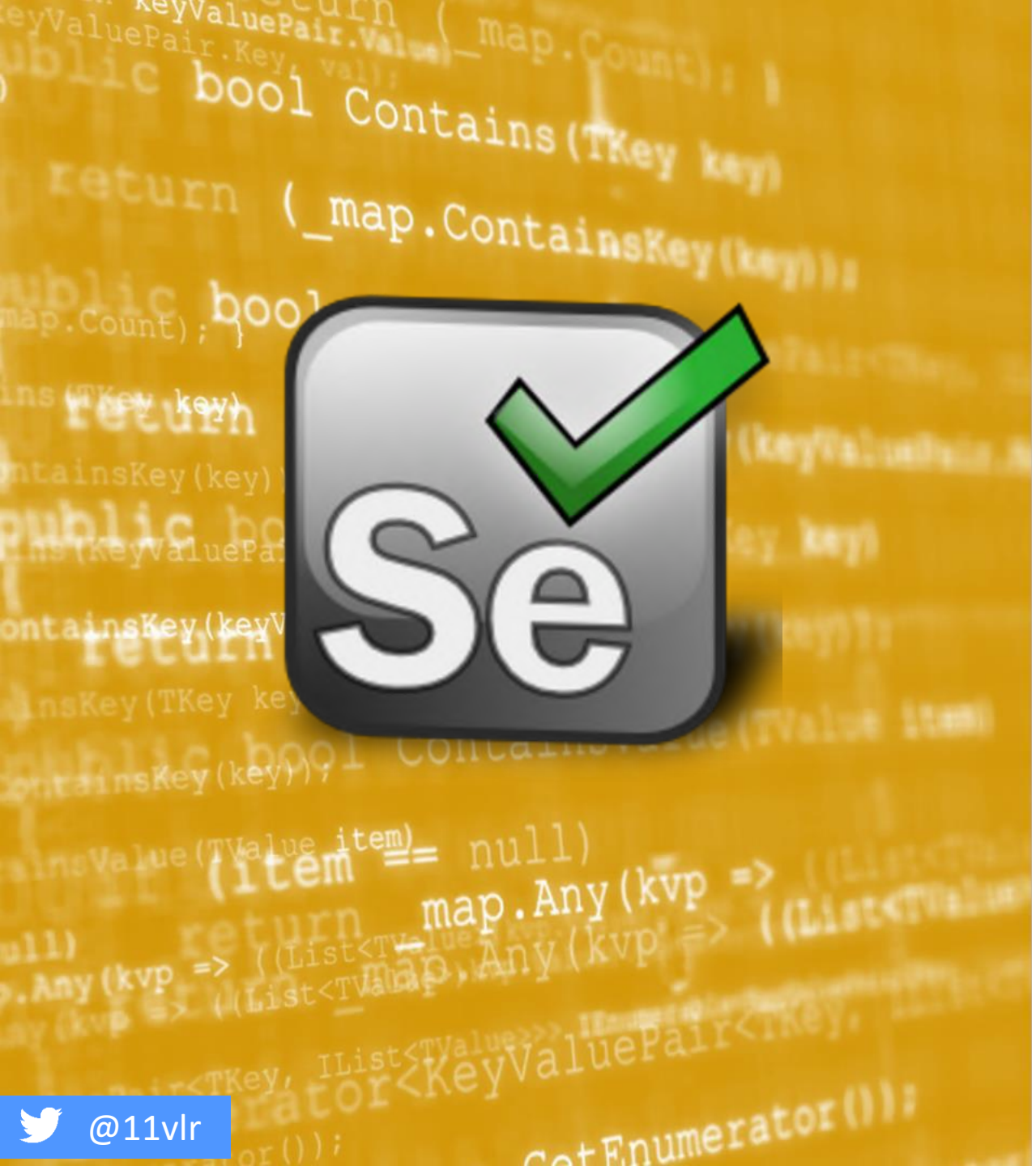
- Can be time intensive
- Is not accurate at all times due to human error, hence it is less reliable.
- Is time-consuming, taking up human resources.
- Is only practical when the test cases are run once or twice, and frequent repetition is not required.

WE DON’T HAVE TIME TO AUTOMATE



CATCH 22

MANY OPTIONS



SELENIUM

- Open source; anybody can see the code and help
- Big community of support
- Can operate on almost any OS
- Multiple language support (Python, Ruby**, .Net**, PHP, Java, JavaScript)
- Supports a range of browsers like Opera, **Chrome**, **IE** (6,7,8,9,10,11), **Mozilla Firefox**, Safari

IT’S NOT A SILVER BULLET

BROWSER COMPATIBILITY ISSUES

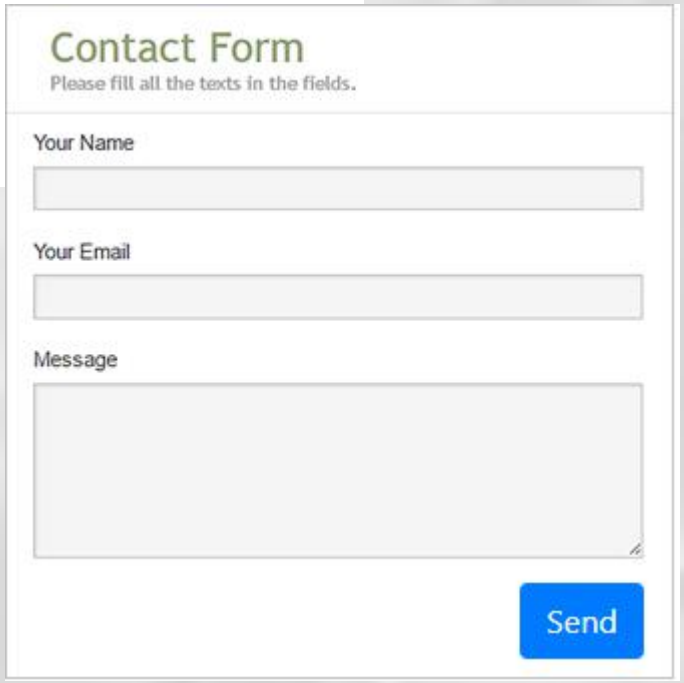
- Speed to execute tests
- Render differently

EXECUTION SPEED

MAINTAINABILITY

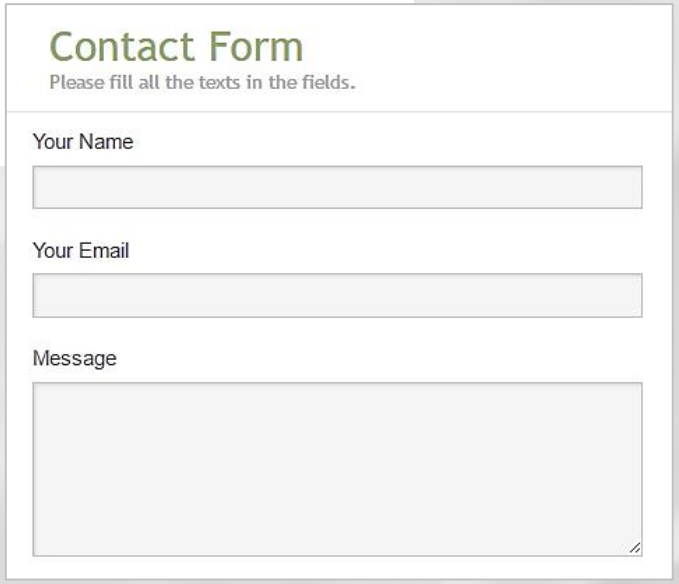
HOW VALUABLE ARE THESE TESTS?

```
// Assert text on form is as expected
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Contact Form"));
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Please fill all the texts in the fields."));
Assert.IsTrue(Driver.FindElement(By.Id("nameLabel")).Text.Equals("Your Name"));
Assert.IsTrue(Driver.FindElement(By.Id("emailLabel")).Text.Equals("Your Email"));
Assert.IsTrue(Driver.FindElement(By.Id("messageLabel")).Text.Equals("Message"));

// Assert button text is as expected
Assert.IsTrue(Driver.FindElement(By.Id("sendButton")).GetAttribute("value").Equals("Send"));
```



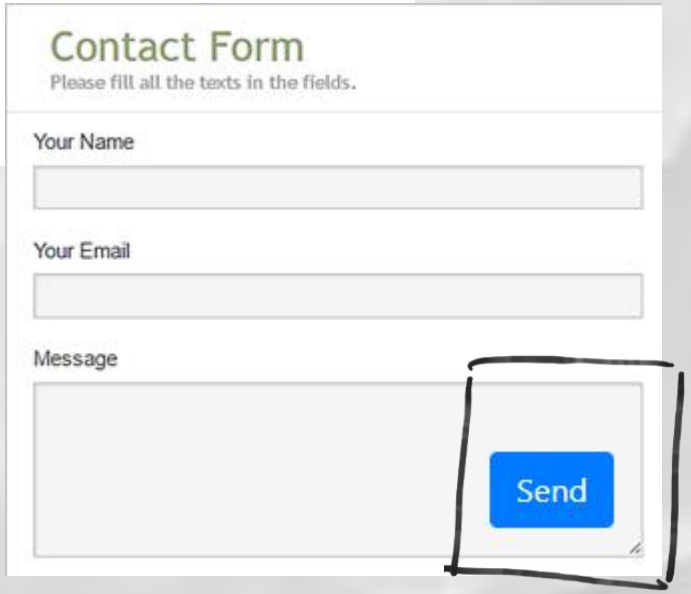
WHAT THE….

```
// Assert text on form is as expected
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Contact Form"));
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Please fill all the texts in the fields."));
Assert.IsTrue(Driver.FindElement(By.Id("nameLabel")).Text.Equals("Your Name"));
Assert.IsTrue(Driver.FindElement(By.Id("emailLabel")).Text.Equals("Your Email"));
Assert.IsTrue(Driver.FindElement(By.Id("messageLabel")).Text.Equals("Message"));

// Assert button text is as expected
Assert.IsTrue(Driver.FindElement(By.Id("sendButton")).GetAttribute("value").Equals("Send"));
```



