Tech Specs

Hunter Duerr

1. Purpose and Overview
   1. Roman numeral conversion application, allow users to input either a Roman Numerals or Arabic integer, the resulting output is a conversion to the opposite number system.
   2. Roman numeral converter is to be user friendly and intended to be used by all age groups.
   3. The application maybe used as a quick reference for learning the symbols of the Roman Numeral System and their corresponding value and/or making quick conversions between the two systems for professional work or personal use.
2. Development Summary
   1. Planning and requirements analysis documentation has been created with goals and deliverables specified.
   2. The requested program will be developed for Windows based PC’s using C# Forms. Furthermore the mobile app, including windows user interface will be developed with Xamarin using virtual studio.
   3. Once the code has been written, the student developers will analyze the code and make comments in the code for identifying the processes and making future improvements easier.
   4. Next the application will be tested by other students assigned by the instructor. Once the code has been signed off by the tester and any final software configurations made, the application will go live on the internet and app stores for consumers.
3. References
   1. Appendix B – Program Flow
   2. Appendix C – Agile Software Development
4. Definitions
   1. Microsoft .NET Framework - [Software framework](https://en.wikipedia.org/wiki/Software_framework) developed by [Microsoft](https://en.wikipedia.org/wiki/Microsoft) that runs software primarily on [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows).
   2. GUI (Graphical User Interface ) – Type of [user interface](https://en.wikipedia.org/wiki/User_interface) that allows [users](https://en.wikipedia.org/wiki/User_(computing)) to [interact with electronic devices](https://en.wikipedia.org/wiki/Human%E2%80%93computer_interaction) through graphical [icons](https://en.wikipedia.org/wiki/Computer_icon) and visual indicators instead of [text-based user interfaces](https://en.wikipedia.org/wiki/Text-based_user_interface),
   3. Windows Forms - (WinForms) is a graphical (GUI) class library included as a part of Microsoft [.NET Framework](https://en.wikipedia.org/wiki/.NET_Framework) providing a platform to write rich client applications for desktop, laptop, and tablet PCs.
   4. Xamarin - [C#](https://en.wikipedia.org/wiki/C_Sharp_(programming_language))-shared codebase, developers can use Xamarin tools to write [native](https://en.wikipedia.org/wiki/Native_(computing)) [Android](https://en.wikipedia.org/wiki/Android_(operating_system)), [iOS](https://en.wikipedia.org/wiki/IOS), and [Windows](https://en.wikipedia.org/wiki/Windows) [apps](https://en.wikipedia.org/wiki/Mobile_app) with native [user interfaces](https://en.wikipedia.org/wiki/User_interface) and share code across multiple platforms, including [Windows](https://en.wikipedia.org/wiki/Windows) and [macOS](https://en.wikipedia.org/wiki/MacOS).
5. Deployment and Packaging
   1. This program will be available free to download on Github, the mobile device app on the windows app store.
6. Security
   1. This program will require users to create an account. After creating an account, the user will be able to input their username and password to gain access to the converter.
7. Data
   1. This program will generate the opposite format for the inputted value.
8. Program Flow
   1. Login Form
      1. When the user runs the program, they will be prompted with a login in form. If they do not have an account, they must create an account via the “Create account” button on the Login form. After creating an account, the information will be stored onto a local sql server. The user will then be able to input both their username and password to gain access to the converter.
         1. Login form contains two text boxes (tbUser and tbPass). These fields are where the user will input their username and password. These fields will be what fills the parameters in the stored procedure spUserAccount.
         2. Login form contains a check box called cbShowPass. When this box is checked, the user will be able to view the text without the password char in the tbPass field.
         3. Login form contains three buttons. btnLogin, btnCreateAcc, and btnExit. btnLogin will be pressed after inputting user information. If user information is correct, the Login form will disappear and the converter will appear. If incorrect, a label named lblError will appear to indicate the username/password was incorrect. btnCreateAcc will bring the user to the CreateAccount form where they can enter information to create an account. This information will be stored in a SQL Database named “Numeric” inside of a table named “Accounts”. btnExit will close the form and exit the program.
   2. Create Account Form
      1. When the user clicks on btnCreateAcc on the Login Form, they will be redirected to this form. Inside of this form, they will find four labels next to four text boxes to indicate to the user what to input in the text box. The text boxes are called tbFirstName, tbLastName, tbUser, and tbPass. They will input their First name, last name, username, and password inside of these four text boxes. After inputting the information, the user will click on the button btnCreateAcc on the Create Account form. The information inside of the text boxes will then be inserted into the SQL Database “Numeric” inside a table named “Accounts”. This is done through a stored procedure on the database named spNewUser which contains parameters “@user, @pass, @firstName, and @lastName”. The text stored in tbUser will supply @user parameter, text stored in tbPass will supply @pass and so on and so forth.
      2. After creating an account, the four text boxes will become Read Only. btnCreateAcc will become invisible. The form will increase in height to reveal a label to indicate that their account has been successfully created. Another button will appear (btnLogin) that will close the Create Account form and reopen the Login form upon click. This is done to prevent any accidental account creations. Also, to prevent spam.
      3. There is a check box on the Create Account form that shows the password in tbPass when checked.
   3. Default Form
      1. After successfully logging into the application, users will be prompted with the Default form. This is essentially the home page for the program. On this page will contain four buttons: btnArabic, btnRoman, btnLogout, and btnExit. There will be a Menu Strip at the top of the form named msMenu. This menu will be displayed on other forms such as Form1(Roman Converter) and Arabic Converter. Inside of msMenu, there will be two menus. One menu is “File” and the other menu is “Help”.
         1. Buttons
            1. btnArabic: Opens Arabic Converter form and closes Default form.
            2. btnRoman: Opens From1 (Roman Converter) and closes Defualt form.
            3. btnLogout: Logs the user out, closes Default form, and returns them to Login form.
            4. btnExit: Exits the application
         2. Menu Strip
            1. msMenu

