BestXYZ Processor

Requirements, Design, Implementation, Testing (RDIT)

|  |  |
| --- | --- |
| User Story | Text |
| OUTPUTFILE | The user was able to specify an Output File in the main window by direct entry |
| OUTPUTSELECT | The user was able to select an existing Output File in the main window by browsing |
|  |  |

1. Requirements
   1. The user is able to type an output file path into a text field on a GUI interface
   2. The user is able to browse for an output file through a standard save file dialog
2. Design
   1. Use a framework to create a main GUI dialog
   2. Add text entry field to allow the user to type in a file path
      1. The text entry field will have placeholder text until the field is otherwise populated
   3. Add a button to the main GUI dialog to allow the user to browse for an output file
      1. Browsing for a file will populate the text entry field
3. Implementation
   1. Create a QT Window
   2. Add a QTextEdit that has an example file path as placeholder text
   3. Add a QButton to the GUI
      1. Upon clicking the QButton prompt the user with a save QFileDialog
      2. Browsing for an output file populates its corresponding QTextEdit and highlights the border of the QTextEdit green
4. Testing  
   Test Setup: A computer with the BestXYZ Processor application loaded and containing at least one reference receiver dataset.
   1. Open the BestXYZProcessor application
   2. Verify that an Output File text entry field is present on the GUI
   3. Verify that the user can type into the text entry field
   4. Verify that a button with a folder icon exists to the right of the text entry field
   5. Use the folder button to pop up the standard save file dialog
   6. Cancel the dialog, verifying that no file path is entered in the text entry
   7. Reopen the standard save file dialog
   8. Browse for a file and verify that the file path is automatically entered into the text field to the left of it.
   9. Verify that the border of the text entry field is changed to green