ALL IS NOT WHAT IT SEEMS

```
// Assert text on form is as expected
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Contact Form"));
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Please fill all the texts in the fields."));
Assert.IsTrue(Driver.FindElement(By.Id("nameLabel")).Text.Equals("Your Name"));
Assert.IsTrue(Driver.FindElement(By.Id("emailLabel")).Text.Equals("Your Email"));
Assert.IsTrue(Driver.FindElement(By.Id("messageLabel")).Text.Equals("Message"));

// Assert button text is as expected
Assert.IsTrue(Driver.FindElement(By.Id("sendButton")).GetAttribute("value").Equals("Send"));
```



@11vlr

MAKING TESTS LESS FLAKY

@11vlr

MAKING TESTS LESS FLAKY

- Page object patterns

http://martinfowler.com/bliki/PageObject.html

@11vlr

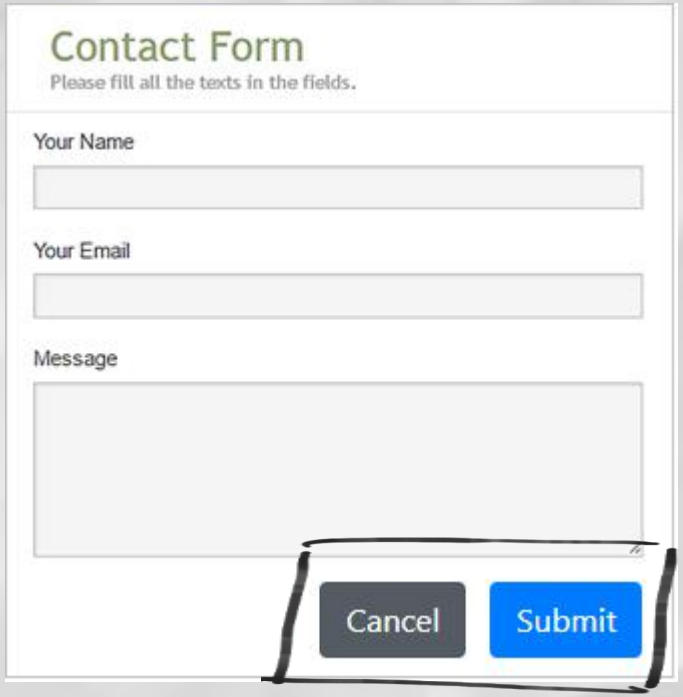
MAKING TESTS LESS FLAKY

- Qa-selectors

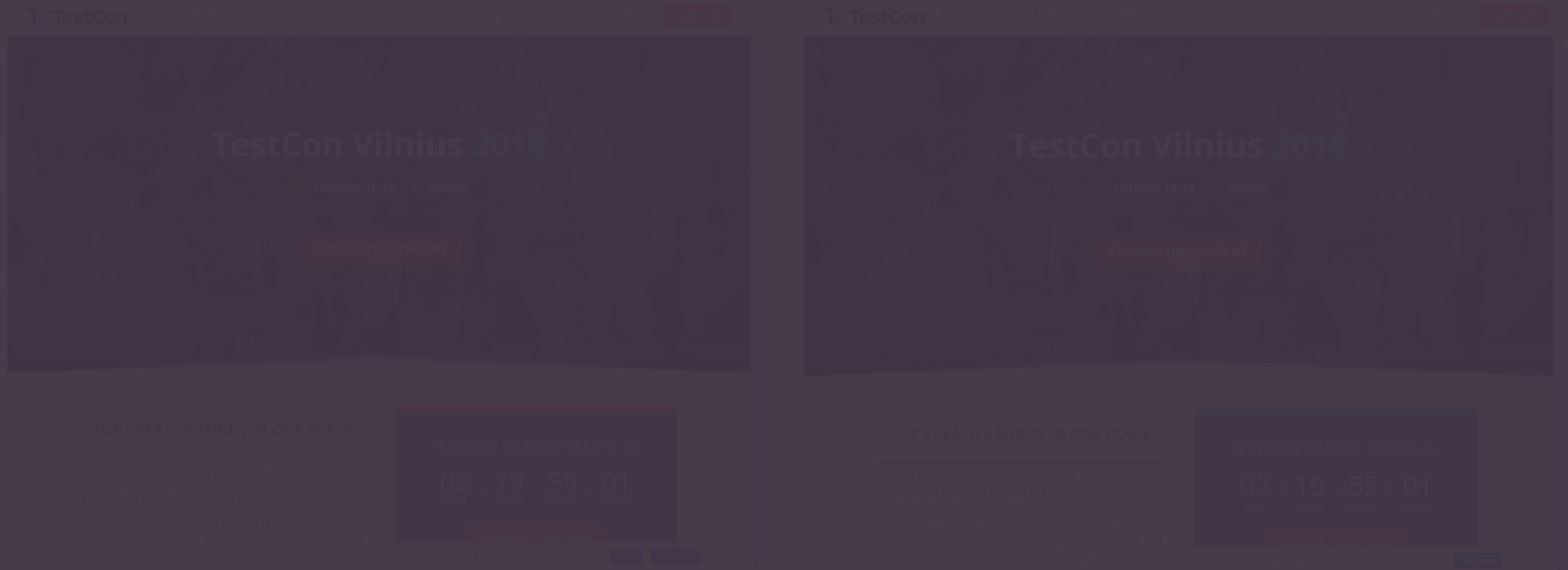
Qa- selectors

```
<label for="fname">Your Name</label>
<input type="text" id="fname" name="name">

<label for="lemail">Your Email</label>
<input type="text" id="lemail" name="email">

<label for="message">Message</label>
<textarea id="lmessage" name="message"></textarea>

<input type="button" class="button" value="Cancel">
<input type="submit" class="button" value="Submit">
```

```
<label for="fname">Your Name</label>
<input type="text" id="fname" name="name">

<label for="lemail">Your Email</label>
<input type="text" id="lemail" name="email">

<label for="message">Message</label>
<textarea id="lmessage" name="message"></textarea>

<input type="button" class="qa-button-cancel" value="Cancel">
<input type="submit" class="qa-button-submit" value="Submit">
```



