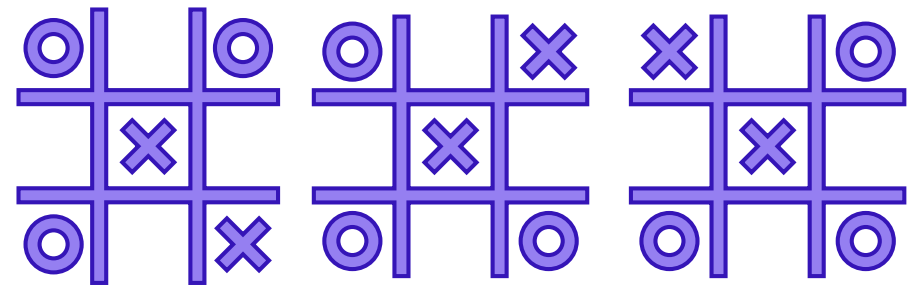


Tic Tac Toe *Testing*

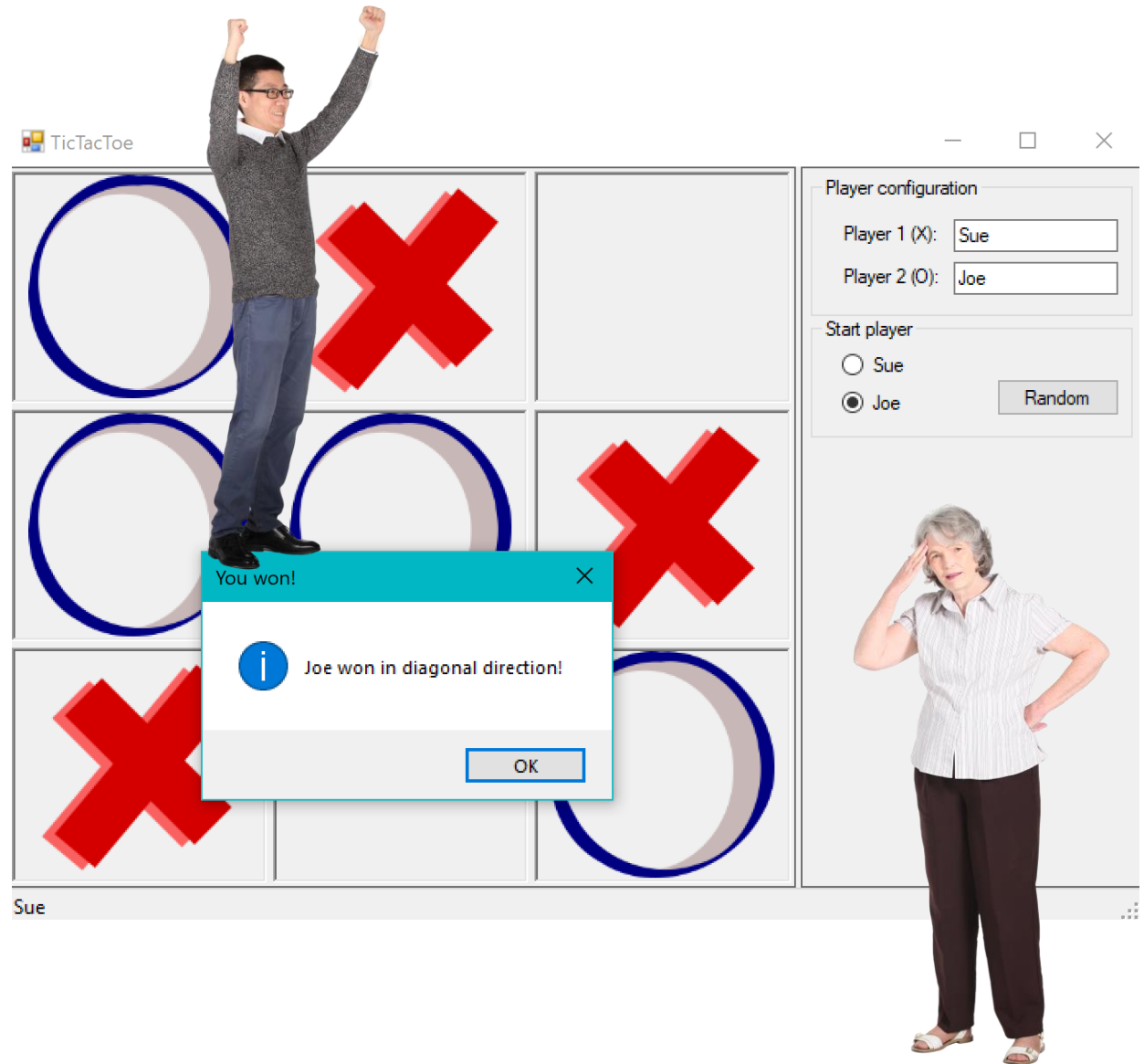


MILESTONE 2

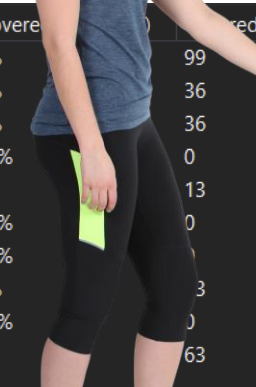
PRESENTATION BY:

SAM, TEJA, AND DOUG

Our Project: Tic Tac Toe




Unit Tests and Coverage



Hierarchy	Not Covered (Blocks)	Not Covered (% Blocks)	Covered (Blocks)	Covered (% Blocks)
▲ Dale_EDITH 2021-05-18 06_49...	921	90.29%	99	8.71%
▲ tictactoe.exe	921	96.24%	36	3.76%
▲ {} TicTacToe	916	96.22%	36	3.76%
▷ Game	151	100.00%	0	0.00%
▷ Game.GameResult	0	0.00%	13	100.00%
▷ GameForm	746	100.00%	0	0.00%
▷ GameForm.<>c	2	100.00%	0	0.00%
▷ Program	17	42.50%	3	57.50%
▷ {} TicTacToe.Properties	5	100.00%	0	0.00%
▷ tictactoe.tests.dll	0	0.00%	63	100.00%

3.76% - Prior to Refactoring



Hierarchy	Not Covered (Blocks)	Not Covered (% Blocks)	Covered (Blocks)	Covered (% Blocks)
▲ Dale_EDITH 2021-05-19 10_12...	887	80.42%	216	19.58%
▲ tictactoe.exe	884	91.51%	82	8.49%
▲ {} TicTacToe	879	91.47%	82	8.53%
▷ ExtractedFunctions	2	4.17%	46	95.83%
▷ Game	112	100.00%	0	0.00%
▷ Game.GameResult	0	0.00%	13	100.00%
▷ GameForm	746	100.00%	0	0.00%
▷ GameForm.<>c	2	100.00%	0	0.00%
▷ Program	17	42.50%	23	135.29%
▷ {} TicTacToe.Properties	5	100.00%	0	0.00%
▷ tictactoe.tests.dll	3	2.19%	60	100.00%

8.49% - After Refactoring

Refactoring for Better Coverage!

19 references | herstadd, 12 days ago | 1 author, 6 changes

```
public static class ExtractedFunctions
```

```
{
```

```
    public static int something = 0;
```

4 references | 🟢 2/2 passing | herstadd, 12 days ago | 1 author, 1 change

```
    public static Game.GameResult CheckWinner(Player lastDia2Elem, Game.GameResult result, int winner)
```

```
    {
```

```
        if (lastDia2Elem != Player.PlayerNull)
```

```
        {
```

```
            Debug.WriteLine("SUCCESS: found result in diagonal " + winner);
```

```
            result.Winner = lastDia2Elem;
```

```
            result.Diagonal = winner;
```

```
        }
```

```
        return result;
```

```
    }
```

