Wimygit Test Plan

|  |  |
| --- | --- |
| Application Name | wimygit |
| Application ID | APP\_3166 |
| Application Version | NA |
| Application Install Location | App Location on Appcompatsrv Server(\\appcompatsrv\Appcompat\DotNetCore\Apps\WimyGit) |
| Source Install Location | https://github.com/zelon/wimygit |
| Application Priority | P5 |
| Technology | WPF |
| .NET Version | .NET Core 3.0 |
| Client Supported | Yes |
| Deployment | XCopy |
| Design Times usage (hours) | NA |

# History Log

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Date** | **Comments** | **Author** |
| 1 | 04/28/2020 | Create the test plan | Xiaoxiao |
| 2 | 04/29/2020 | Update the test plan | Xiaoxiao |
|  |  |  |  |

# Configurations

Following configurations are tested in the manual phase:

|  |  |  |  |
| --- | --- | --- | --- |
| Config # | Platforms + NetFx | Target To | Comments |
| 1 | Windows 10 X64 + .NET Core 3.0 | Default version | Design test plan |
|  |  |  |  |

Table of Contents

[History Log 1](#_Toc19101169)

[Configurations 1](#_Toc19101170)

[Summary 2](#_Toc19101171)

[Pre-requisite 2](#_Toc19101172)

[Brief Introduction 2](#_Toc19101173)

[Installation or Deployment 2](#_Toc19101174)

[Uninstallation 2](#_Toc19101175)

[Scenarios 2](#_Toc19101176)

[Scenario 1: Test New PDF 2](#_Toc19101177)

[Issues 6](#_Toc19101178)

# Summary

Following table shows scenarios tested for the app and their pass/fail status.

|  |  |  |  |
| --- | --- | --- | --- |
| Items | Test Case Title | Technology Description | Config #1 |
| Install | -- | -- | Pass |
| Uninstall | -- | -- | Pass |
| Look and feel | -- | -- | Pass |
| Performance | -- | -- | Pass |
| APP\_3166\_001 | Test function |  | Pass |
| APP\_3166\_002 | Test function |  | Pass |
| APP\_3166\_003 | Test function |  | Pass |
| APP\_3166\_004 | Test function |  | Pass |
| APP\_3166\_005 | Test function |  | Fail |
| APP\_3166\_006 | Test function |  | Pass |
| APP\_3166\_007 | Test function |  | Pass |
| APP\_3166\_008 | Test function |  | Pass |
| APP\_3166\_009 | Test function |  | Pass |
| Issues | NA |  | NA |
|  |  |  |  |

# Pre-requisite s

|  |  |  |
| --- | --- | --- |
| **No.** | **Name** | **Comment** |
| 1 | Dotnet sdk | Version 3.0 |
|  |  |  |
|  |  |  |

# Brief Introduction

Wimgit app offer a GUI which can pull/push/fetch code by click buttons, instead of writing command.

# Installation or Deployment

Copy all files from [\\appcompatsrv\Appcompat\DotNetCore\Apps\WimyGit](file:///\\appcompatsrv\Appcompat\DotNetCore\Apps\WimyGit)

to %APP\_CLIENT\_ROOT%\application\WimyGit folder.

# Uninstallation

NA

# Scenarios

## Scenario 1: Test Refresh

**Case ID:** APP\_3166\_001

**Clean Environment:**

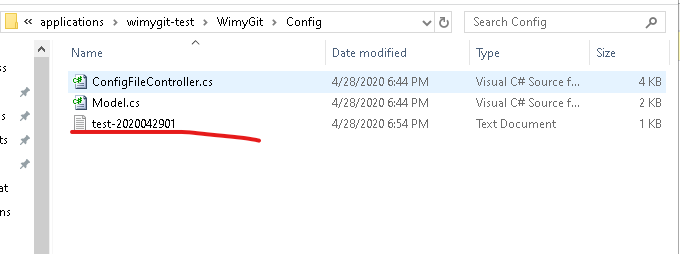
Make sure it is a clean environment and it is clean now

**Test Steps:**

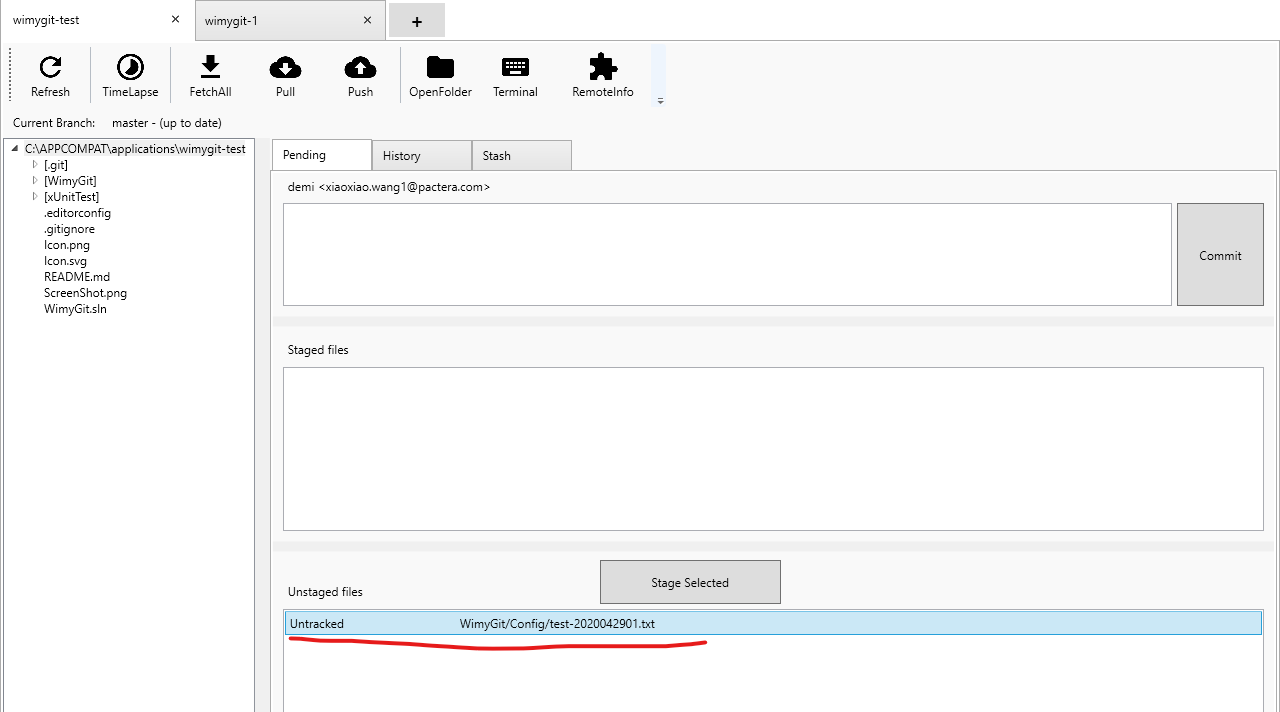
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe"
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



1. Create a new file in “%APP\_CLIENT\_ROOT%\wimygit-test\WimyGit\Config” folder.



1. Click “Refresh” button,and verify the new created file will show in unstaged files



## Scenario 2: Test Push

**Case ID:** APP\_3166\_002

**Clean Environment:**

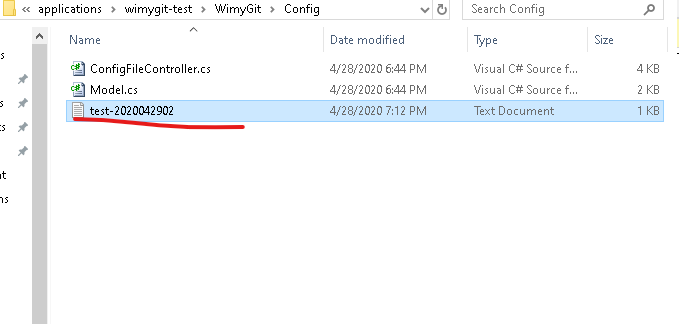
Delete “test-2020042901”file from “%APP\_CLIENT\_ROOT%\wimygit-test\WimyGit\Config” folder

**Test Steps:**

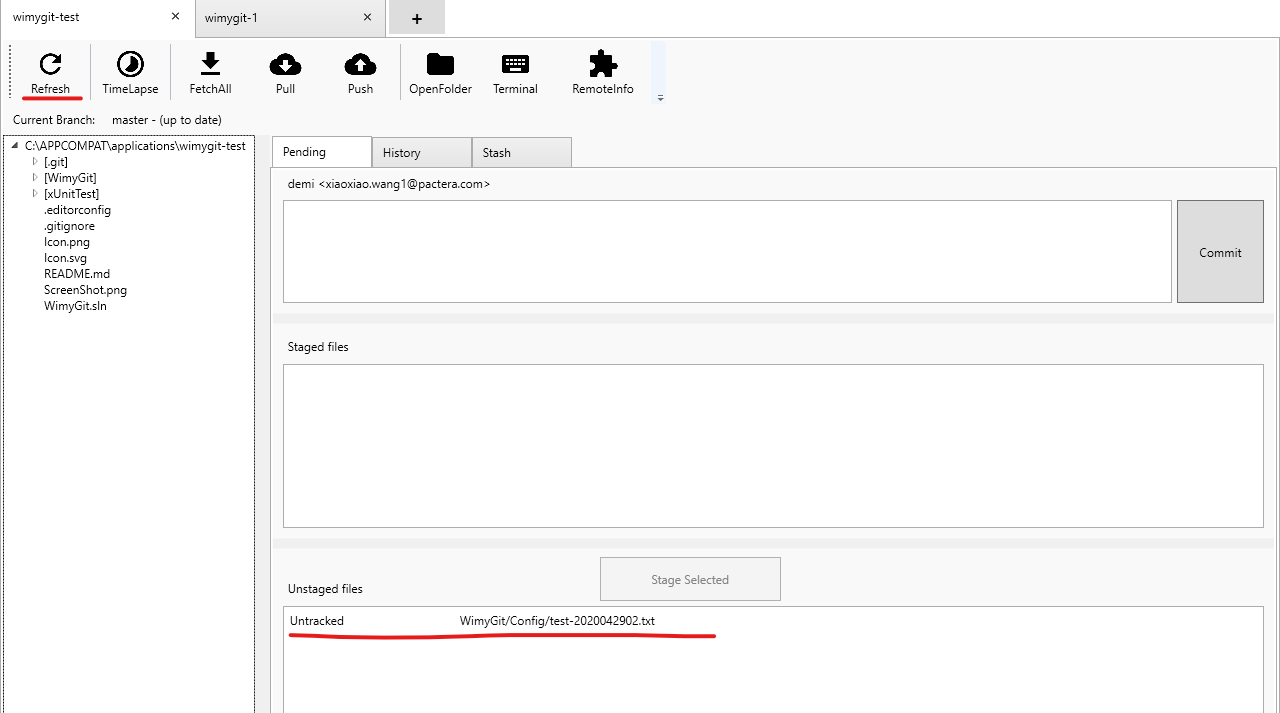
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