File: General commands

Exit: Exits the application

Logout: Logs the user out, closes Default form, and returns them to Login form.

Help:

Get Help!: Opens Help form but keeps Default form open.

* + 1. Form1 (Roman Converter)
       1. This form will contain a Menu Strip, three buttons, two labels, and one text box.
          1. Menu Strip (msMenu)

File: General Commands:

Home: Closes Form1 and returns the user back to Default form

Logout: Logs the user out, closes Form 1 and returns the user to Login Form.

Exit: Exits the application

Converter: Change converter

Arabic: Closes Form 1 and opens Arabic Converter

Help

Get Help!: opens Help form

* + - * 1. Text box (romanNumeralTB)

Enter the Roman Numeral that you want to be converted into an Arabic Number

* + - * 1. Labels:

outputLB: outputs the converted number

lblRoman: Tells the user what to enter into the text box (romanNumeralTB)

* + - * 1. Buttons:

calculateBTN: converts the value in the textbox to Arabic and displays it into outputLB.

Button1 (Clear): clears text inside of romanNumeralTB and outputLB

btnExit: Exits the application

* + 1. Arabic Converter Form
       1. This form contains a menu strip, four labels, three buttons, and one text box.
          1. Menu Strip (msMenu)

File: General Commands:

Home: Closes Form1 and returns the user back to Default form

Logout: Logs the user out, closes Form 1 and returns the user to Login Form.

Exit: Exits the application

Converter: Change converter

Roman: Closes current form and opens Form 1 (Roman Converter)

Help

Get Help!: opens Help form

* + - * 1. Text box (arabicNumeralTB)

Enter an Arabic number that will be converted to a Roman Numeral

* + - * 1. Labels

lblInvalid:

If user enters a number inside of arabicNumeralTB that is greater than or equal to 4000, the label will display “Number exceeded 3999!” in red.