New static class with extracted functionality

Automated Regression Testing

```
Windows PowerShell
PS C:\Users\PRATHYUSHA> .\Desktop\newonne\look.ps1

          DOUGs Team
***Script for Regression testing***

Select any one below options:

1. BUILD
2. BUILD & RUN
3. MOVE LOGS TO CLOUD
4. SEND MESSAGE TO TEAMS CHANNEL
5. Exit

Enter your option:
```

- Easy to run

```
=====BUILD SUCCEEDED=====

Passed CanSelectPictureHelper_PlayerNull_Should_Be_True [11 ms]
Passed CanSelectPictureHelper_PlayerO_Should_Be_False [1 ms]
Passed CheckWinner_Diagonal1_Should_Pass [5 ms]
Passed CheckWinner_Diagonal2_Should_Pass [< 1 ms]
Passed SwitchPlayer_SelectPlayerInvalid_Should_Fail [38 ms]
Passed SwitchPlayer_SelectPlayerO_Should_Pass [5 ms]
Passed SwitchPlayer_SelectPlayerX_Should_Pass [< 1 ms]
Passed TestColumnsForWinner_PlayerNull_Should_Pass [11 ms]
Passed TestColumnsForWinner_PlayerO_Should_Pass [< 1 ms]
Passed TestColumnsForWinner_PlayerX_Should_Fail [5 ms]
Passed TestRowsForWinner_PlayerNull_Should_Pass [< 1 ms]
Passed TestRowsForWinner_PlayerO_Should_Pass [< 1 ms]
Passed TestRowsForWinner_PlayerX_Should_Fail [< 1 ms]
Passed GameResults_Valid_ColumnGetSet [1 ms]
Passed GameResults_Valid_DiagonalGetSet [< 1 ms]
Passed GameResults_Valid_LineGetSet [< 1 ms]
Passed GameResults_Valid_WinnerGetSet [< 1 ms]
Passed FirstLetterToUpper_Valid_ParamNull_ReturnNull [< 1 ms]
Passed FirstLetterToUpper_Valid_StrEmpty_ReturnEmpty [< 1 ms]
Passed FirstLetterToUpper_Valid_StrLenGreaterThan1_ReturnCorrect [< 1 ms]
Passed FirstLetterToUpper_Valid_StrLenIsOne_ReturnCorrect [< 1 ms]
Passed Random_Valid_SetsCorrectlyAndGetsCorrectly [< 1 ms]
Passed ReplaceLastOccurrence_Source_find_match_ReturnReplaced_string [< 1 ms]
Passed ReplaceLastOccurrence_Source_find_notmatch_Return_minusone [< 1 ms]
Failed FirstLetterToUpper_Valid_ParamNull_ReturnNull [8 ms]
Failed FirstLetterToUpper_Valid_StrEmpty_ReturnEmpty [24 ms]
Failed FirstLetterToUpper_Valid_StrLenGreaterThan1_ReturnCorrect [1 ms]
Failed FirstLetterToUpper_Valid_StrLenIsOne_ReturnCorrect [1 ms]
Failed ReplaceLastOccurrence_Source_find_match_ReturnReplaced_string [7 ms]
Failed ReplaceLastOccurrence_Source_find_notmatch_Return_minusone [1 ms]
dotnet : Test Run Failed.
+ CategoryInfo          : NotSpecified: (Test Run Failed.:String) [], RemoteException
dotnet : Test Run Failed.
+ CategoryInfo          : NotSpecified: (Test Run Failed.:String) [], RemoteException
Passed: 24
Failed: 6
```

- Clear useful output

Build Application

To build C# core applications we used msbuild.exe to build our project
This extension requires “buildutils” modules

So , we imported “buildutils” modules in PowerShell

```
#dependencies
if(!(Get-InstalledModule | where-Object {$_.Name -like "buildut*"})){
    Install-Module Buildutils -Scope CurrentUser -Force}
$msbuildLocation = Get-LatestMsbuildLocation
set-alias msb $msbuildLocation
```

Run Test Code via a script.

