COP3503 Project 4 – Dietary Survey Form

Objectives:

- Show an understanding of how to
 - Create and layout graphical user interface components
 - Use events to create a dynamic and responsive interface
 - o Read UML diagrams and implement the classes and methods outlined in the diagram

Submission Requirements:

- Submit your project via the Canvas course
 - Upload a zip file containing your 3 class files specified below
 - The zip file should be named Project4_N#.zip
 - For Example: Project3 N00123456.zip
- Project requires 1 zip file to be submitted

Design Specification Requirements: 100 points

Use the Project 4 UML Diagram as an outline for the project.

- 1. You will need to create 3 java files (As outlined in the UML diagram). Project4, FileHandler, CustomJFRame
- 2. Project 4 will contain the main method were the CustomJFrame will be initialized and displayed
- 3. FileHandler will contain 2 methods. The constructor FileHandler() and writeResults(String surveyData)
 - a. FileHandler() constructor needs to create a csv file called survey_results.csv and create a header for the csv file
 - i. Header:
 - 1. DateTime,FirstName,LastName,PhoneNumber,Email,Sex,Water,Meals,Wheat,Sugar,Dair y,Miles,Weight
 - ii. See example in the survey_results_EXAMPLE.csv
 - b. writeResults(String surveyData) needs to append the surveyData string to the survey_results.csv file
 - i. This method will be similar to the logger method for project 3
- 4. CustomJFrame will contain the constructor method CustomJFrame() and 1 inner class InnerActionListener.
 - a. CustomJFrame() needs to initialize all the class variable components, configure them properly, and then add them to the frame using a layout manager
 - i. GridBagLayout manager is recommended but not required
 - b. InnerActionListener will contain 2 methods. actionPerformed() and clearForm()
 - i. actionPerformed() needs to clear the data of the form and if the submit button was pressed the data needs to be captured and written to the csv file
 - The submitted data needs to be written to the survey_results.csv file using the FileHandler writeResults() method
 - 2. The submit and clear buttons both need to register this action listener
 - ii. clearForm() needs to reset all fields to their default values

Note: Refer to the Project4_GUI screenshot file for an example of the needed GUI layout. Also, refer to survey_results_EXAMPLE.csv file to see the required format for the output csv file.

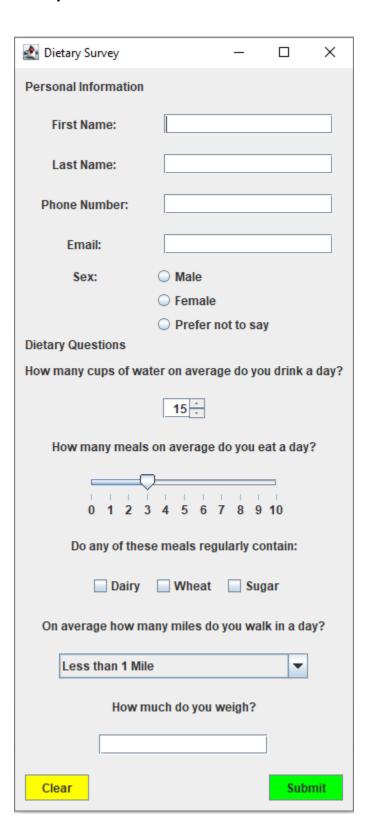
Minimum Requirement: -100 points

 All graphic components shown in the example GUI for Project 4 must be present and formatted in a logical manner

Additional Notes:

- The user should be able to hit submit or clear at any time without the program crashing
 - Even if they have not filled out any fields
 - See last row in example csv for an example of a submission with not data
- The CustomJFrame constructor should also initialize the FileHandler object
- Only one radio button should be selectable at a time
- JSpinner configuration: min = 0, max = 50, step = 1, default = 15
- JSlider configuration: min = 0, max = 10, default = 3
- JComboBox Options: Less than 1 Mile, More than 1 mile but less than 2 miles, More than 2 miles but less than
 3 miles, More than 3 miles
 - Default = Less than 1 Mile
- JFormattedTextField should only accept numeric values
 - This is the only input verification required for the project
- Your GUI does not need to match the example GUI format exactly, however, all components must be present and displayed in an aesthetically pleasing manner
- The format of your output file must match the format of the survey_results_EXAMPLE.csv file
- No console output is required for this project
- The FileHandler constructor should overwrite any existing file
- The writeResults method should append to the output file. It should not overwrite the existing data
- All input and output streams must be closed properly
- All input and output streams must be surrounded by a try catch
- The CustomJFrame class needs to inherit from the JFrame class
- The InnerActionListener needs to implement the ActionListener interface
- The submit button should also clear the form after the data has been submitted
- Follow the commenting and programming guidelines outlined in the Commenting and Programming guide documents
 - Failure to do so will result in up to a 25-point reduction

Example GUI:



UML Diagram:

