

Climbing Trees with UI Automation

Jasmine Greenaway
@paladique

Jasmine Greenaway
@paladique

Various Ways We Test



- Unit
- Integration
- Peers
- Manual

Reasons we manually test



Bug Fixes/Debugging

Did you truly fix the thing that broke?

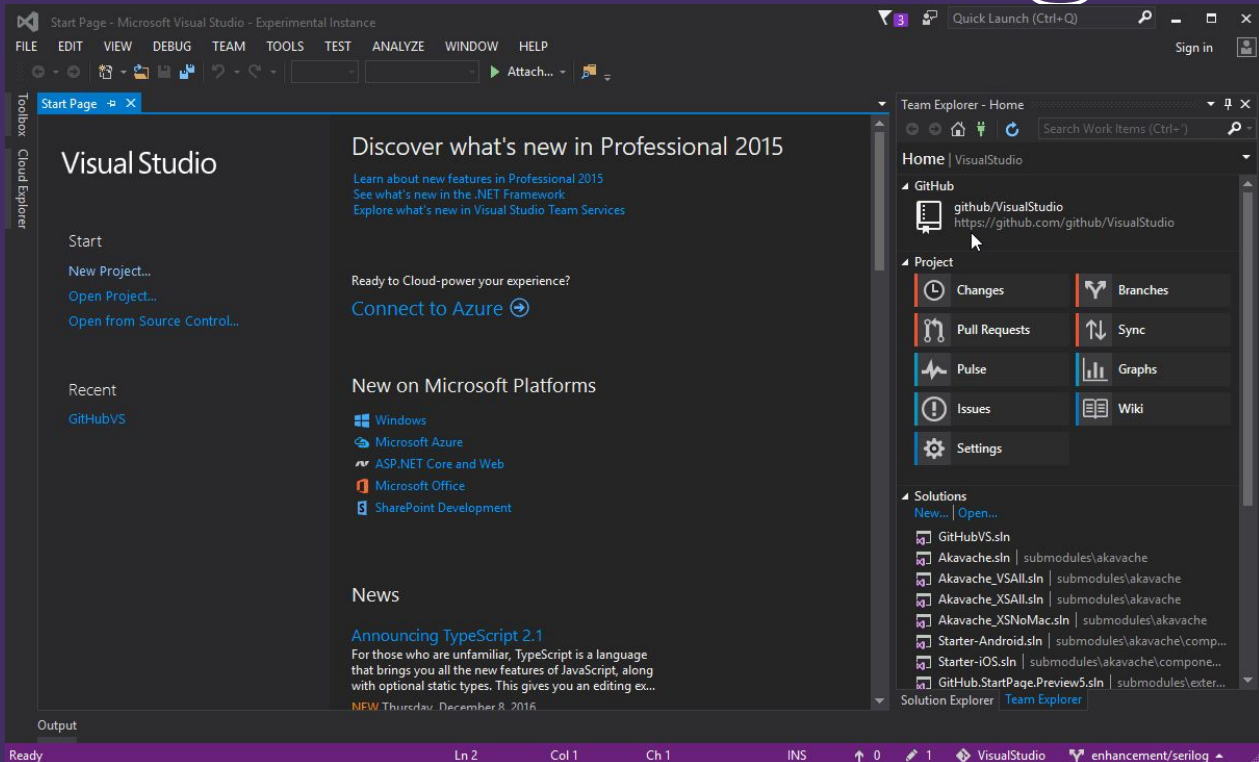
New Feature

Does this thing actually work?

Pre-release once over


Did I fix all the things that broke and actually add a feature that works?

Editor Tools' Bugs




Broken by Yours Truly

▲ Publish to GitHub

 This repository does not have a remote. Fill out the form to publish it to GitHub.