As our test cases are in NUnit framework but the application is completely standalone which is purely on .net core , we do not have leverage to import NUnit modules into PowerShell to run test code although our unit tests were written in NUnit framework this is because NUnit Console is compiled using the full .NET Framework and does not currently support .NET Core.

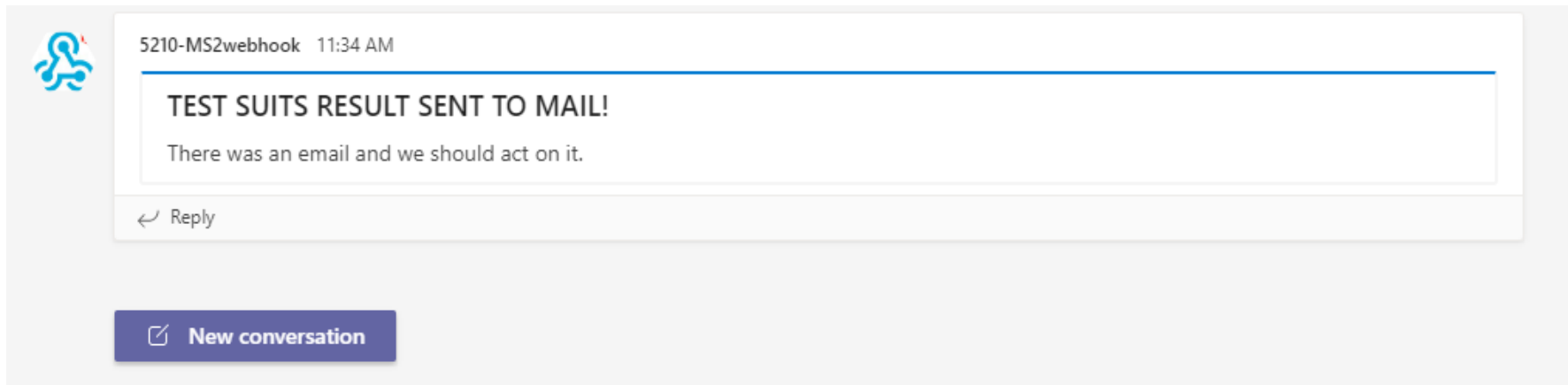
```
Test Run Failed.  
Total tests: 30  
    Passed: 24  
    Failed: 6  
Total time: 1.2810 Seconds  
Transcript stopped, output file is C:\Users\PRATHYUSHA\source\repos\RAVITEJAPOOSALA.log  
  
cmdlet buildTestSuite.ps1 at command pipeline position 1  
Supply values for the following parameters:  
Address:
```

So, we used dotnet test drivers which can run .net core tests from command line and used Outlook class to create object and send our test suit over email.

Mail service and Teams Alerts

Sending mails to outlook and other domains(gmail,yahoo) straight forward. We used Outlook class and called send(),method to send our mail to teams

For Teams alerts we imported teams modules into powershell engine -> connected to SU account -> generated webhook API and followup steps



```

10 references | 0 changes | 0 authors, 0 changes
internal static string FirstLetterToUpper(string str)
{
    if (str == null)
        return null;

    if (str.Length > 1)
        return char.ToUpper(str[0]) + str.Substring(1);

    return str.ToUpper();
}

```

- Before – working code

```

internal static string FirstLetterToUpper(string str)
{
    if (str != null)
        return null;

    if (str.Length > 1)
        return char.ToUpper(str[0]) + str.Substring(1);

    return str.ToUpper();
}