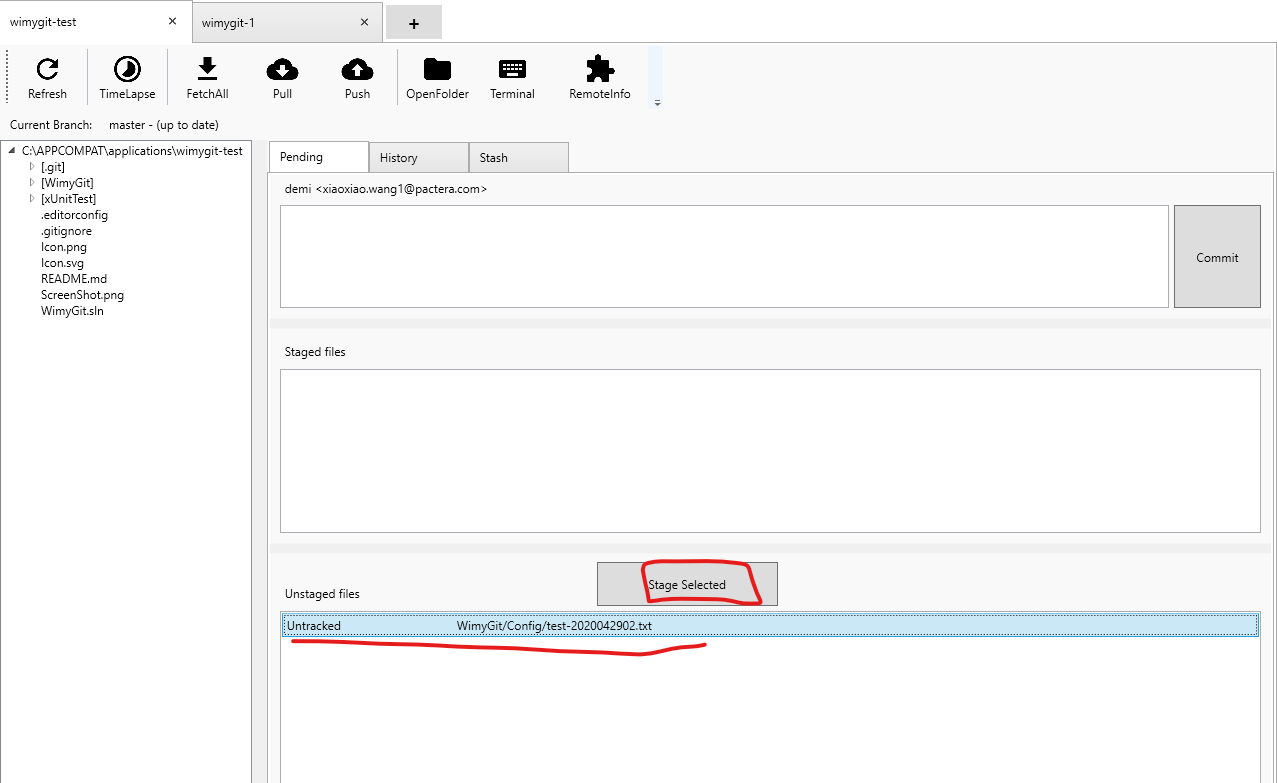
1. Create a new file in “%APP\_CLIENT\_ROOT%\wimygit-test\WimyGit\Config” folder.



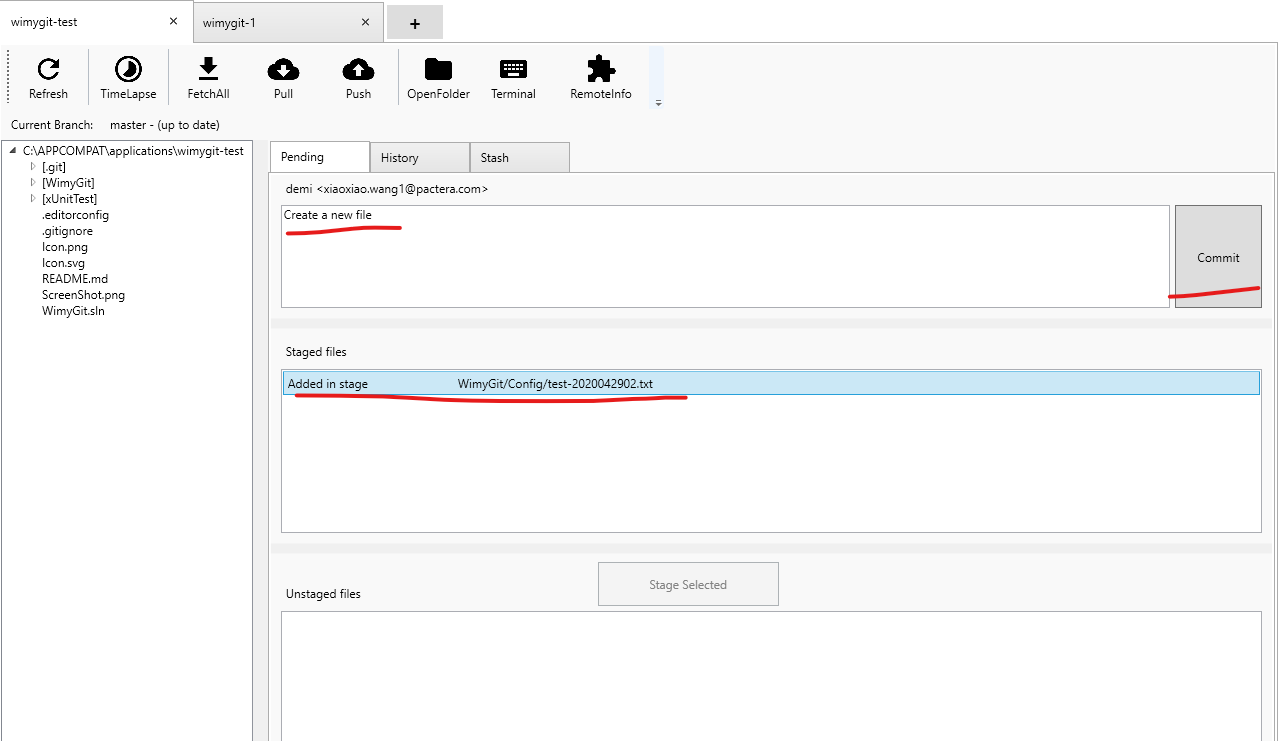
1. Click “refresh” buttton, the new file will show in the unstaged files area.



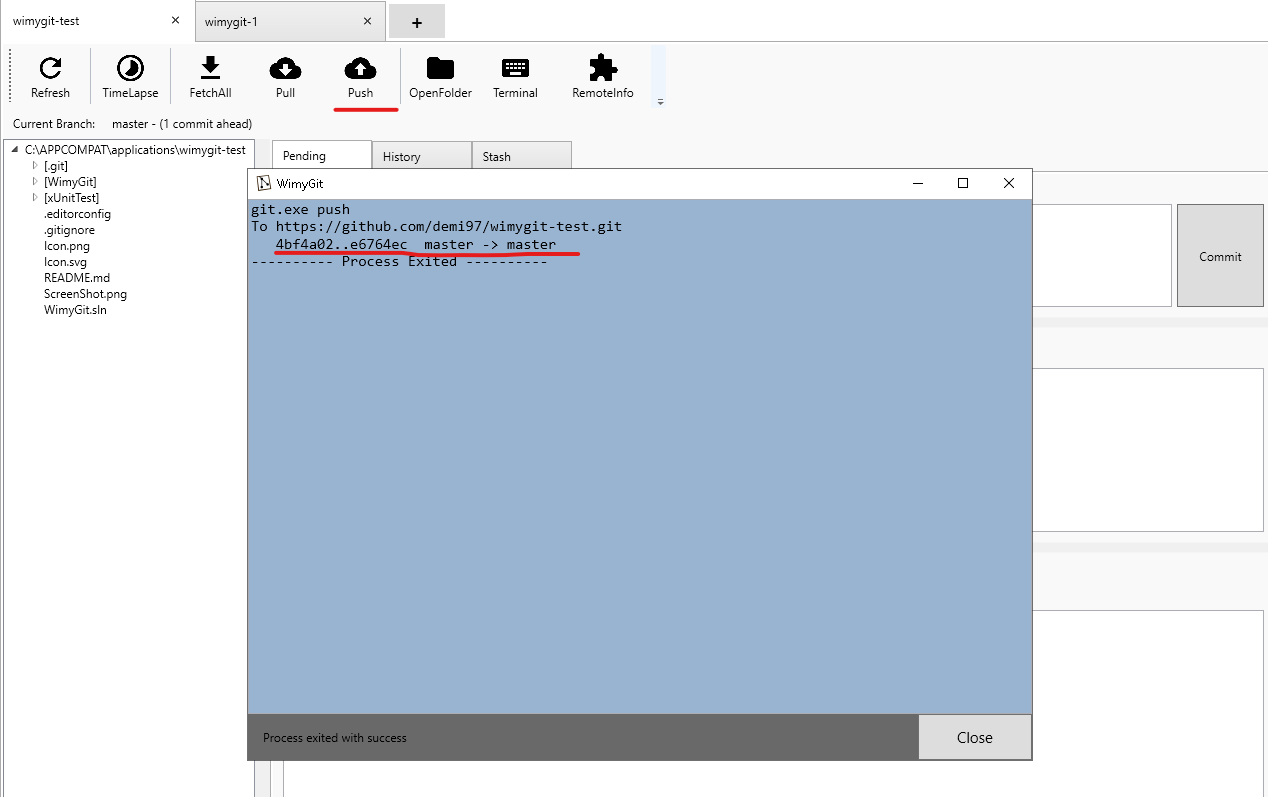
1. Select the file in unstaged files and click “Stage Selected” button to make it into Staged files area.