HOW GOOD ARE YOU AT SPOT THE DIFFERENCE



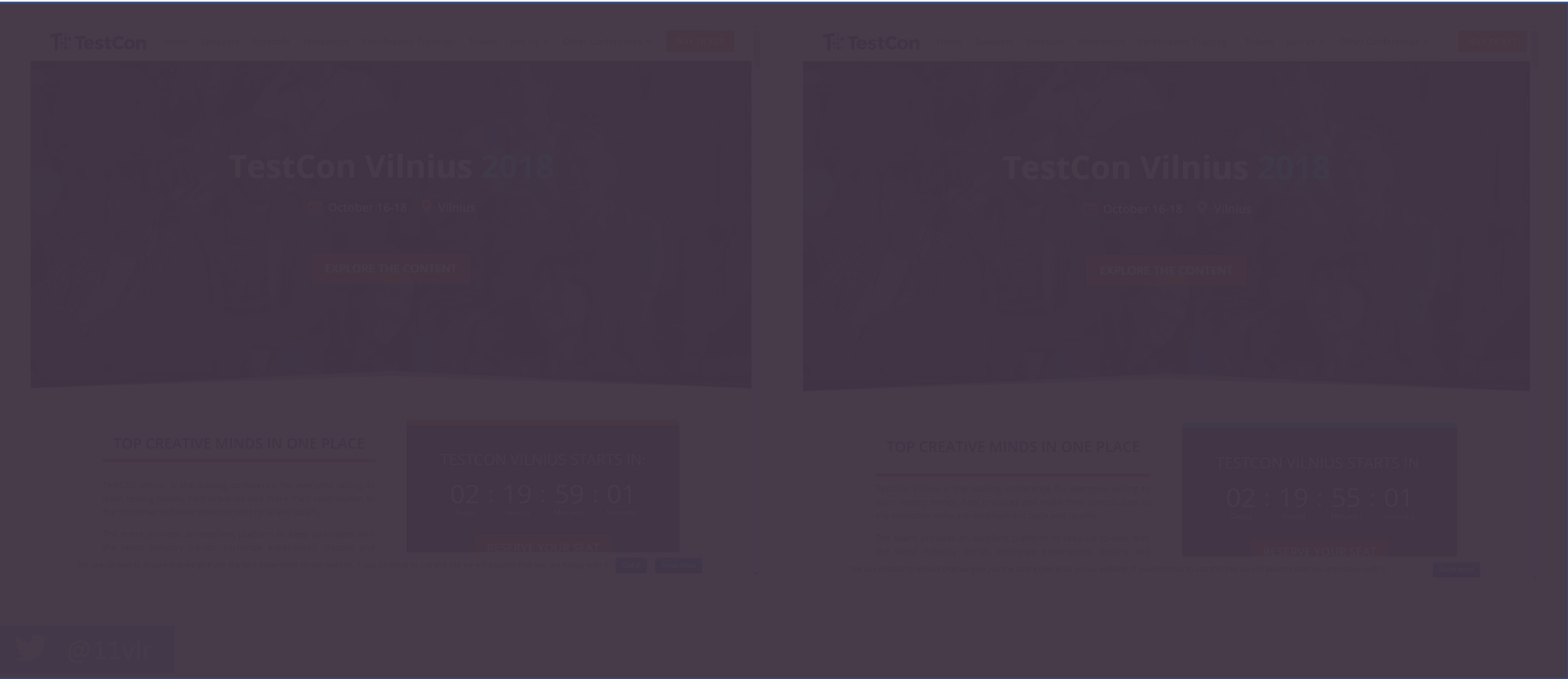
@11vlr

@11vlr

HOW MANY DIFFERENCES CAN YOU FIND?

@11vlr

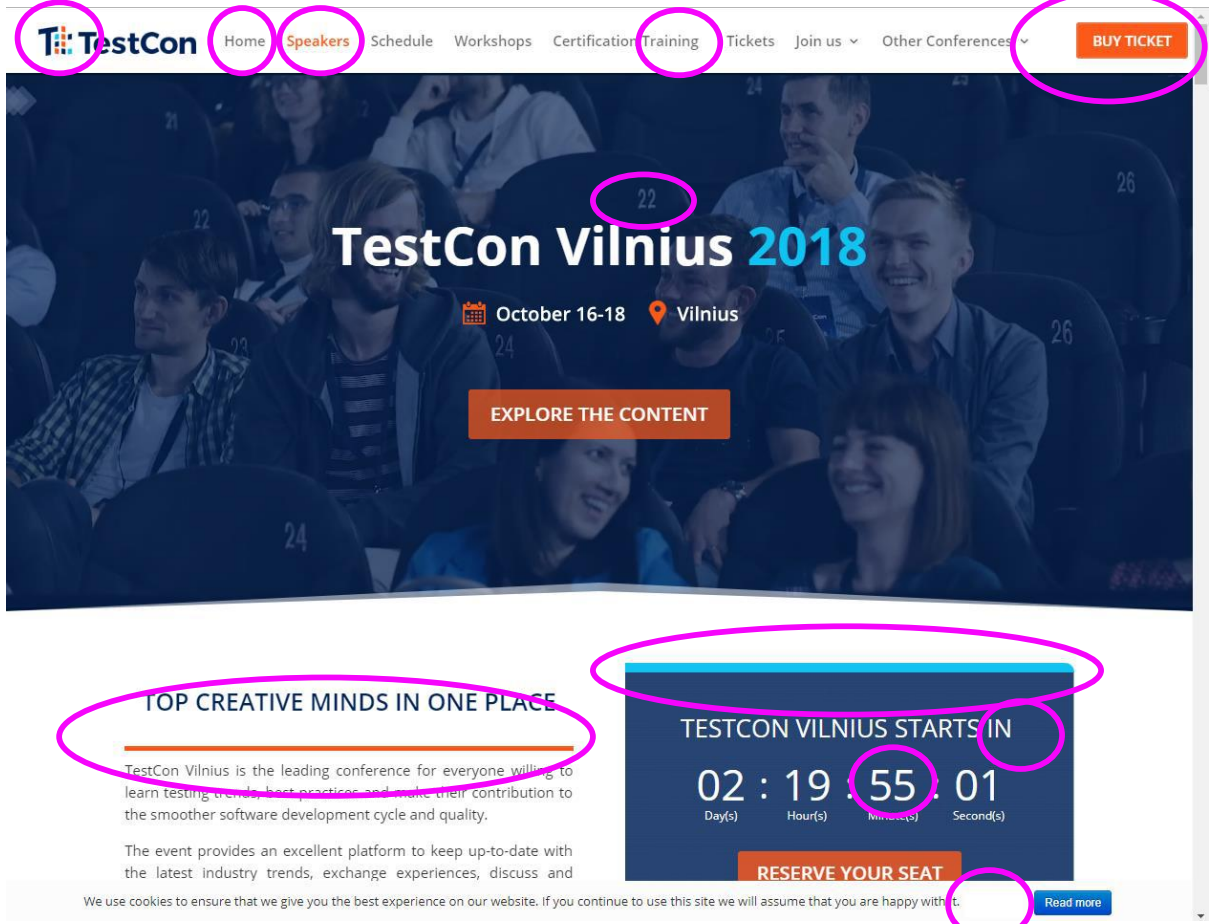
HOW MANY DIFFERENCES DID YOU FIND?



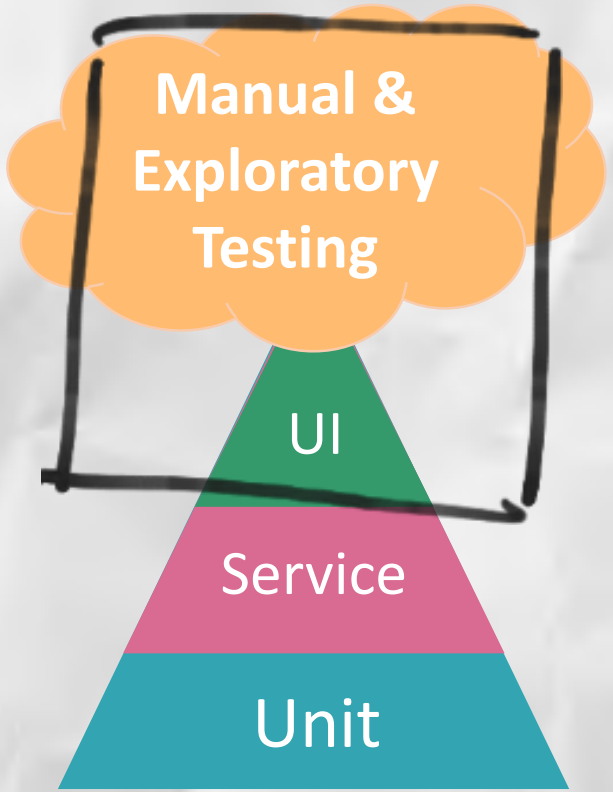
@11vlr

@11vlr

THERE WERE 11 DIFFERENCES



HOW DO YOU KNOW WHEN YOU'VE FOUND THEM ALL?



CAN WE ADD MORE VALUE TO OUR TESTS?

