Git Basic

Release 1 Test Plan

Prepared by Matthew Ristine

February 22, 2018

COMP 495

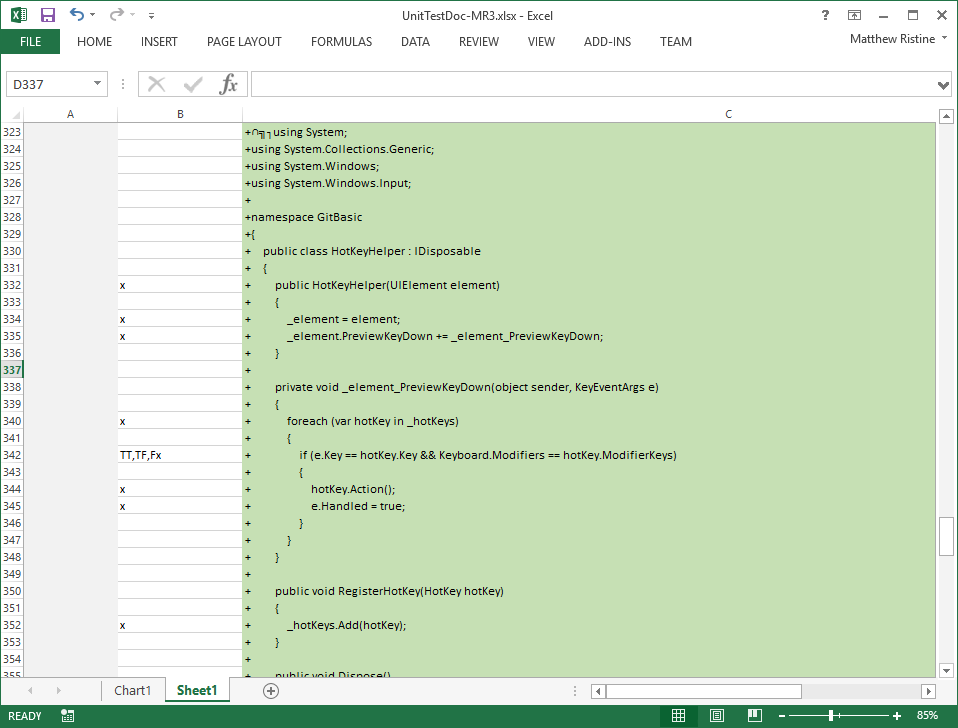
# Test Types

There are 3 kinds of testing which will be completed on this first release of Git Basic:

* Unit testing
* Functional testing
* Validation testing

## Unit Testing

* All unit tests will be performed manually.
* All code from feature branches must be unit tested before being merged into the main develop.
* Unit test coverage must be documented. This documentation will be checked in to the repository under: *<RepoRoot>/Documentation/Test/CompletedTests*
* The *UnitTestDoc.xltx* template (found in *<RepoRoot>/Documentation/Test*)should be used to document manual unit tests.
* Steps for test execution and documentation:
  1. Get a diff of all changes made in the current feature branch against the current state of develop. The *ut.bat* in the repository root may be run to obtain this diff.
  2. Paste diff into excel template (*UnitTestDoc.xltx*).
  3. Execute code in Visual Studio debugger inspecting unit inputs and outputs (use breakpoints) to ensure expected behavior and correctness. All branches of conditionals should be checked for 100% coverage.
* Here is a sample completed *UnitTestDoc* and screenshot thereof*:*



## Functional Testing

* Functional tests will be focused on the four primary components of the application:
  + Console Control
  + Diff Viewer
  + File Status Control
  + Status Bar
* Functional testing will be manual.
* The test cases listed in the functional test documents will be executed, and the results will be recorded in the documents.
* Here are the functional test documents. They can be found in *<RepoRoot>/Documentation/Test*



* The test results will be checked in to the repository under: *<RepoRoot>/Documentation/Test/CompletedTests*
* Unfuddle tickets will be created to log any bugs or improvement ideas. These tickets should list the following details:
  + Indication of type: Bug/Improvement
  + Severity
  + Explanation of the bug/improvement
  + If a bug, step-by-step reproduction.
* After functional testing is complete and bugs are fixed, the test cases directly associated with the bug will be rerun. The test results for this bug fix verification will be logged in the associated bug Unfuddle ticket.

## Validation Tests

* Validation testing will be performed by the technical lead (Matt Ristine).
* Validation testing will consist of using the app in production for 1 week.

## Notes

### Validation versus Functional Testing

Validation testing differs from functional testing in the following ways:

Validation testing is focused on using the tool exactly like the end user. Therefore validation testing generally focuses on the happy path and the most common functionality. It is focused on usability and whether the application is fulfilling its purpose.

Functional testing is focused on all functionality. As such, functional testing not only tests the happy path but also focuses on the corner cases. It targets all features. It is a rigorous and active exercise of all functionality instead of just the primary use cases.

### Test Exclusions

The following interfacing components will not be explicitly tested. They are only tested in as much as the application’s features exercise them:

* Git
* Cmd.exe
* LibGit2Sharp (3rd party api)
* GitSharp (3rd party api)

# Test Schedule

* Unit testing is carried out during feature development.
* All functional tests will be executed in week 8 (3/12/18 to 3/18/18).
* Validation testing will also occur in week 8 (3/12/18 to 3/18/18).
* Bugs will be fixed and retested during week 9 (3/19/18 to 3/25/18).
* All test documentation should be completed, approved, and submitted to the repository by 3/25/18.