1. Write comment message and click “commit” button to sumit the modified content to local repo.



1. Click “push” button and verify it has been commited to remote server successfully.



## Scenario 3: Test Pull

**Case ID: APP\_3166\_003**

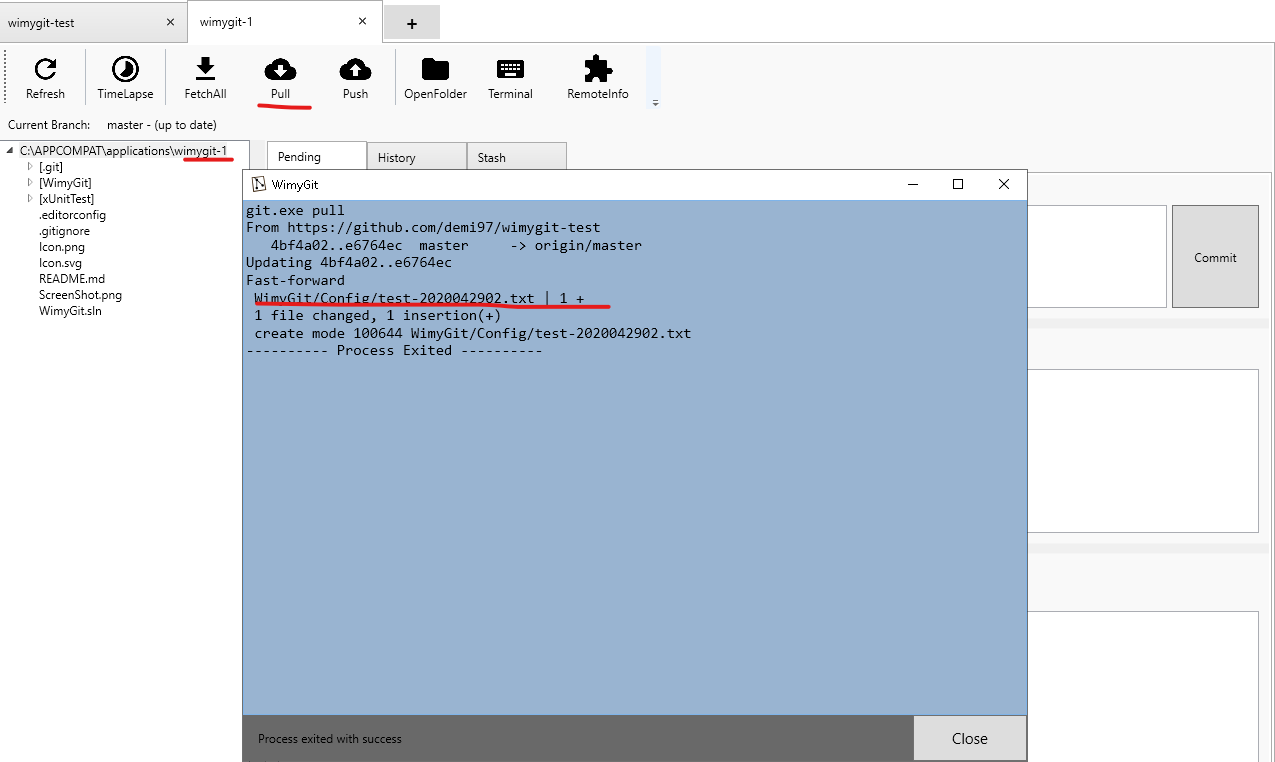
**Clean Environment:NA**

**Test Steps:**

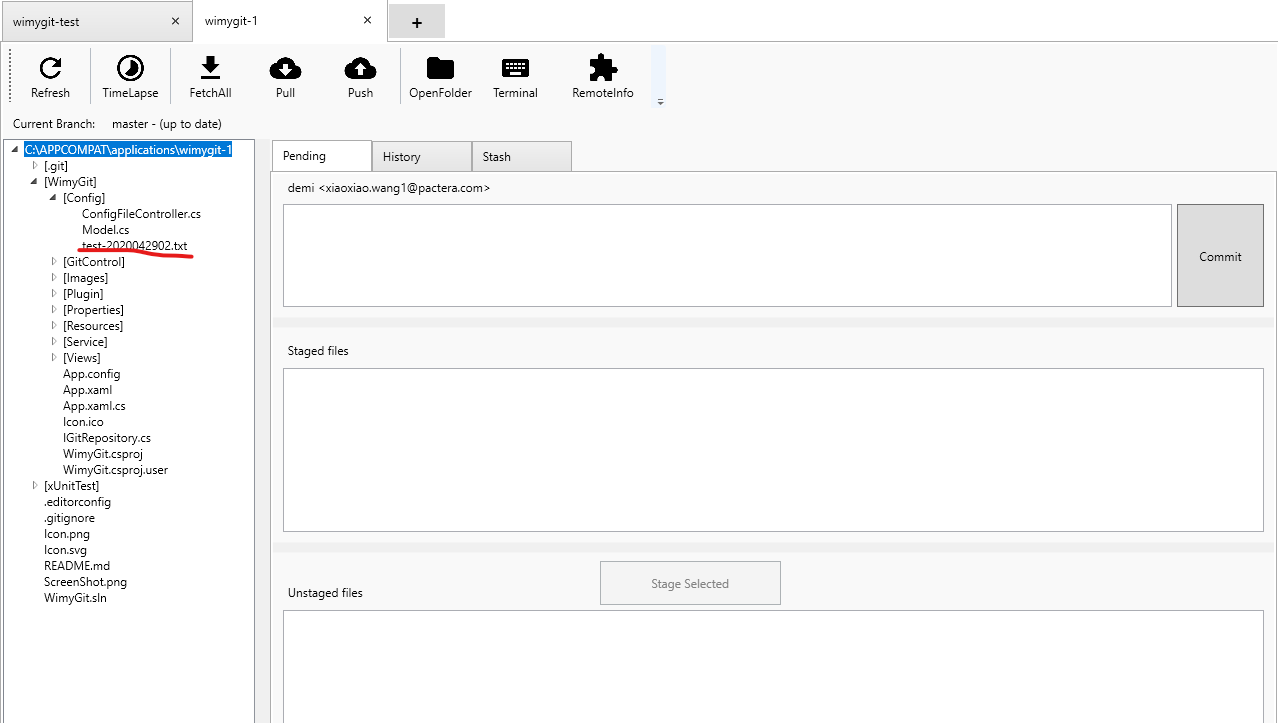
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-1”into here



1. Click “Pull” button to pull new content from remote server to local workspace



1. Verify the new content has been pulled to local workspace successfully.



## Scenario 4: Test FetchAll

**Case ID: APP\_3166\_004**

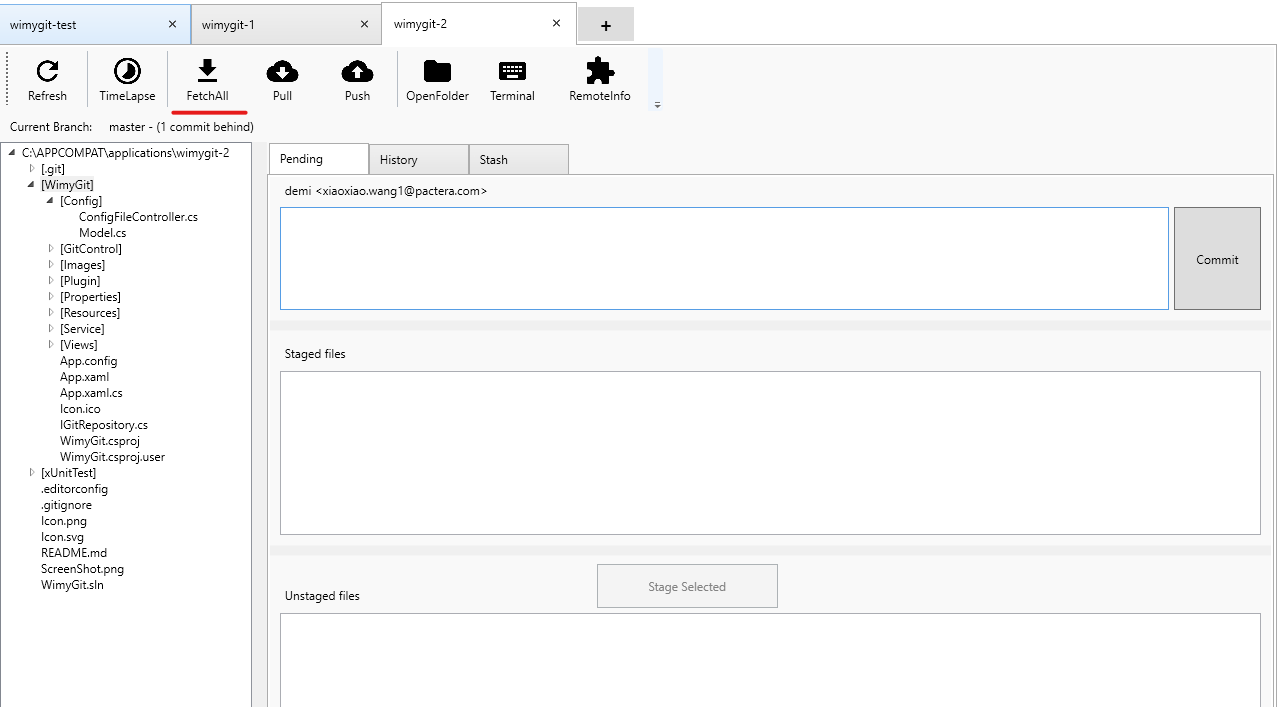
**Clean Environment:**

**Test Steps:**

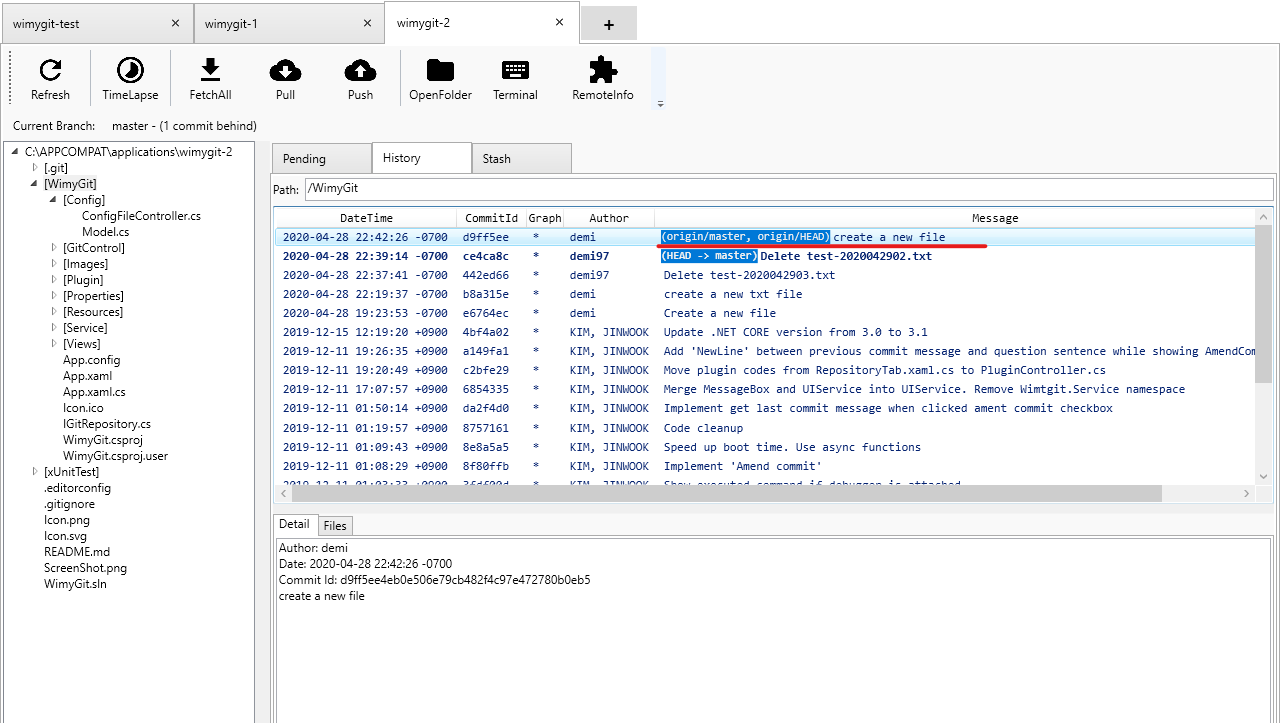
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-2”into here



1. Click “FetchAll” button to update new content from remote server to local repo



1. Verify the new content has been updated to local repo successfully.



## Scenario 5: Test TimeLapse

**Case ID: APP\_3166\_005**

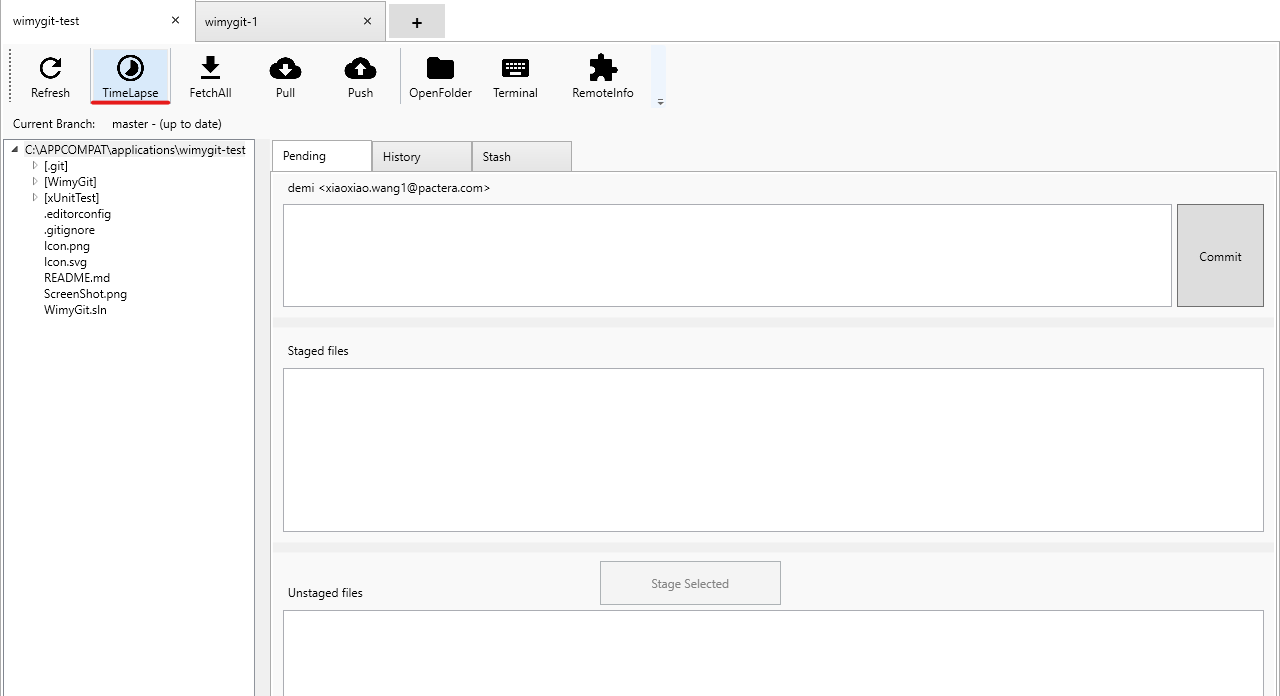
**Clean Environment:NA**

**Test Steps:**

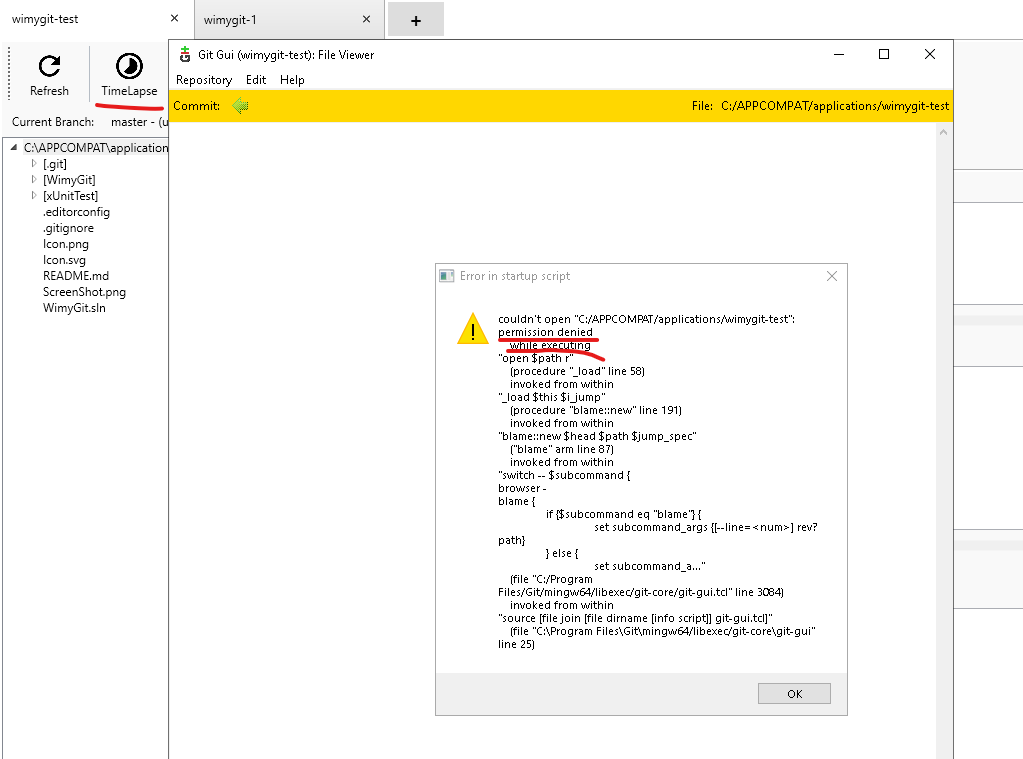
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