WHAT IF WE COULD SIMPLY

Assert

equals

VISUAL TESTING

SCREENSTER

BackstopJS

Ghost Inspector

applitools
AI powered Visual Testing & Monitoring

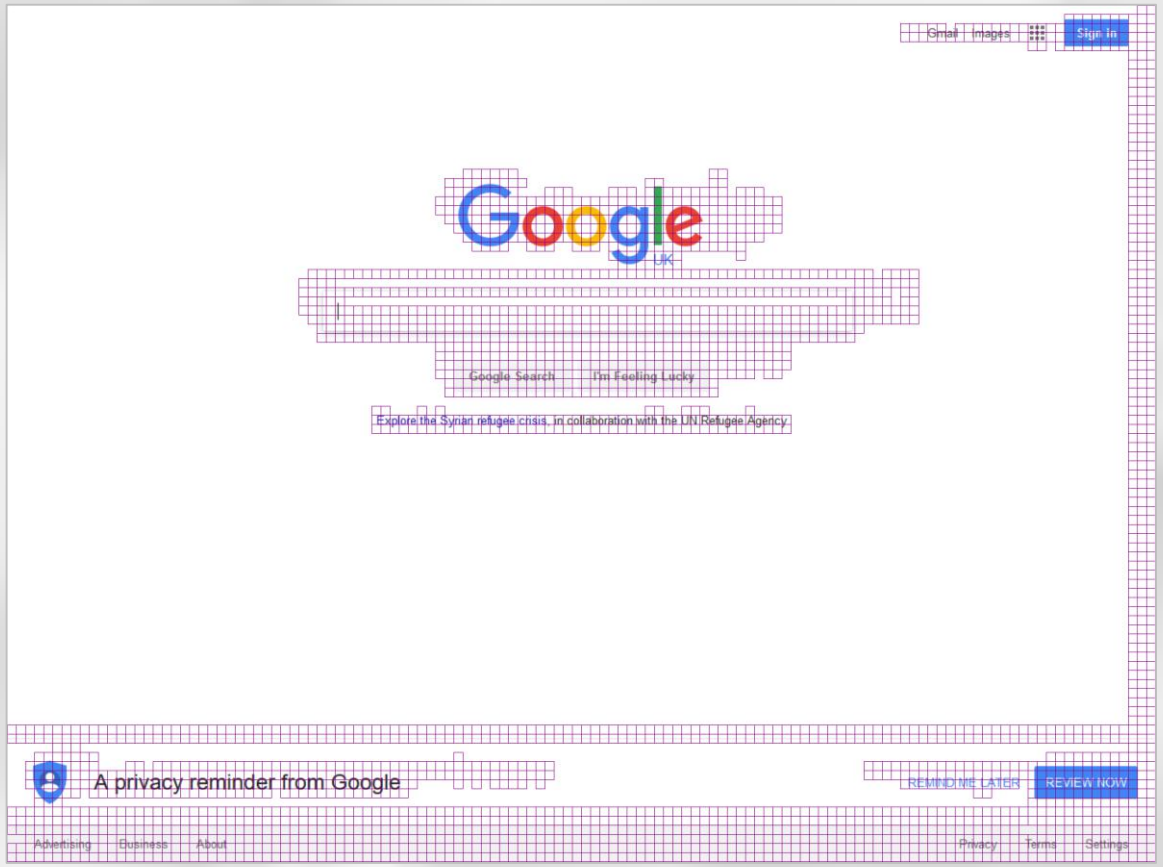
@11vlr

ISSUES WITH OTHER OFFERINGS

- Didn’t work straight out of the box and/or contained lots of bloat
- Steep learning curve
- Small community so lack of support
- Needed to work with our current automation framework
- Costs

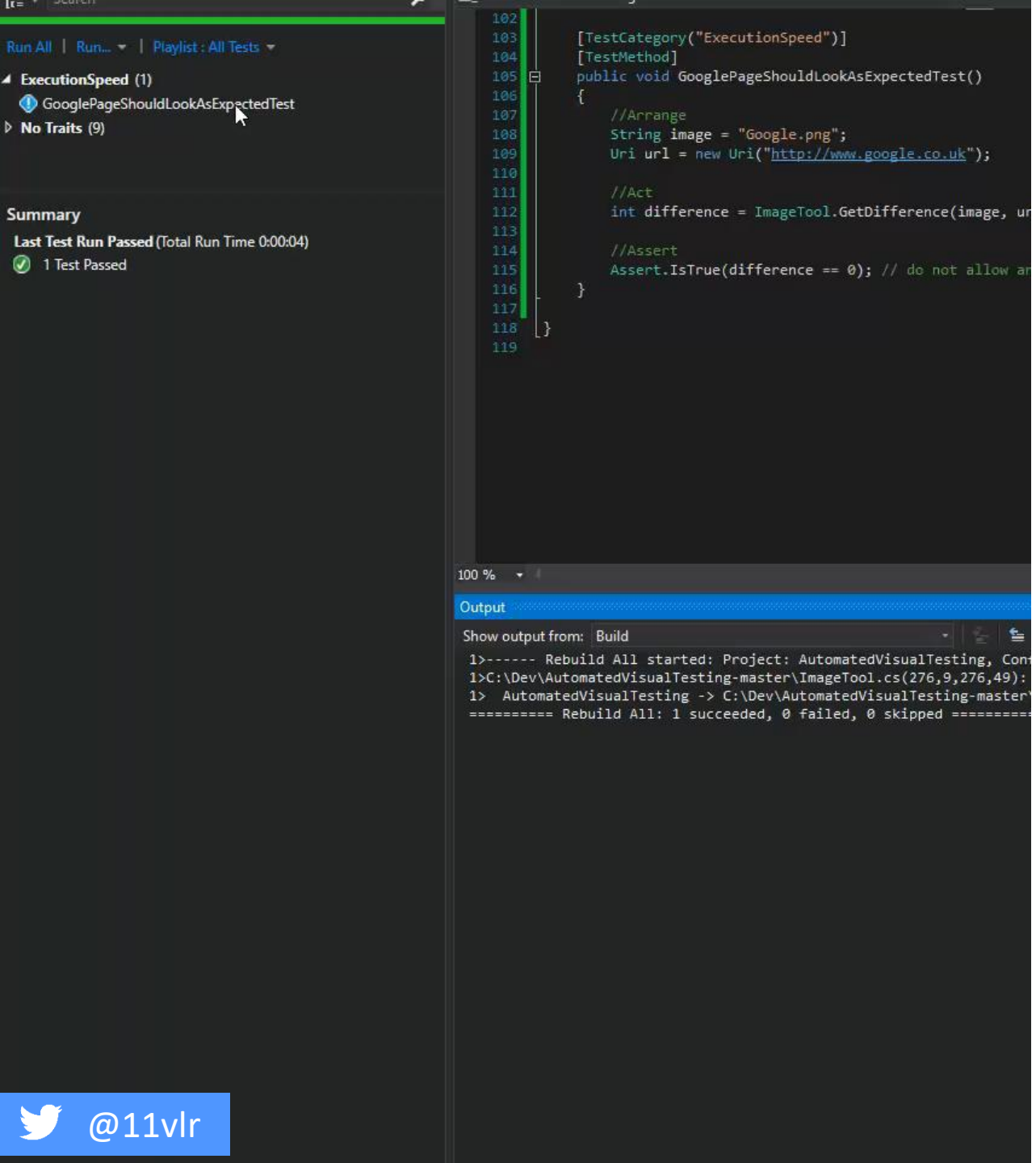
ROLLING OUR OWN

- Been in development 1 year
- Facing many challenges which are because of the products we offer, the ways in which we have developed them and the way in which we test them



RELIABILITY

- Browser rendering issues



EXECUTION SPEED

@11vlr

HUMAN VS MACHINE

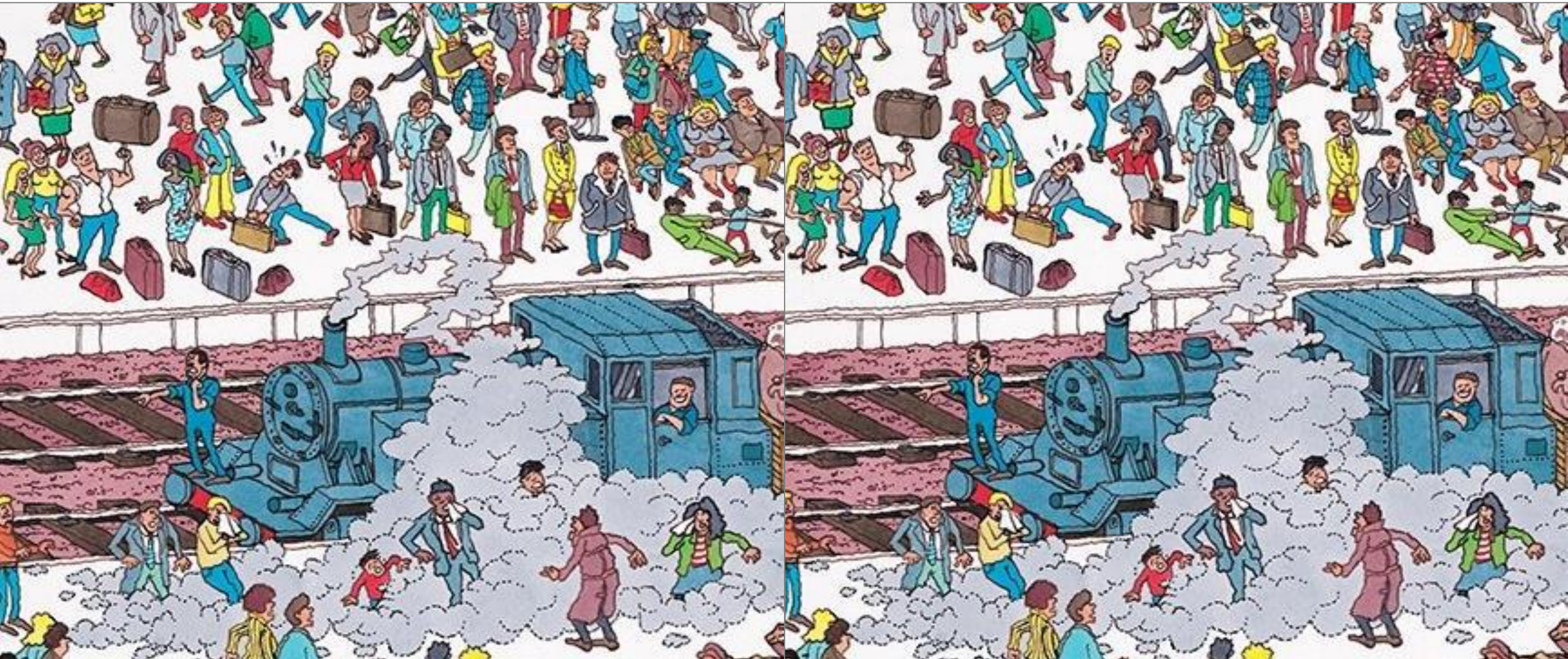
HUMAN VS MACHINE

HUFF, PUFF. WALLY SPOTTERS!
I LOVE RAILWAY STATIONS.
DON'T YOU? THERE WAS SO
MUCH GOING ON TODAY –
COWS ESCAPING FROM TRUCKS,
A MAN ASLEEP BETWEEN THE
RAILS, DOGS CAUSING ALL
SORTS OF TROUBLE!
IT WAS REALLY WILD!
Wally
TO:
WALLY SPOTTERS,
ACROSS THE SEA,
DOWN THE ROAD,
ROUND THE BEND.