```

- After – Erroneous code

Mutant Testing

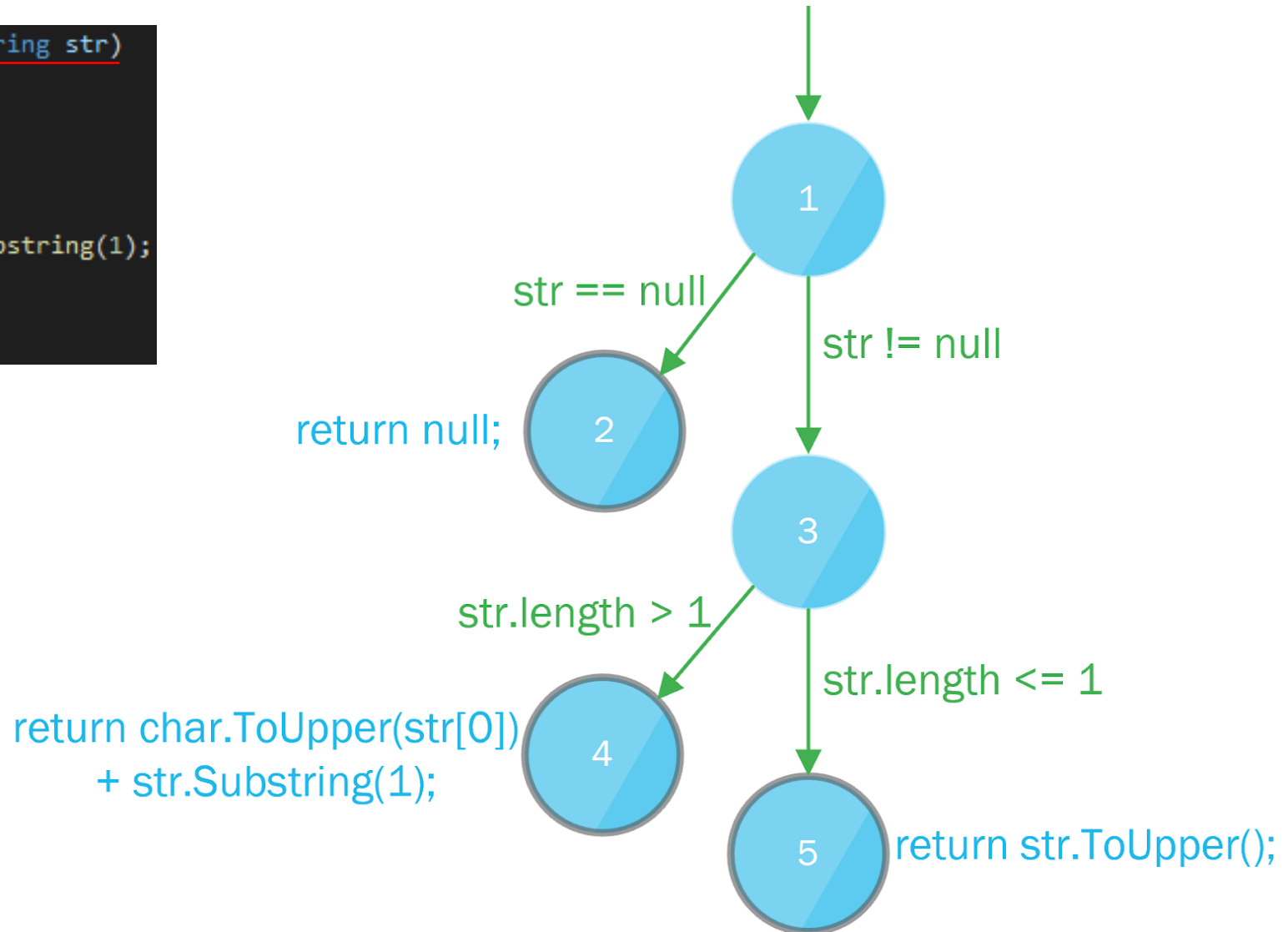
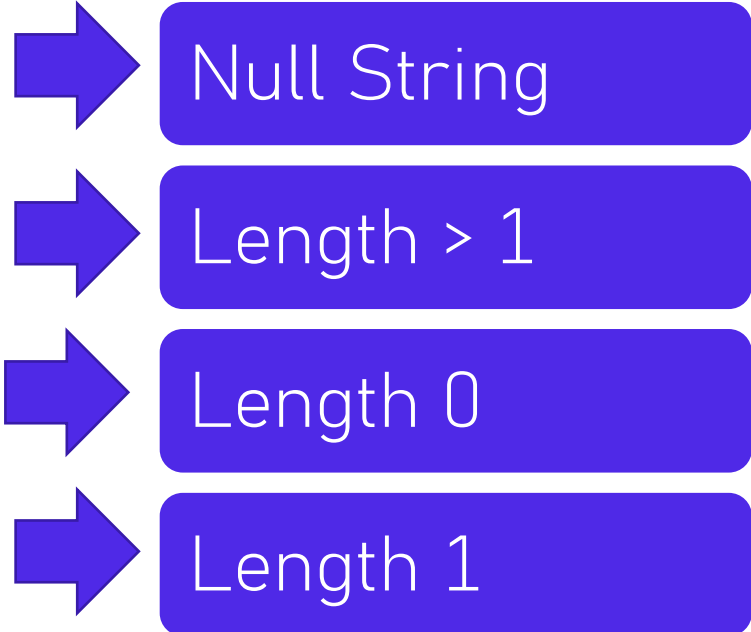
Test	Duration	Traits	Error Message
✖ TicTacToeTests (24)	106 ms		
✖ TicTacToeTests (24)	106 ms		
✔ ExtractedFunctionTests	78 ms		
✔ GameTests (4)	1 ms		
✖ ProgramTests (7)	27 ms		
✖ FirstLetterToUpper_...	14 ms		System.NullReferenceException : Object reference...
✖ FirstLetterToUpper_...	13 ms		Expected: null But was: <string.Empty>
✖ FirstLetterToUpper_...	< 1 ms		Expected: null But was: "Lower"
✖ FirstLetterToUpper_...	< 1 ms		Expected: null But was: "L"
✔ Random_Valid_Sets...	< 1 ms		
✔ ReplaceLastOccurre...	< 1 ms		
✔ ReplaceLastOccurre...	< 1 ms		

