Assignment: Validating input

Regular expressions are a powerful way to determine if a user's input is valid, but also to detect certain patterns in a text. This assignment consists of small tasks requiring you to use the Regex-class to explore these benefits. Make use of the start-up project to create an application that follows the user interaction.

User interaction

A user should be able to do the following things:

- By clicking one of the buttons in the GroupBox *Regex: IsMatch:*
  - A string is supplied and checked if it is a valid (dependent on the button):
    - Phone number. A regional phone number can only exist out of digits and spaces
      E.g. 06123456789 and 06 123 456 789 are valid, while 0612E456789 and 06-123456 789 are not valid.
    - Dutch phone number. A Dutch phone number must start with +31 followed by 9 digits
      E.g. +31612345678 and +31401234568 are valid, while 316123456789 and +31 612345 6790 12345 are not valid.
    - Street name. A street name can only exist out of letters, a .(dot) and spaces.
      E.g. Rachelsmolen and John F. Kennedylaan are valid, while R@chelsmolen and John F. Kennedylaan 50 are not valid.
    - Address. An address follows the same format as a street name and in addition has a house number. A house number must start with a digit and may end with a letter.
      E.g. Rachelsmolen 5 and John F. Kennedylaan 50a are valid, while Rachelsmolen and John F. Kennedylaan a are not valid.
  - Show a message whether the input is valid/invalid.
- By inputting a pattern in GroupBox *Regex: Match* and conforming:
  - The amount of matches found in the TextBox should be shown in a MessageBox;
    - E.g. if the pattern is ipsum, a count of 5 should be displayed, if the pattern is [a-zA-Z]{10,12}, a count of 33 should be displayed.
  - The inputted pattern should also be added to the ComboBox.