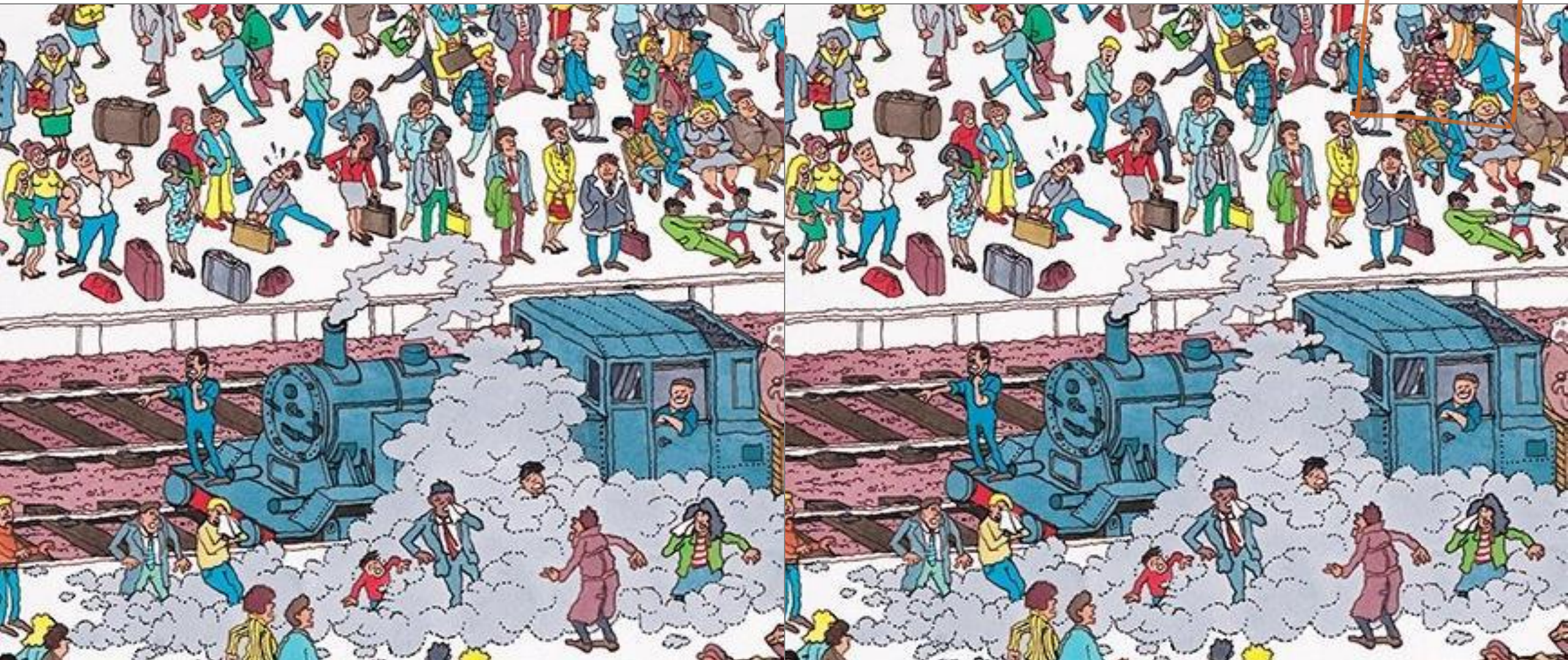
@11vlr

LETS ZOOM IN A LITTLE

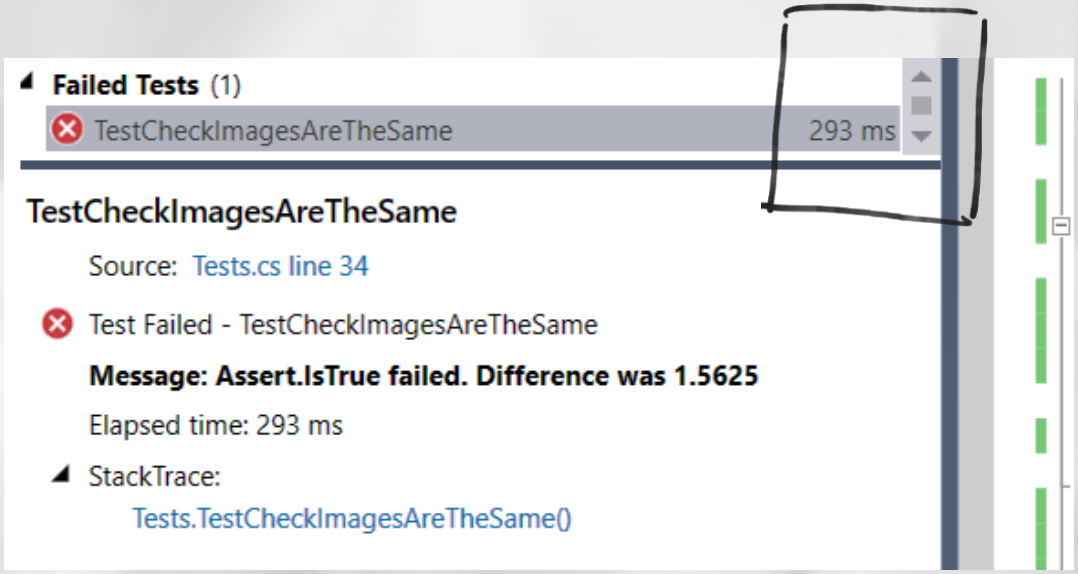
LETS ZOOM IN A LITTLE BIT MORE…



GOT HIM!



HOW QUICK WAS THE MACHINE?

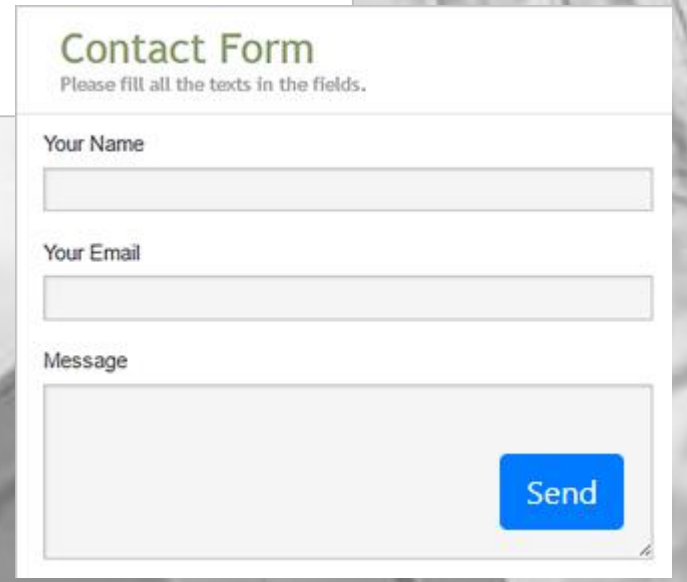



```
String imageWithWally = "Wally.png";
String imageWithNoWally = "NoWally.png";

int difference = GetDifference(imageWithWally, imageWithNoWally);

Assert.IsTrue(difference == 0);
```

MAINTAINABILITY

Old code

```
// Assert text on form is as expected
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Contact Form"));
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Please fill all the texts in the fields."));
Assert.IsTrue(Driver.FindElement(By.Id("nameLabel")).Text.Equals("Your Name"));
Assert.IsTrue(Driver.FindElement(By.Id("emailLabel")).Text.Equals("Your Email"));
Assert.IsTrue(Driver.FindElement(By.Id("messageLabel")).Text.Equals("Message"));

// Assert button text is as expected
Assert.IsTrue(Driver.FindElement(By.Id("sendButton")).GetAttribute("value").Equals("Send"));
```



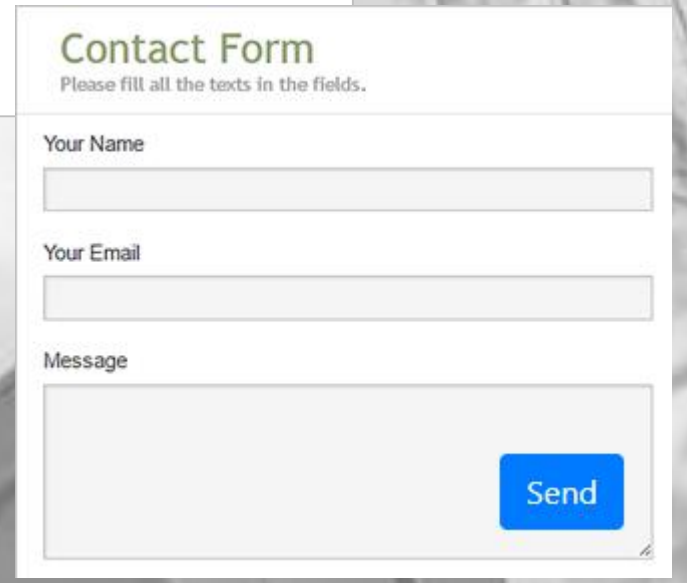
MAINTAINABILITY

Old code

```
// Assert text on form is as expected
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Contact Form"));
Assert.IsTrue(Driver.FindElement(By.CssSelector("h1")).Text.Contains("Please fill all the texts in the fields."));
Assert.IsTrue(Driver.FindElement(By.Id("nameLabel")).Text.Equals("Your Name"));
Assert.IsTrue(Driver.FindElement(By.Id("emailLabel")).Text.Equals("Your Email"));
Assert.IsTrue(Driver.FindElement(By.Id("messageLabel")).Text.Equals("Message"));

// Assert button text is as expected
Assert.IsTrue(Driver.FindElement(By.Id("sendButton")).GetAttribute("value").Equals("Send"));
```