If user enters a number between 1-3999, the label will state “Successful” in green.

lblOne: indicates the lowest value that should be entered. When a user enters a number greater than or equal to 4000, the color will change to red. When user enters in a correct value, color will change to green.

lblThree: indicates the highest value that should be entered. When a user enters a number greater than or equal to 4000, the color will change to red. When user enters in a correct value, color will change to green.

Label3: Displays the converted number

* + - * 1. Buttons:

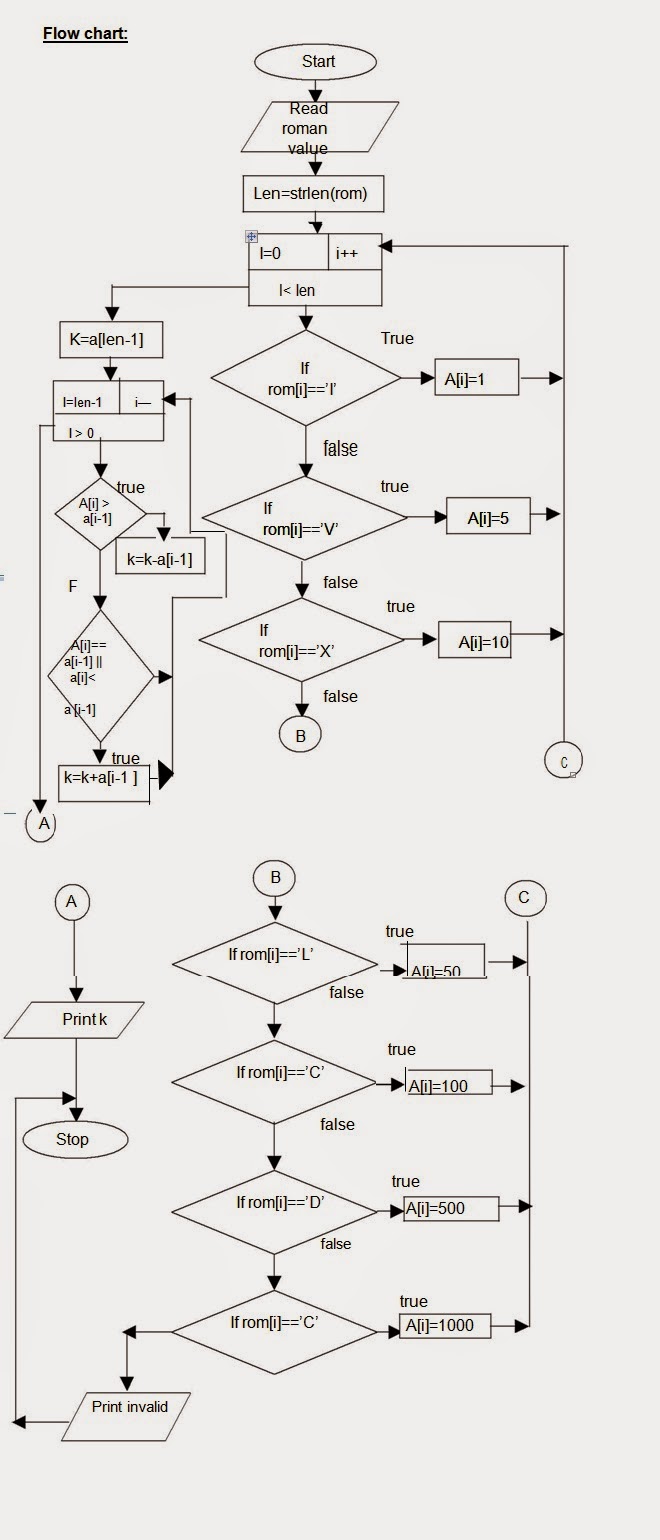
Calculate2Button: Converts the Arabic number in the text box to a roman numeral that will be displayed in Label3.

Button1: Clears text inside the text box and Label3

btnExit: Exits the application.

* + 1. Help Form
       1. Displays a label that gives helpful advice to the user if they experience trouble.

**Appendix C – Program Flow**



**Appendix C – Agile Software Development**

