Numeric Converter

Technical Requirements Specification

Written by Zachary Babcock

Date: April 2, 2018

1. Purpose and Overview
   1. The purpose of the Numeric Converter is to allow users to input either a Roman numeral or an Arabic number, and convert between those two formats.
2. Development Summary
   1. This program will be developed as a Windows form application that will be installed on various different computers. The program will be developed in Microsoft Visual Studio, using the programming language C#.
3. Program Scope
   1. This program should not access any external files that the user will have stored before or after installation, nor should it create any new files during runtime to access.
4. User Interface
   1. This program will be contained within two window forms, consisting of a main form that contains a text field for the user to input their desired value to be converted, a text field for the output of said conversion to display in, a single button to initialize the conversion, a button to clear the two text fields that were previously mentioned, and a button to exit out of the application.
   2. A reference table that displays all the Arabic numbers and their respective Roman numerals, up to 3,999, should be included in the program in the other form. This form will exist for reference’s sake. It should be very clear in its purpose, so a user that knows very little about Roman numerals would understand what each Roman numeral translates into.
   3. The form should be able to recognize whether the user inserted an Arabic number or a Roman numeral without the need for the user to specify which sort of conversion they’re looking for.
5. Security
   1. Due to the nature of the program, there will be no special security measures taken with the creation of this program.
6. Program Flow
   1. When the user begins the program, the user should see a window form that contains a text box labeled as a spot for their input, a text box to contain the output of the conversion, and a few button controls at the bottom.
   2. The user can input either an Arabic number or a Roman numeral into the top, input text box, and then click a conversion button to convert the value they inputted to its respective counterpart. This button should be a bit more prominent than the other buttons on the form, due to its importance, and its repeated use.
   3. Once the user has done a conversion and wishes to clear the work area, they should be able to click a button that clears the input text box and the output text box on the main form.
   4. If the user wishes to see a list of conversions, they should be able to click another button on the main form to open another form that contains such a list of conversions. This new form will take the focus from the main form, until it is closed or the user clicks on the previous, main form.
   5. The secondary conversion list form should have a button that closes only the conversion list form somewhere on it.