GitHub

 ouchmyfoothurts

ConsoleApplication4

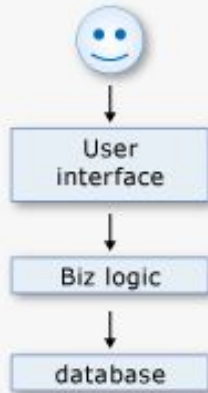
Description (Optional)

☐ Private Repository

Publish

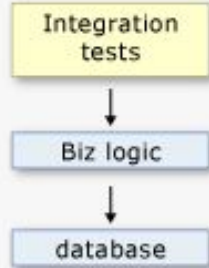
Coded UI – Automated UI Testing

F5 experience
- build and manually test whole application through the UI



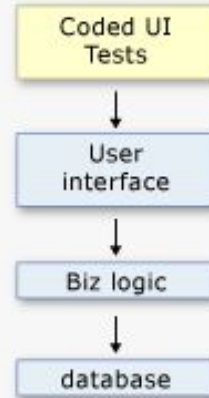
Manual

Typical non-CUIT test
- testing the whole application not through the UI (for example, unit tests)

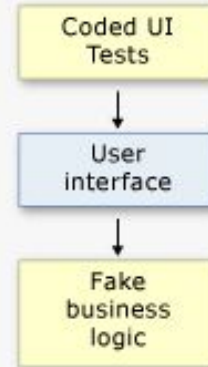


Coded

Typical use of CUITs
- testing the whole application through the UI automatically



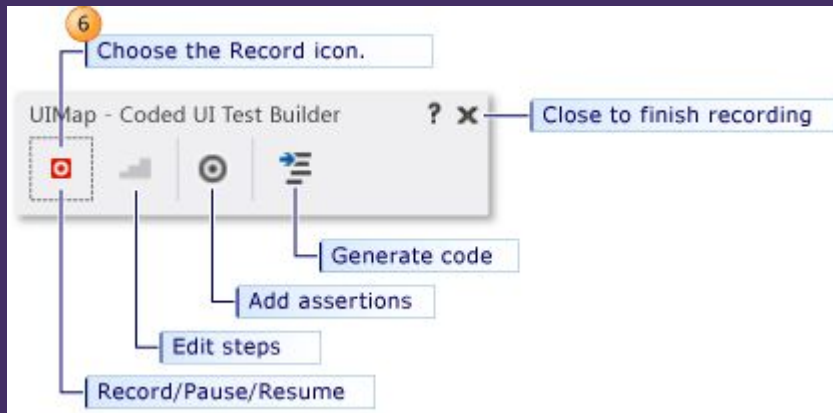
Tests that verify the user interface
- testing the UI in isolation



Coded UI – Automated UI Testing

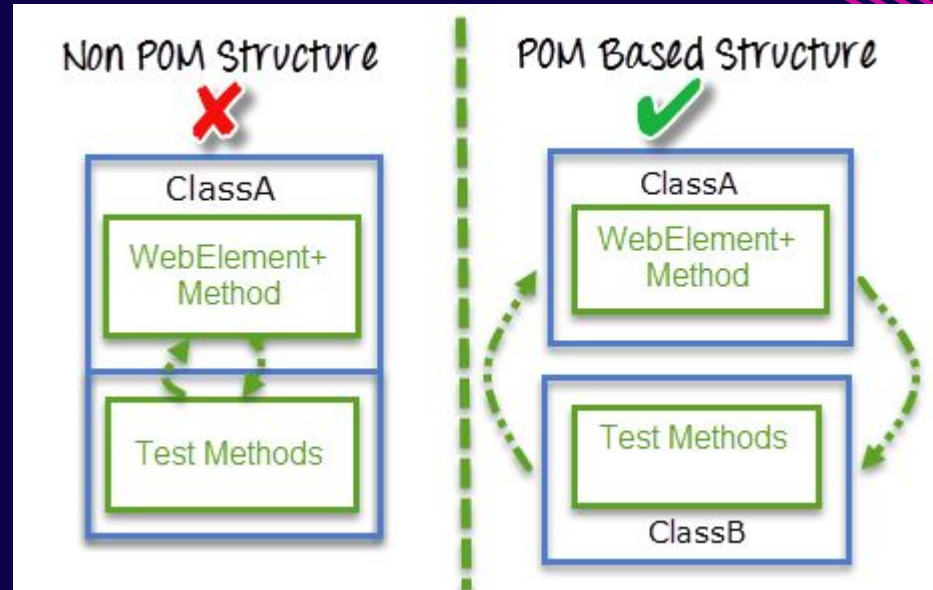


- Tests driven by user interface
- Support for: Win Forms, WPF, Web, Windows Phone and Store Apps
- Generate tests or code your own framework



Page Object Model

- Separated views accessible via an API
- Build tests by doing actions through API
- Actions alter state, not the tests
- Actions lead to assertions



Page Object Model Implemented:

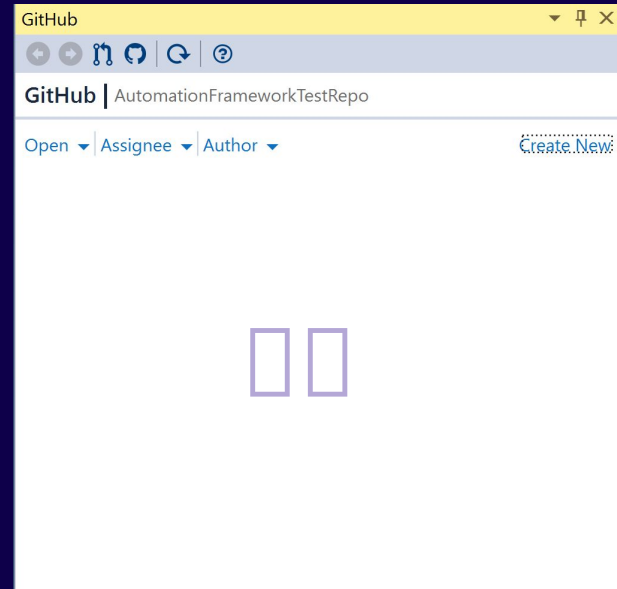
Confirm there's 0 Pull requests in a repo

- Open a repo with 0 Pull requests
 - Get to the Team Explorer Connect Page
 - Use the Local Repositories list to find the repo and open it
- View the Pull Requests Pane
 - Open the GitHub Pane
 - Show Pull Request List
 - Count the Pull Requests
- Confirm 0 pull requests

```
public void PullRequestCountMatchesDotCom()
{
    var mainWindow = vs;
    var repoName = "AutomationFrameworkTestRepo";
    mainWindow.ShowTeamExplorer()
        .ShowConnectPage().LocalRepositories.OpenRepository(repoName);

    var pullRequestCount = mainWindow.ShowGitHubPane()
        .ShowPullRequestList()
        .GetPullRequestItems().Count;

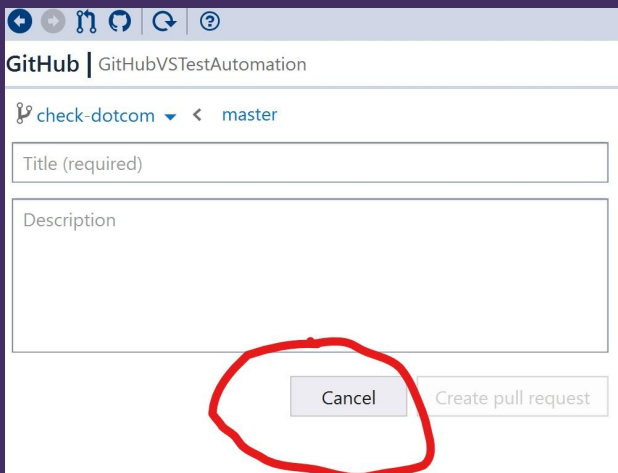
    Assert.AreEqual(pullRequestCount, 0);
}
```



Tell Coded UI Everything!



```
cancelPullRequestButton = new WpfButton(this);  
cancelPullRequestButton.AddSearchProperties(  
    SetSearchProperty(PropertyNames.Name, "Cancel"),  
    SetSearchProperty(PropertyNames.ControlType, "Button"),  
    SetSearchProperty(PropertyNames.AutomationId, AutomationIDs.PullRequestCreationCancelButton));
```