New code using some helpers

```
//Arrange
String image = "ContactPage.png";
Uri url = new Uri("http://www.mypage.co.uk/contact-form");

//Act
int difference = GetDifference(image, url);

//Assert
Assert.IsTrue(difference == 0); // do not allow any difference
```



New code using Specflow

```
@PageComparison
Scenario: home page is rendered as expected
    When the url 'www.homepage.co.uk' is visited
    Then the page should match the 'HomePage.png' base image

@PageComparison
Scenario: contact page is rendered as expected
    When the url 'www.homepage.co.uk/contact' is visited
    Then the page should match the 'Contact.png' base image
```

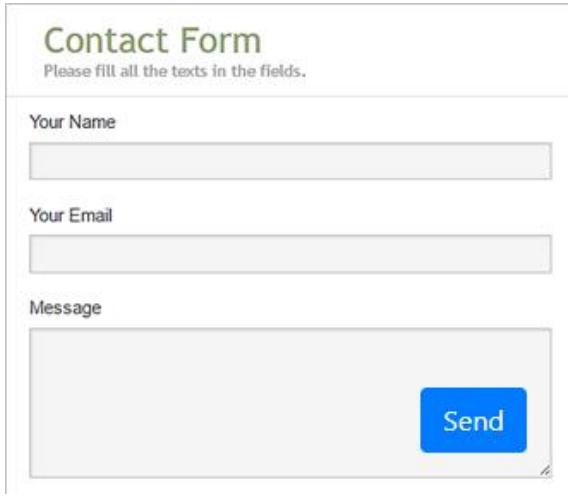
```
[When(@"the url '(.*)' is visited")]
public void WhenTheURLIsVisited(string url)
{
    //Navigate to the given url
    Driver.Navigate().GoToUrl(url);
}

[Then(@"the page should match the '(.*)' base image")]
public void ThenThePageShouldMatchTheBaseImage(string baseImage)
{
    //Get Image of currently loaded web page
    Image imageFromCurrentPage = GetCurrentPageAsImage();

    //Check for differences
    int difference = GetDifference(baseImage, imageFromCurrentPage);

    //assert no differences found
    Assert.IsTrue(difference == 0);
}
```

@11vlr

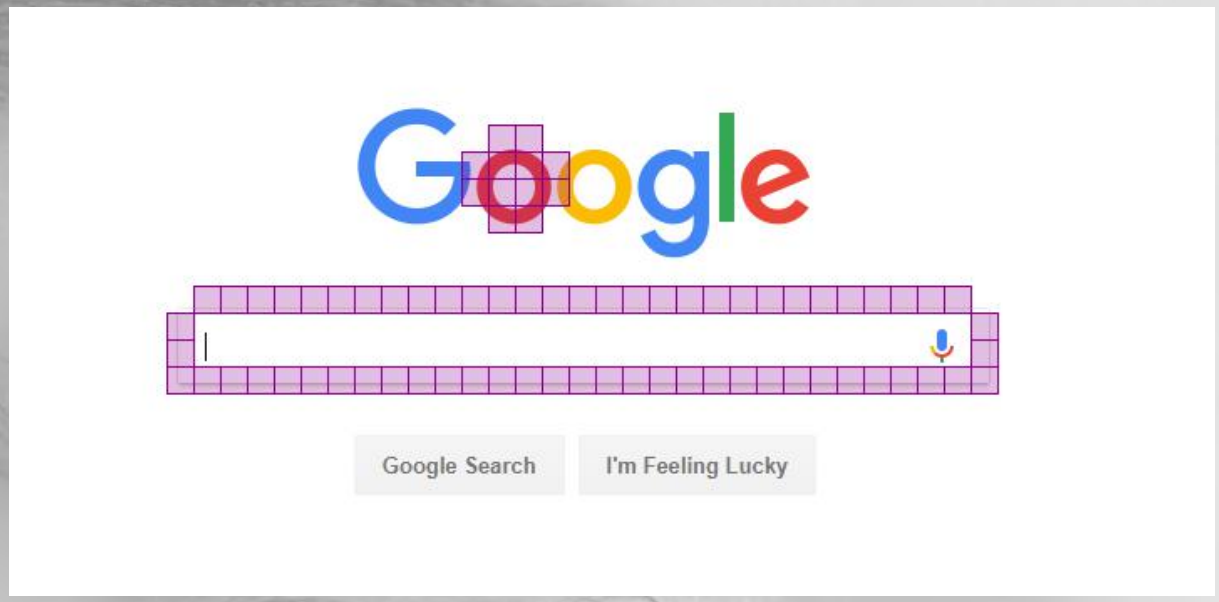
MAINTAINABILITY

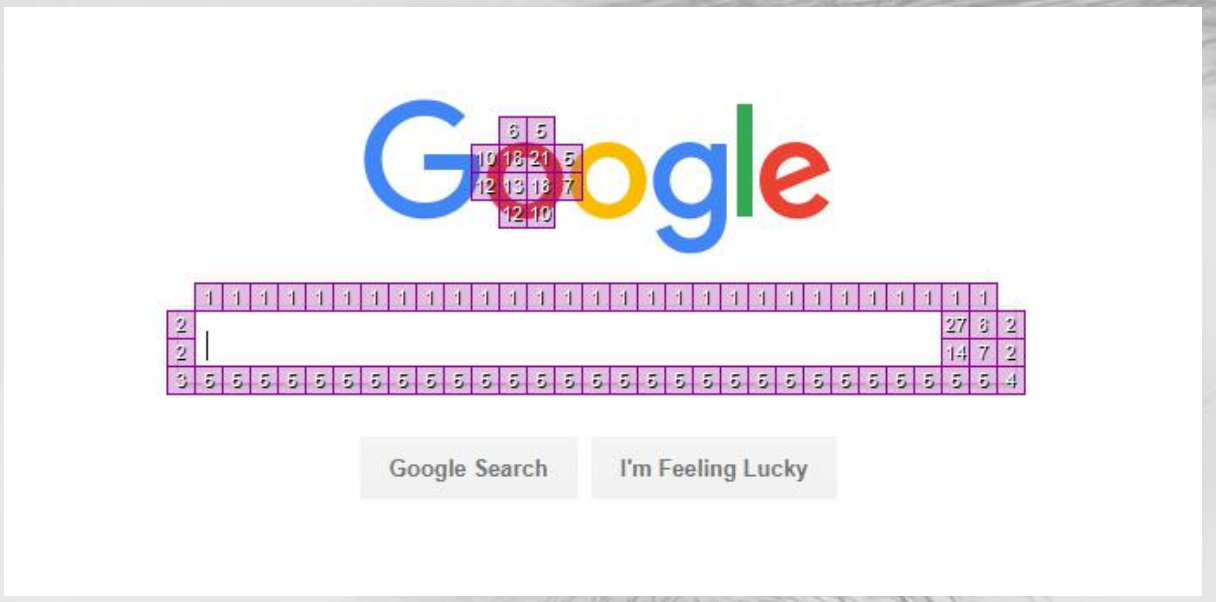
New code using some helpers

```
//Arrange
String image = "ContactPage.png";
Uri url = new Uri("http://www.mypage.co.uk/contact-form");

//Act
int difference = GetDifference(image, url);

//Assert
Assert.IsTrue(difference == 0); // do not allow any difference
```