1. click “TimeLapse” button



1. Verify the app doesn’t support this because of the pemission.



## Scenario 6: Test OpenFolder

**Case ID: APP\_3166\_006**

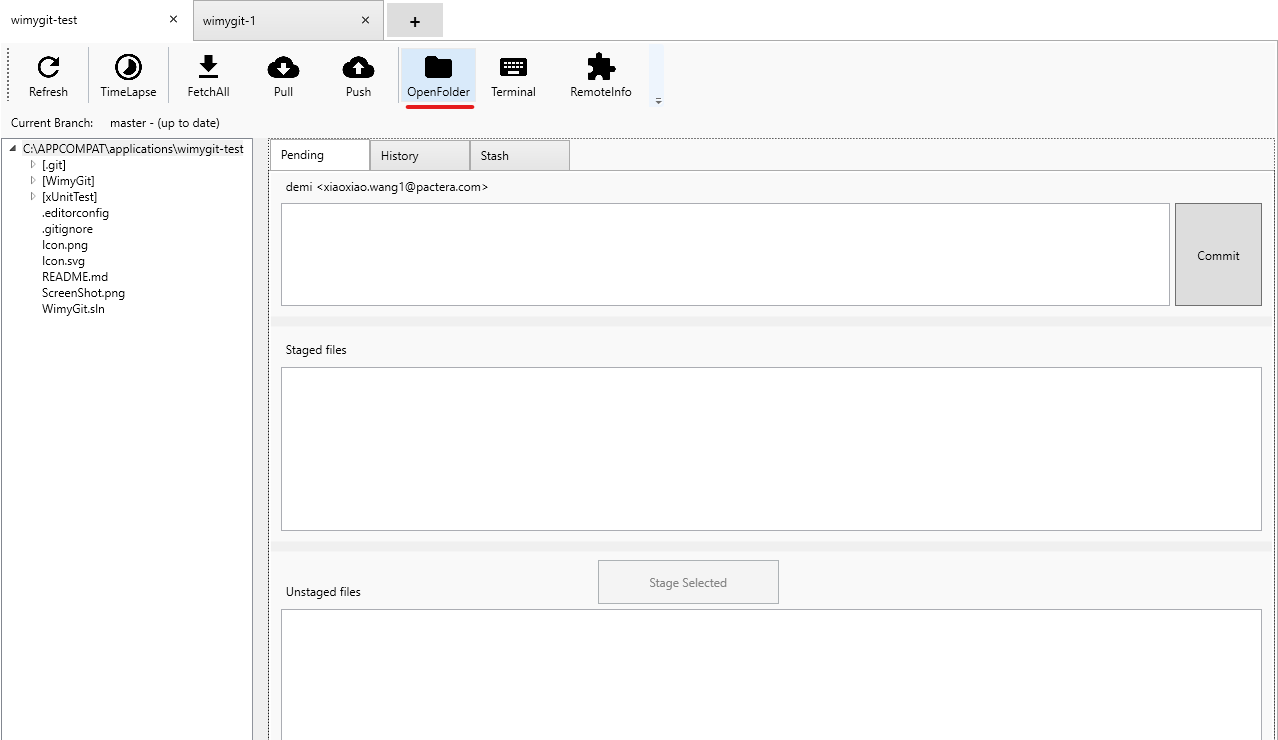
**Clean Environment:NA**

**Test Steps:**

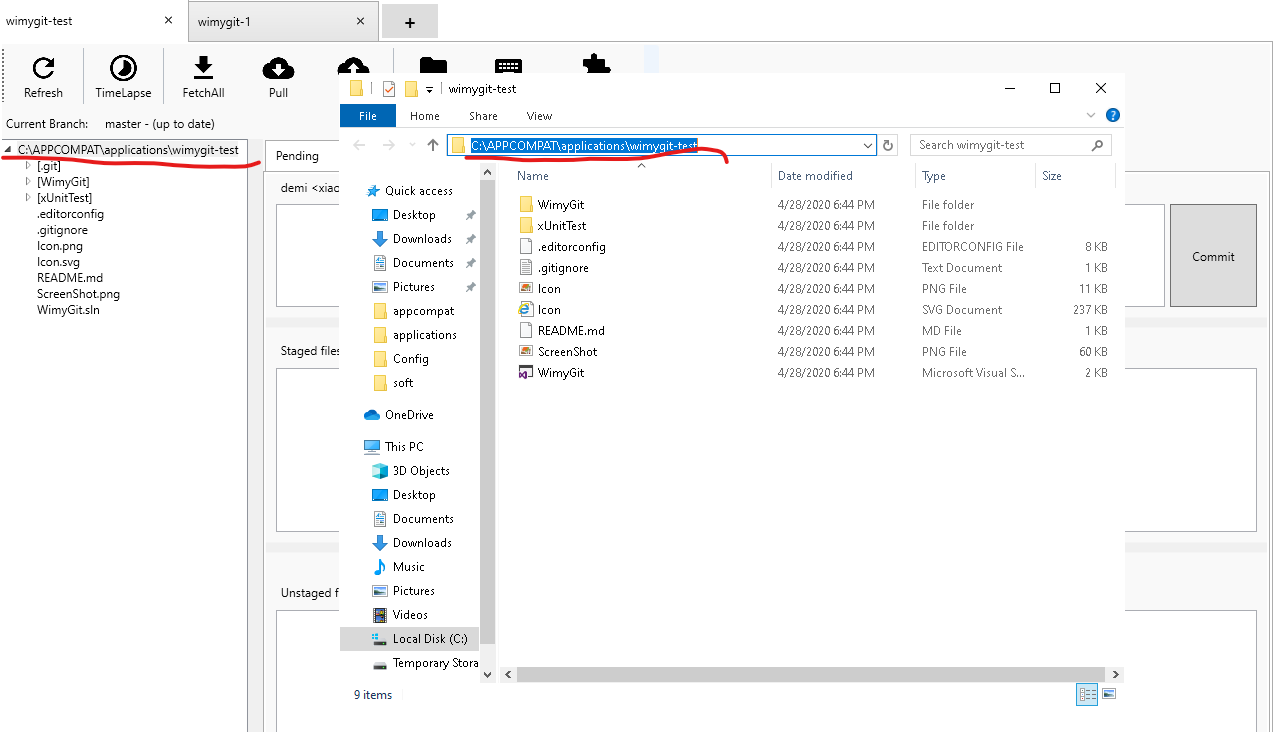
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



1. Click “OpenFolder” button.



1. Verify the path of Openfolder is same as the app path



## Scenario 7: Test Terminal

**Case ID: APP\_3166\_007**

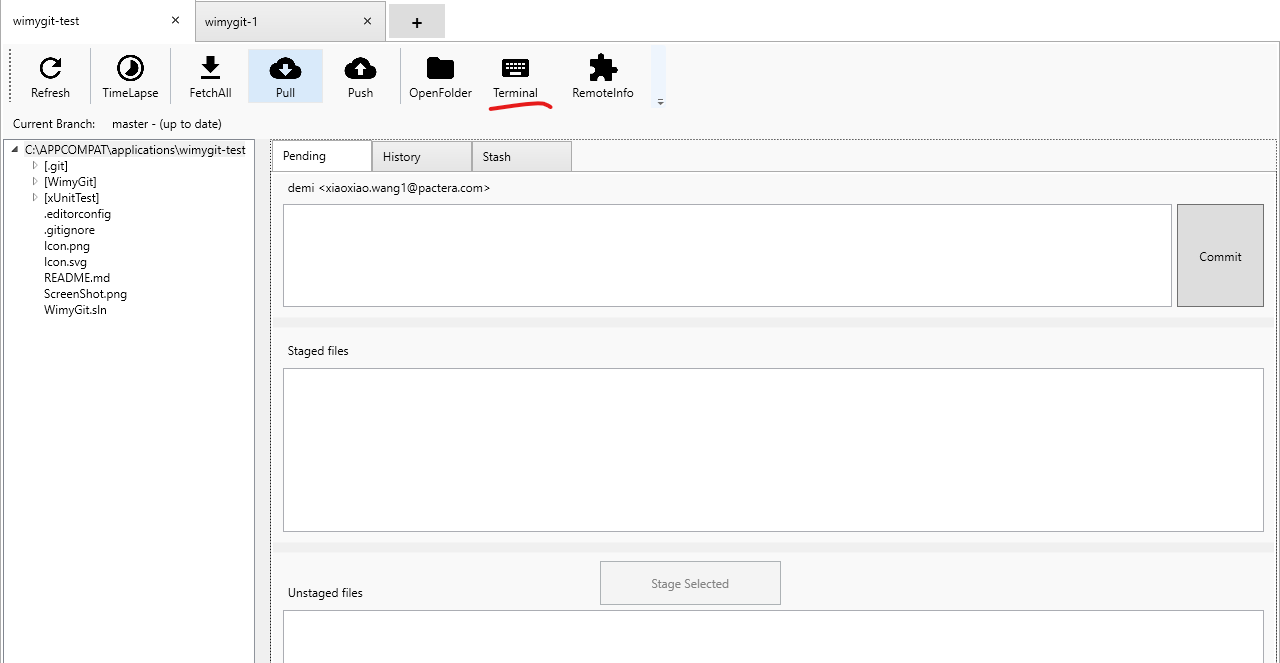
**Clean Environment:NA**

**Test Steps:**

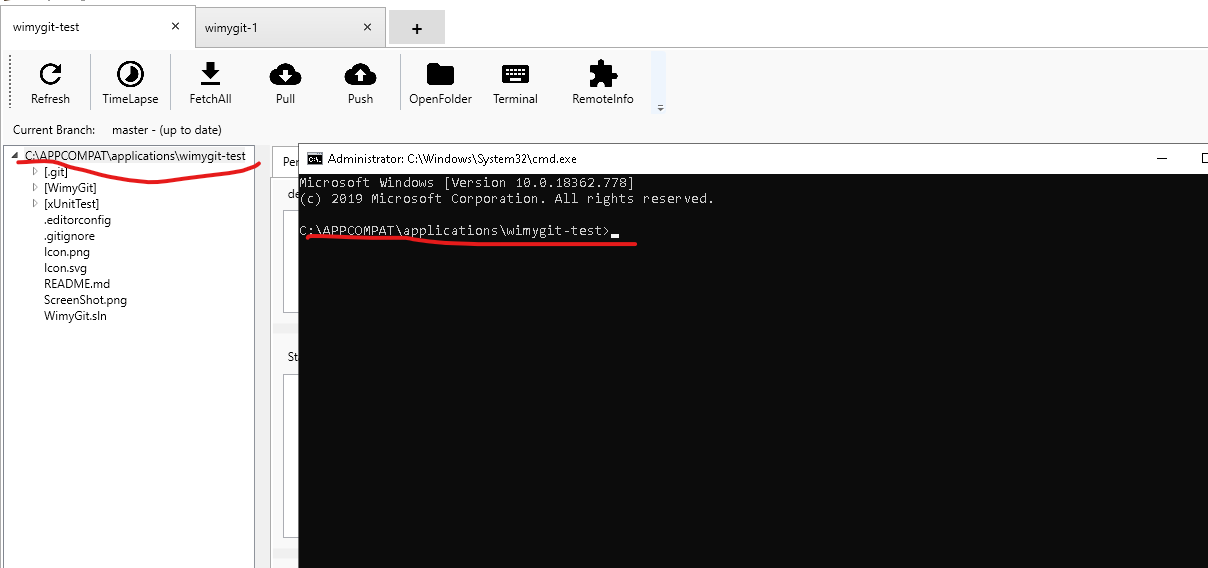
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