Equivalence Partitions & Code Flow Graph

```
internal static string FirstLetterToUpper(string str)
{
    if (str == null)
        return null;

    if (str.Length > 1)
        return char.ToUpper(str[0]) + str.Substring(1);

    return str.ToUpper();
}
```



Truth Table

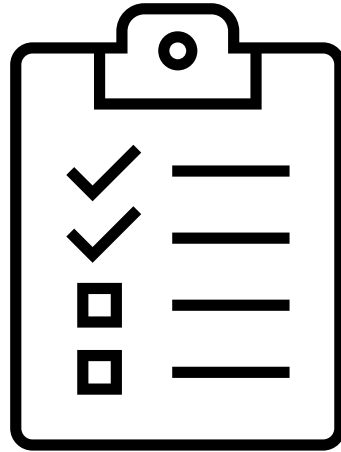
```
public static bool GameOver(bool XWin, bool OWin, bool boardFilled)
{
    if(!boardFilled && !XWin && !OWin) {
        return false;
    }
    return true;
}
```

P =

**!boardFilled &&
!XWin &&
!OWin**

boardFill	Xwin	Owin	P	P(boardFill)	P(Xwin)	P(Owin)
T	T	T	T	F	F	F
T	T	F	T	F	F	F
T	F	T	T	F	F	F
T	F	F	T	T	F	F
F	T	T	T	F	F	F
F	T	F	T	F	T	F
F	F	T	T	F	F	T
F	F	F	F	T	T	T

Test Plan



- Intended audience: future testers on the internet
- Summary of what we did, didn't do
- Methodology: Waterfall instead of Agile
- Future Testing Suggestions:
GUI Testing, Bug tracking software, Manual exploratory testing



Lessons Learned

- Finding testable opensource projects is hard!
- Refactoring can help
- GUI testing can be a challenge
- PowerShell can help with automated regression testing



Questions?