EASILY IDENTIFY THE DIFFERENCES





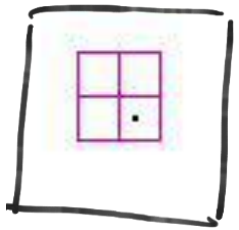
ACCURACY

Create a Meetup | Login | Sign-up

Pembrokeshire Software Development Community

Location

Pembroke Dock, United Kingdom



Members

272

Organisers

Elliot W. and 1 other

Join us

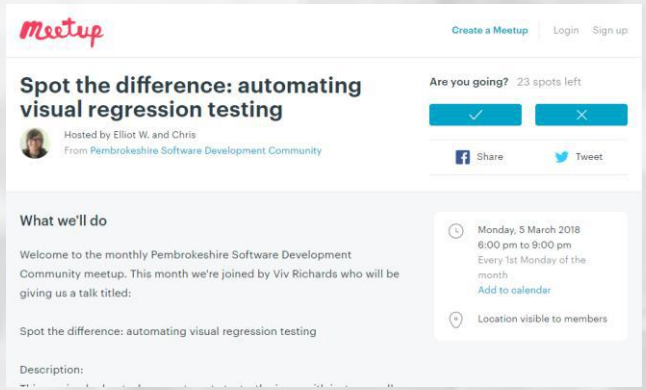
IMAGE QUALITY DIFFERENCE

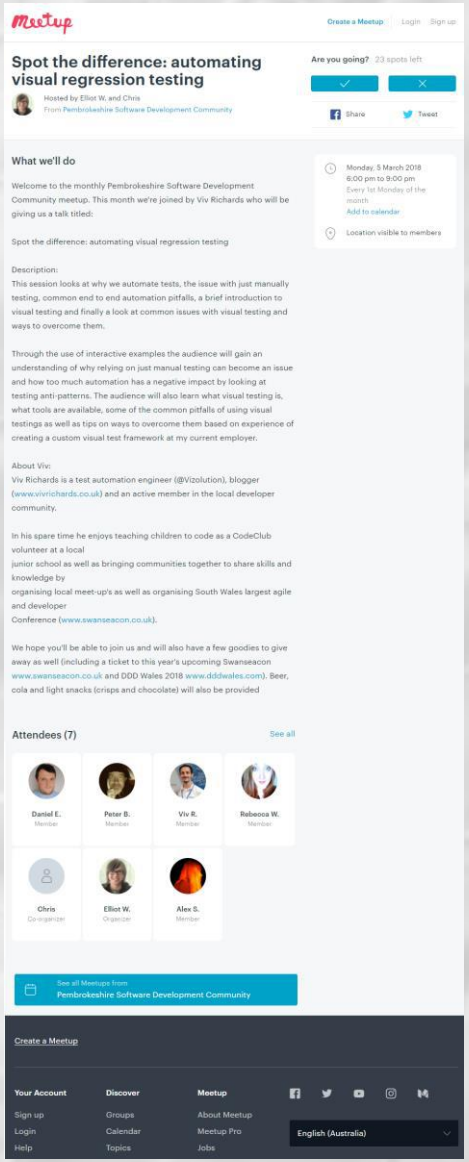
CHALLENGES SO FAR

@11vlr

MAINTAINING BASE IMAGES

Keep a base image for each driver type





SCREENSHOTS

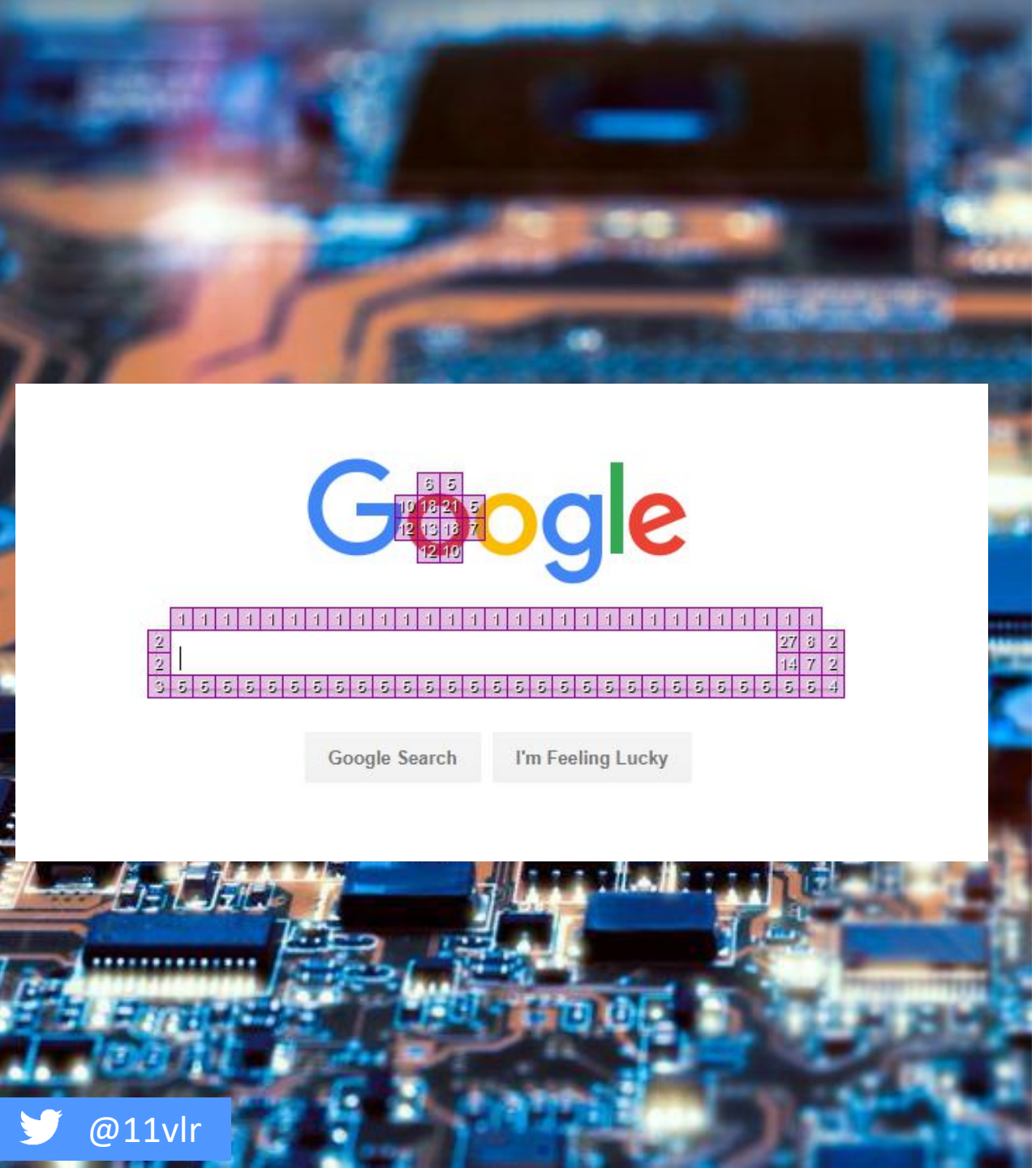
- Viewport (default - WebDriver W3C standard)
- Full Screen (stitching)
- Capture images of just elements