1. Click “Terminal” button.



1. Verify the path of Terminal is same as the app path



## Scenario 8: Test RemoteInfo

**Case ID: APP\_3166\_008**

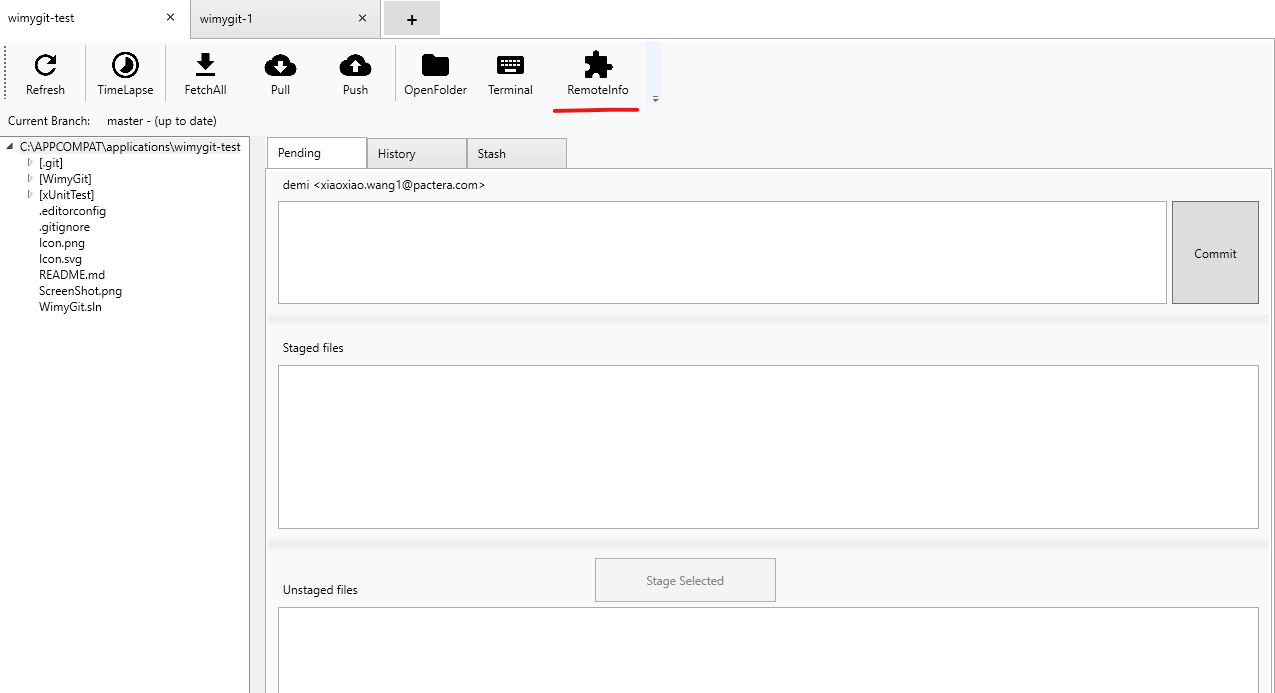
**Clean Environment:NA**

**Test Steps:**

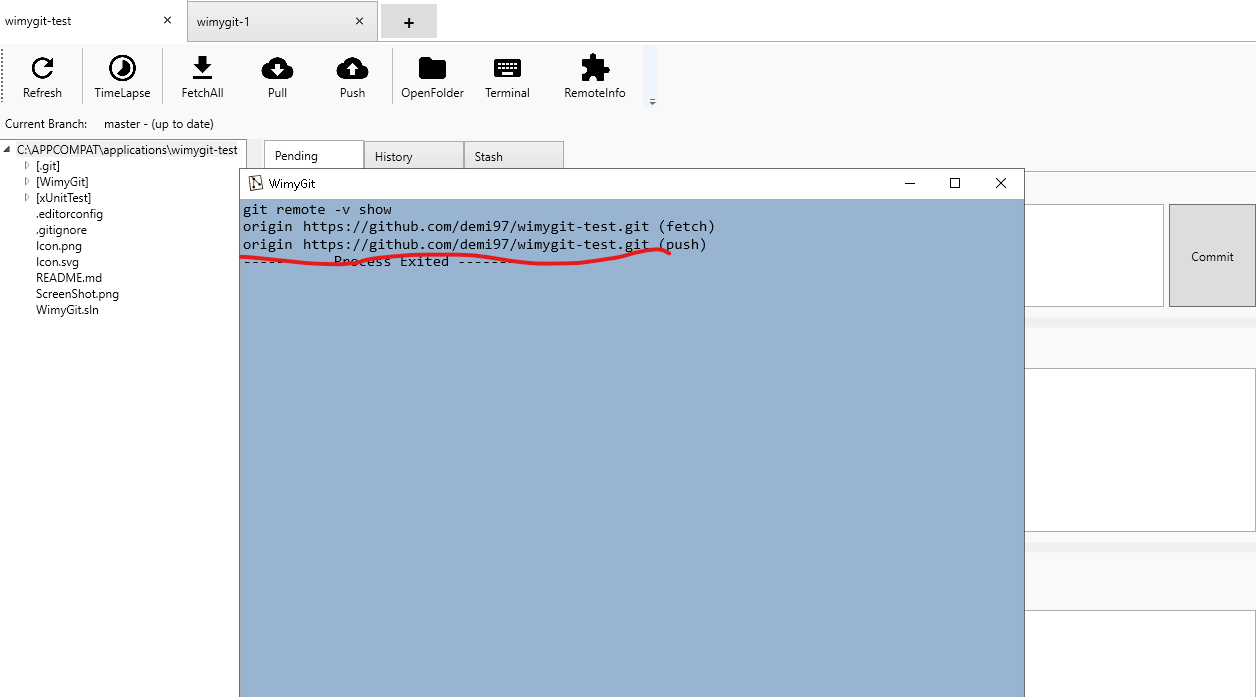
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



1. Click “RemoteInfo” button.



1. Verify the path of remote server is right.



## Scenario 9: Test Stash

**Case ID: APP\_3166\_009**

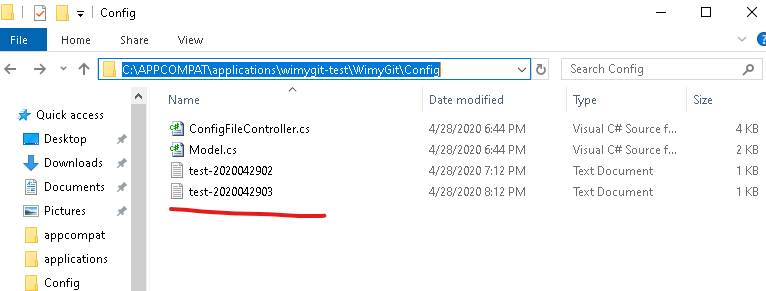
**Clean Environment:NA**

**Test Steps:**

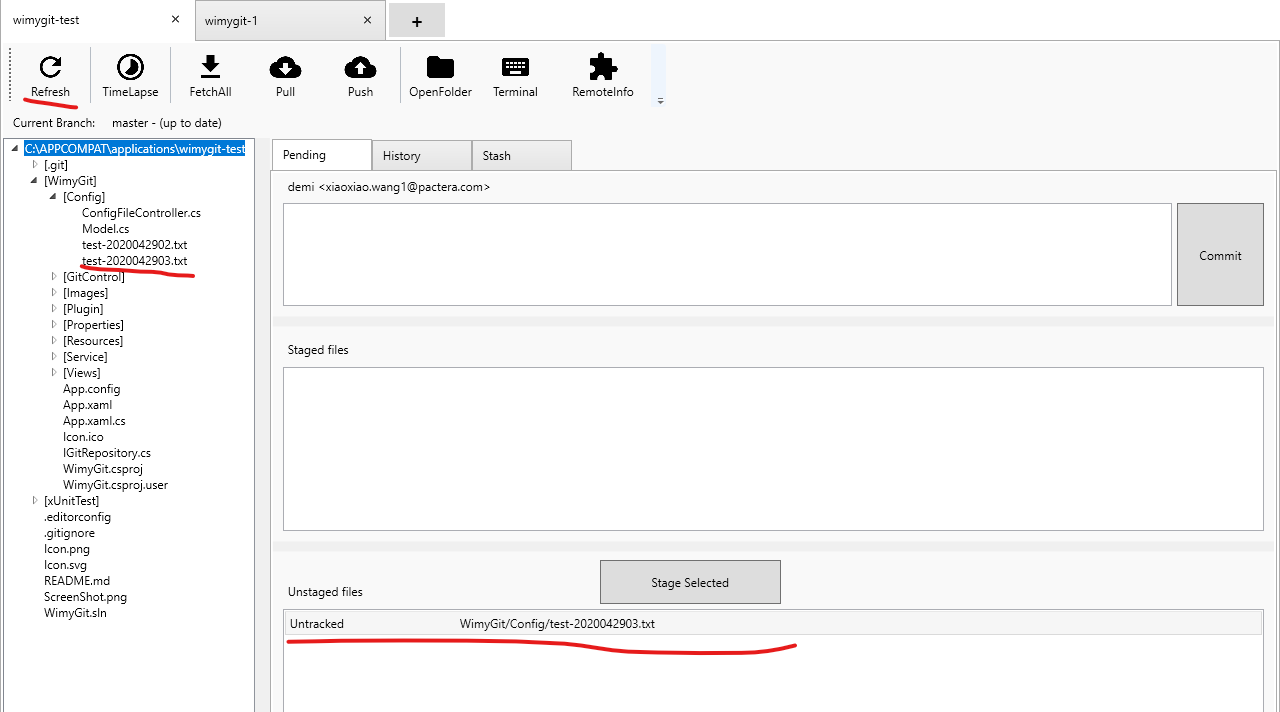
1. Launch appliciaton from "%APP\_CLIENT\_ROOT%\application\WimyGit\WimyGit.exe "
2. Drag wimygit source code from "%APP\_CLIENT\_ROOT%\wimygit-test”into here



