```
using System;
using System.Collections.Generic;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;


/*
 * Title:   PageActivity CS
 * Author:  Paul McKillop
 * Date:    March 2020
 * Purpose: Code behind for functionality
 */


namespace GymTrackingV
{
    /// <summary>
    /// Interaction logic for PagePerson.xaml
    /// </summary>
    public partial class PagePerson : Page
    {
        internal bool formDataValid = false;

        public PagePerson()
        {
            InitializeComponent();

            ClearControls();

            //-- DEBUG Data. Remove on completion
            //PersonNameTextBox.Text = "Paul";
            //PersonAgeTextBox.Text = "67";
            //PersonWeightTextBox.Text = "170";
        }

        /// <summary>
        /// Events raised on click
        /// </summary>
        /// <param name="sender"></param>
        /// <param name="e"></param>
        private void PagePersonClearButton_Click(object sender, RoutedEventArgs e)
        {
            ClearControls();
        }

        private void ActivitiesButton_Click(object sender, RoutedEventArgs e)
        {
            //-- Get the form's data by using private method below
            //-- populate a Person object with the data gathered to pass on
            //-- to page PageActivity
            var personData = HarvestData();
            //-- Form data valid?
            if (formDataValid)
            {
                //-- Now have the data required.
                //-- can pass this to the Activity page
                //-- Include as an argument on the Navigate method
                var pageActivity = new PageActivity(personData);
                this.NavigationService.Navigate(pageActivity);
            }
        }

        /// <summary>
        /// Clears the interface controls and sets up for new
        /// data entry
        /// </summary>
        private void ClearControls()
        {
            PersonNameTextBox.Text = string.Empty;
            PersonAgeTextBox.Text = string.Empty;
            PersonWeightTextBox.Text = string.Empty;

            //-- set focus to name box. Logical GUI control.
            //-- Help the user where you can.
            PersonNameTextBox.Focus();
        }

        private Person HarvestData()
        {
            //-- data handler variables
            var tempPerson = new Person();
            var countOfValidFields = 0;

            //-- set the rule in one place
            var requiredValidFields = 3;

            //-- Use Try .. Catch
            try
            {
                //-- validate the name data and assign if there
                if (!string.IsNullOrEmpty(PersonNameTextBox.Text))
                {
                    tempPerson.PersonName = PersonNameTextBox.Text;
                    countOfValidFields += 1;
                }
                else
                {
                    MessageBox.Show("You must enter a name");
                }

                //-- validate the Age data
                if (!string.IsNullOrEmpty(PersonAgeTextBox.Text))
                {
                    tempPerson.Age = Convert.ToInt32(PersonAgeTextBox.Text);
                    countOfValidFields += 1;
                }
                else
                {
                    MessageBox.Show("You must enter your Age");

                }

                //-- Validate the weight data
                if (!string.IsNullOrEmpty(PersonWeightTextBox.Text))
                {
                    tempPerson.Weight = float.Parse(PersonWeightTextBox.Text);
                    countOfValidFields += 1;
                }
                else
                {
                    MessageBox.Show("You must enter your Weight");
                }

            }
            catch (Exception)
            {

                throw;
            }

            //-- Check if all required data present.
            if (countOfValidFields == requiredValidFields)
            {
                formDataValid = true;
                return tempPerson;
            }
            else
            {
                MessageBox.Show("Form data is not valid");
                PersonNameTextBox.Focus();
            }

            return tempPerson;

        }

    }
}

```