@11vlr

SCREEN RESOLUTIONS

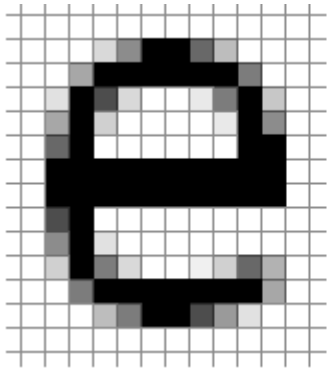
SCREEN RESOLUTIONS

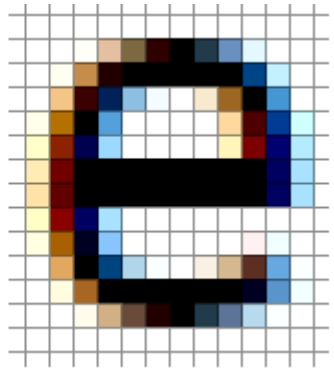
Responsive design



HARDWARE ISSUES

Machine hardware rendering snapshots slightly differently

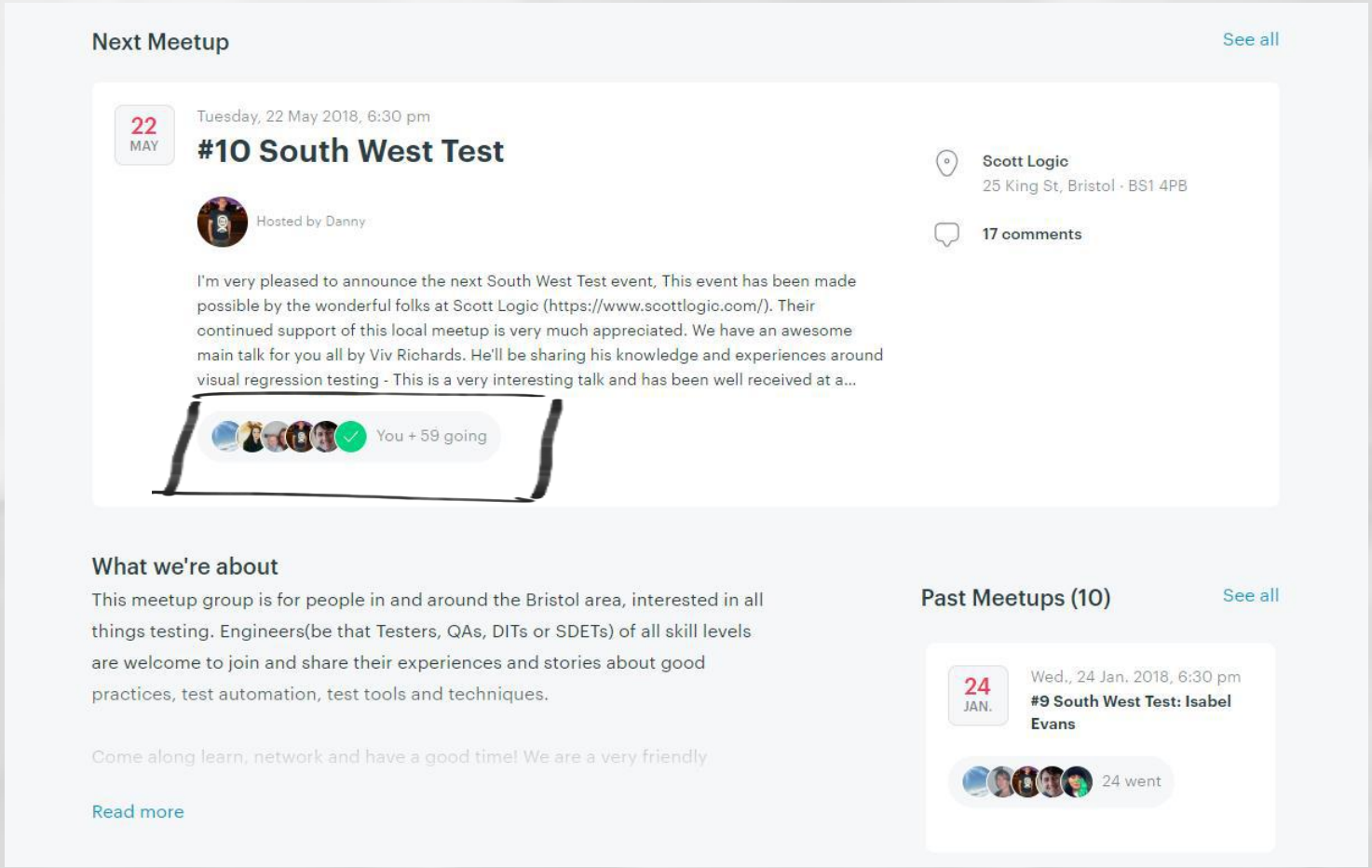




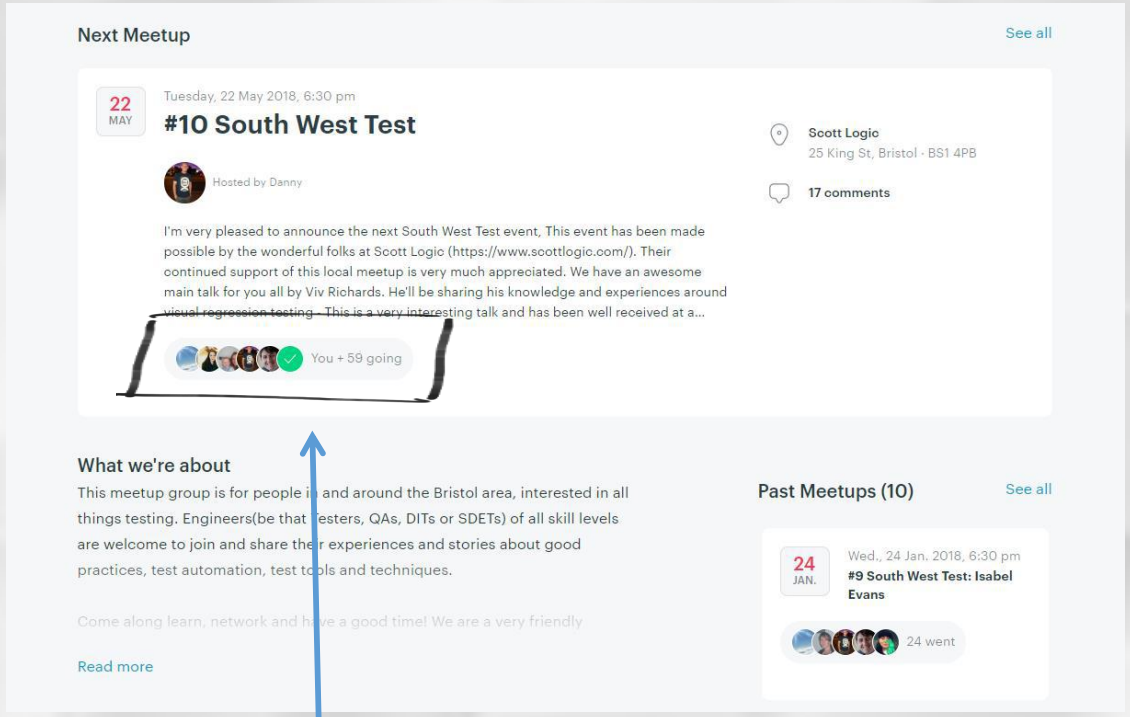
STORAGE

Database / Online storage – file sizes (many images)

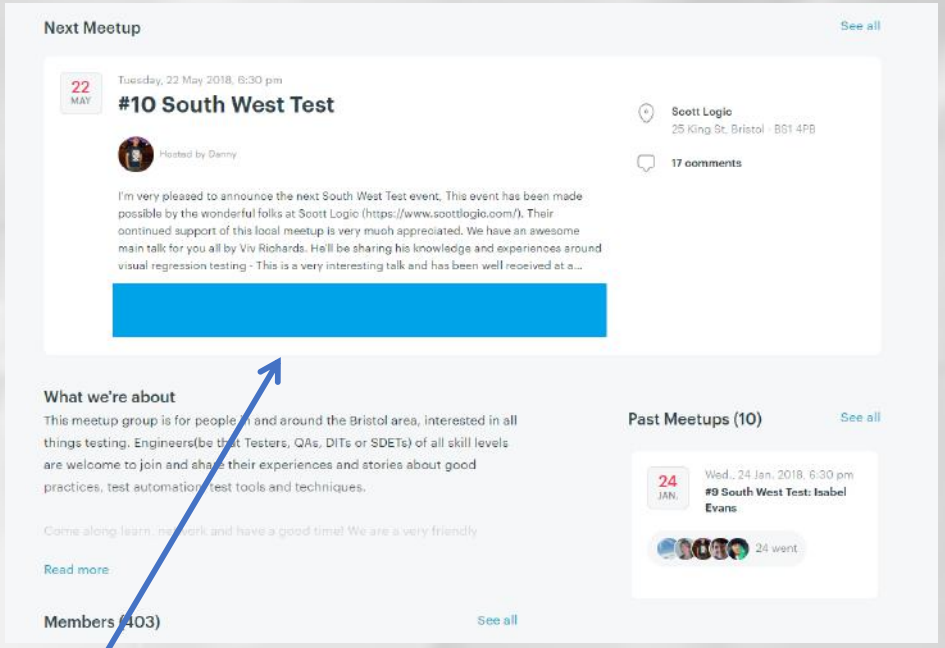
DYNAMIC CONTENT



DYNAMIC CONTENT



class="attendeesLink"

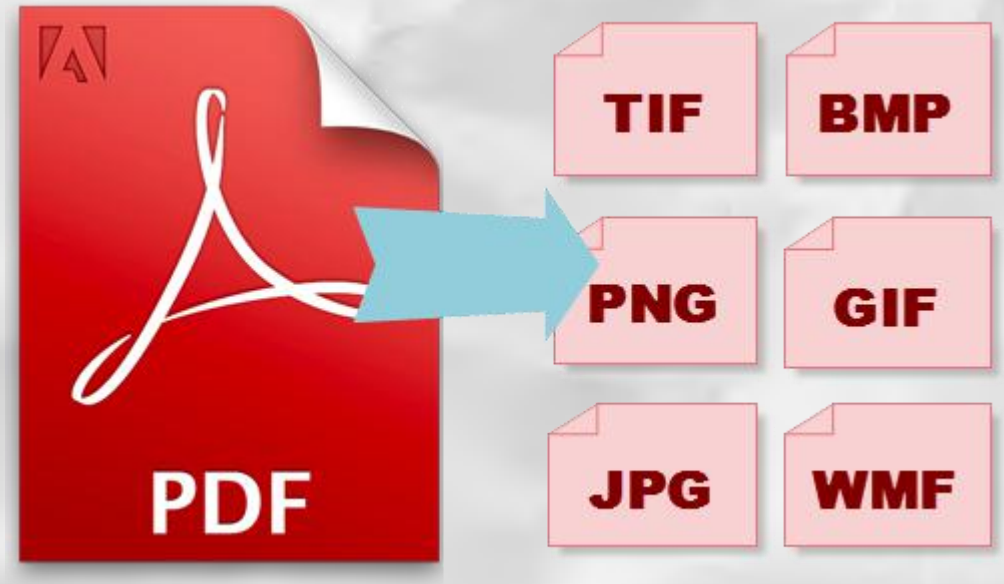



```
// Navigate to the web page
Driver.Navigate().GoToUrl("https://www.meetup.com/");

// cover dynamic attendees element
CoverDynamicElementBySelector("attendeesLink");
```

MOVING ELEMENTS

- CSS animations
- Videos

VISUALLY CHECKING DOCUMENTS



PDF DYNAMIC CONTENT

- A way to blanket content within documents

UPCOMING CHALLENGES

MORE COMPARISON OPTIONS

- Text only
- Layout only

ANOTHER TOOL TO ADD TO YOUR BELT

@11vlr

THANK YOU

Git repository: http://bit.ly/VisualTesting

NuGet.org Package:

```
PM> Install-Package AutomatedVisualTesting -Version 2.0.0
```

@11VLR

WWW.VIVRICHARDS.CO.UK