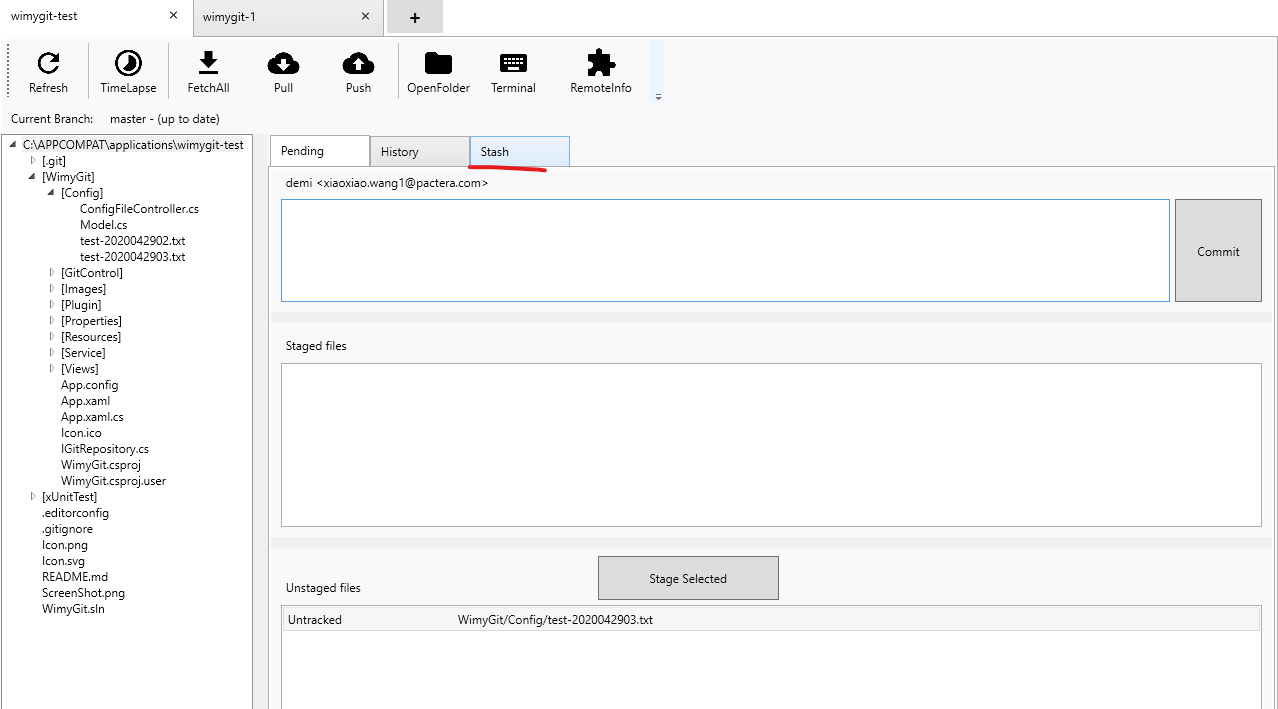
1. Create a new file in “%APP\_CLIENT\_ROOT%\wimygit-test\WimyGit\Config” folder.



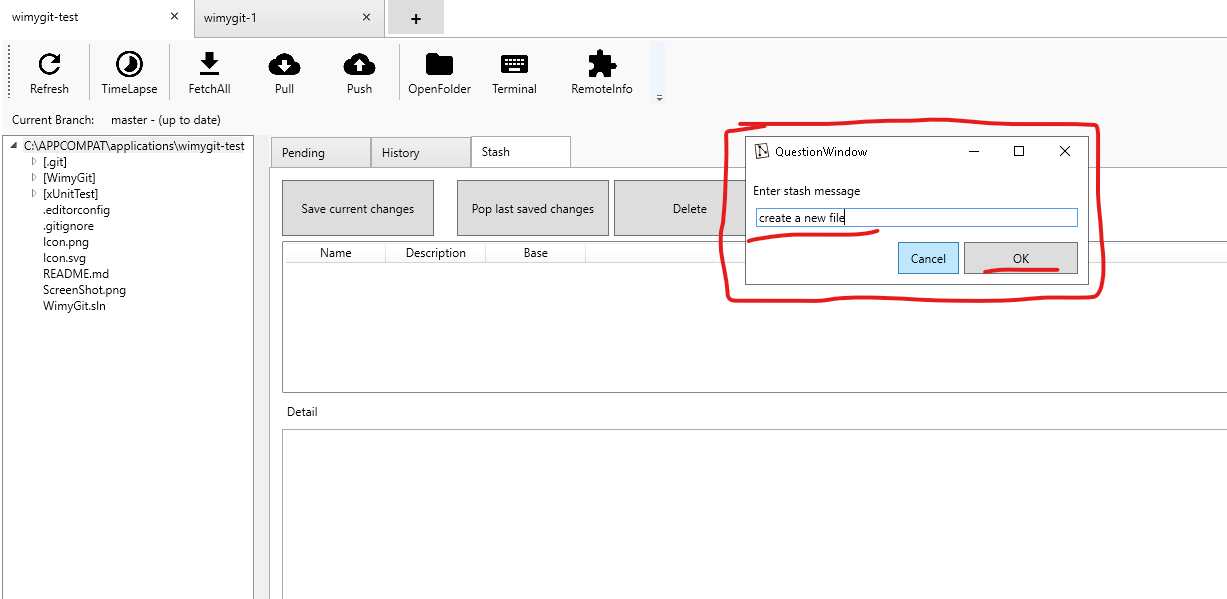
1. Click “refresh” buttton, the new file will show in the unstaged files area.