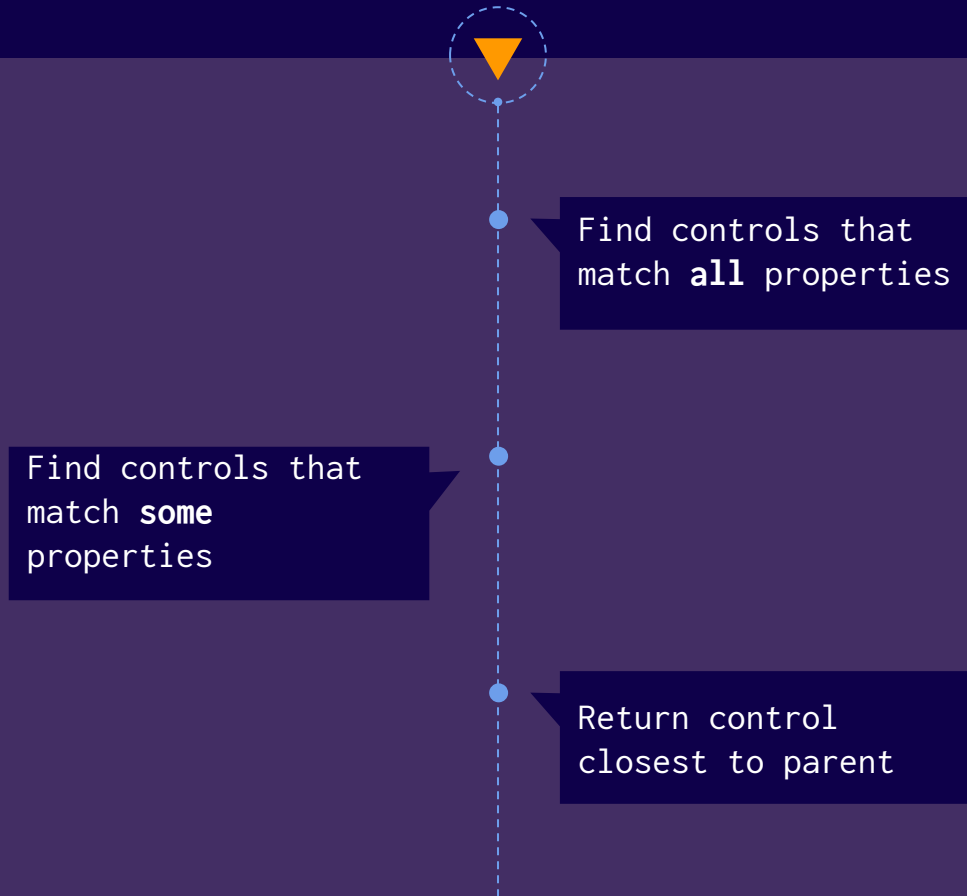


Make sure control is there before you use
it!



```
public static void Ensure(this WpfControl control, bool includeEnabled = true)
{
    if (!(control.Exists && (includeEnabled ? control.Enabled : true)))
    {
        throw new AutomationExceptions.ControlNotVisible(control.SearchProperties);
    }
}
```

Coded UI's Search for the ❖Perfect❖ Control



From manifest...

“Setting the target and source branch as the same branches disables the "Create Pull Request" button and displays an error message at the top of pane.”

To test case!

```
public void CannotSetSameSourceAndTarget()
{
    var mainWindow = vs;
    var github = mainWindow.ShowGitHubPane();
    var pullRequestCreationView = github.ShowPullRequestList().ShowCreationView();
    var sourceBranch = pullRequestCreationView.SourceBranchButton;
    pullRequestCreationView.ChangeTargetBranch(sourceBranch.Name);
    var createPRButton = pullRequestCreationView.CreatePullRequestButton;

    pullRequestCreationView.PullRequestTitleTextBox.SetFocus();
    Keyboard.SendKeys("Title of the Pull Request");

    Assert.IsTrue(!createPRButton.Enabled);
}
```



What else?



App usage tracking

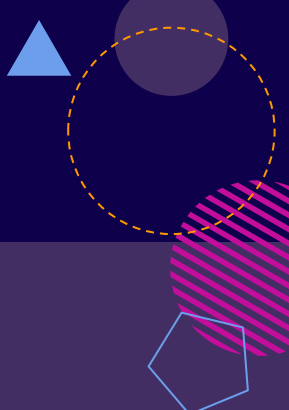
Run some tests and count how many steps it took

Bug Sniffing:

Run some additional targeted tests and take some screenshots to simulate user input.

Documentation

Do some actions and have Coded UI take screenshots for you.



Our Wishlist



Chat command

Automate all the tests, or a set of tests through chat commands.

Integrated in Release Process

Supplement manual testing during releases.

Clarity for QA

Engineers, specifically quality engineers can easily add tests because it makes sense.

Open source

Sharing is caring.

Steady coverage at 90%

Keep on top of established features and constantly test for regressions.

Dedicated Virtual Machine

Machine just for running tests, which can take some time.



Thanks!

Any questions?

You can find me at @paladique (Twitter/GitHub)

Talk Transcript github.com/paladique/talks

GitHub for Visual Studio github.com/github/visualstudio

GitHub for Unity <https://github.com/github-beta/unity-preview>

OR unity.github.com

Presentation template by [SlidesCarnival](https://slidescarnival.com)