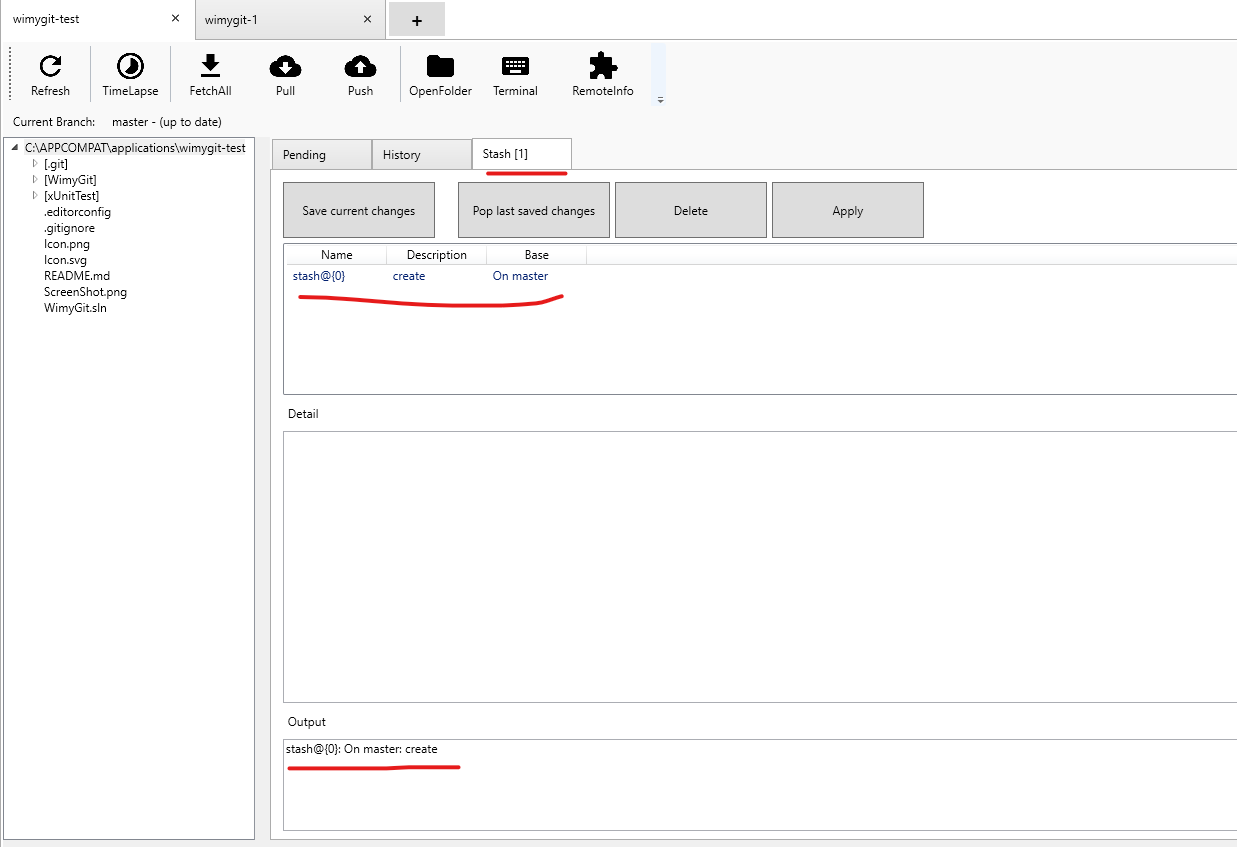
1. Click “Stash” button



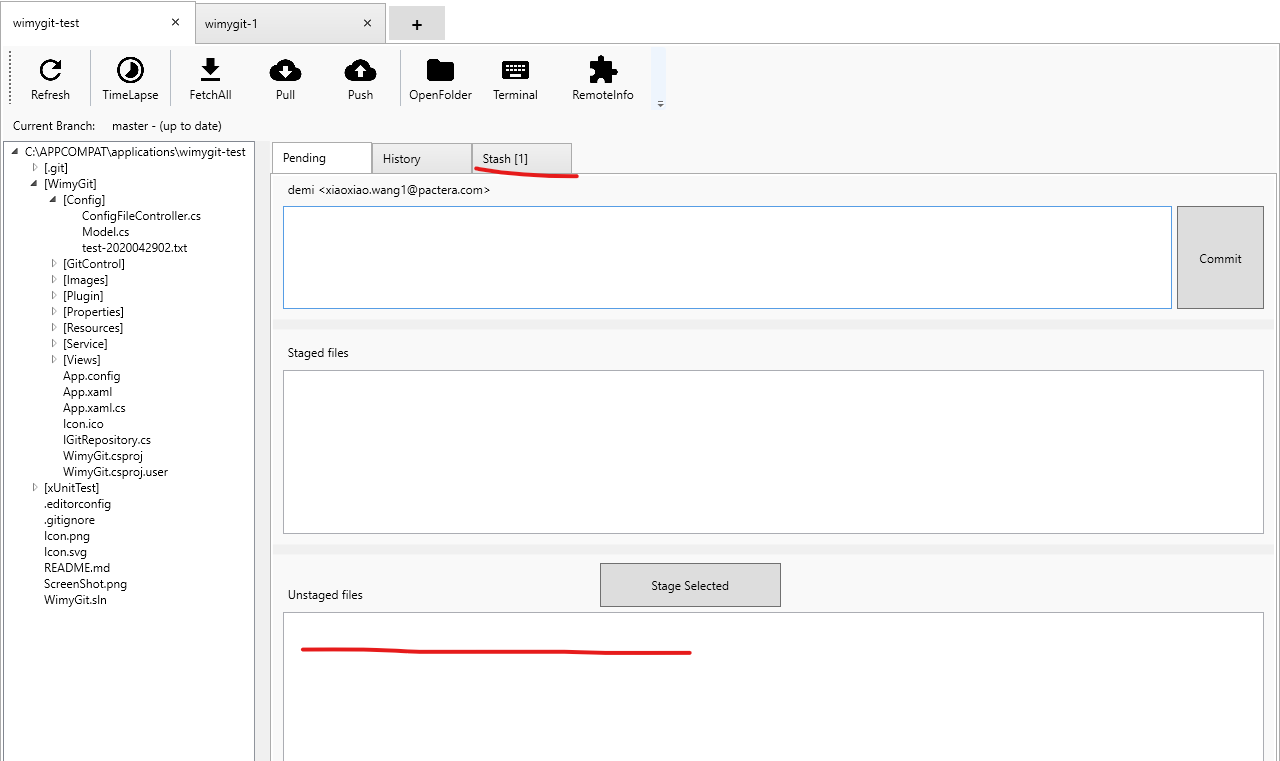
1. Click “Save current changes” button.it will pop up a pop-up box,enter stash message,and click “ok” button.



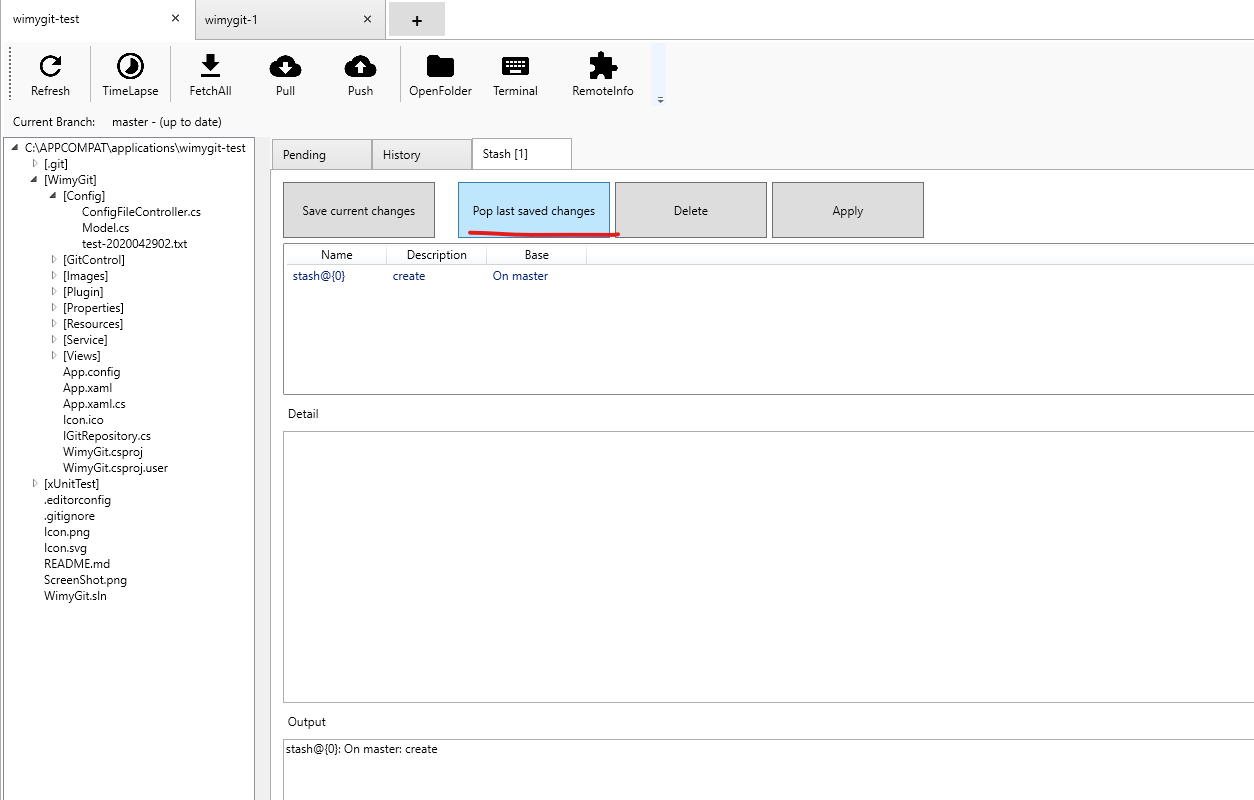
1. The stash page will be show like below



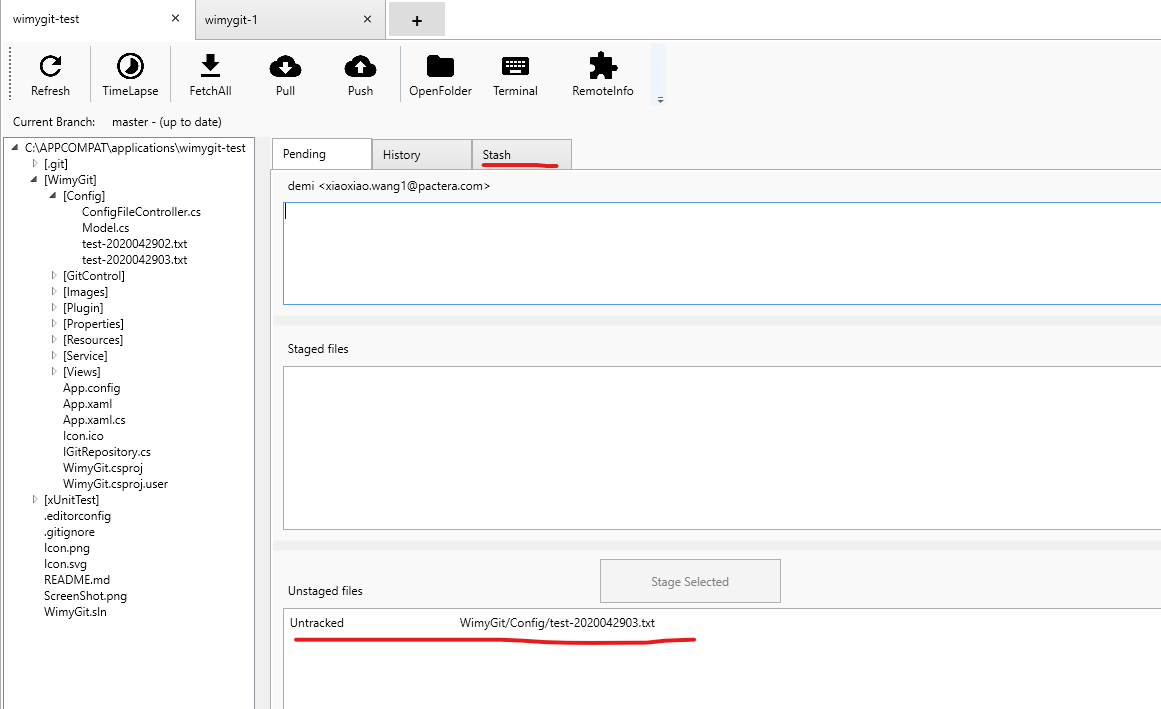
1. Verify the file in unstaged files area is disappear and stash in stash area



1. Switch to stash area and click “pop last saved changes” button



1. Switch to pending area, verify the file stashed in stash area before and show in unstaged files area now.



# Issues

No issue