1) How long (in elapsed time) would it take you to implement this solution?

Maybe ~2 hours

2) How much overall effort (in person hours) will this take?

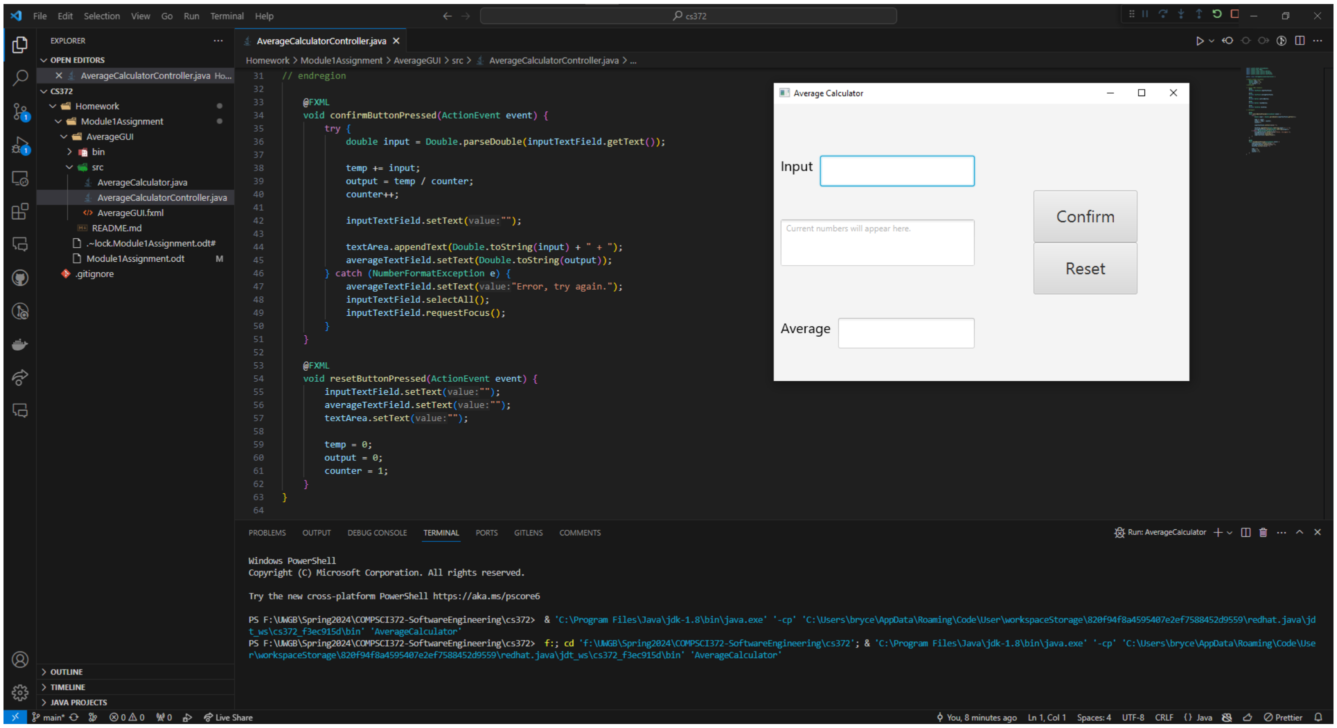
~Maybe 4-6 hours

3) How well will your solution match the problem? Discuss.

My solution will complete the problem as well as I can make it. There may still be bugs or edge cases that I miss, but that stuff happens. It will have a GUI, and a place to input text, as well as a non-editable text field that displays the average. There will be a button to "confirm" the current number into the average.

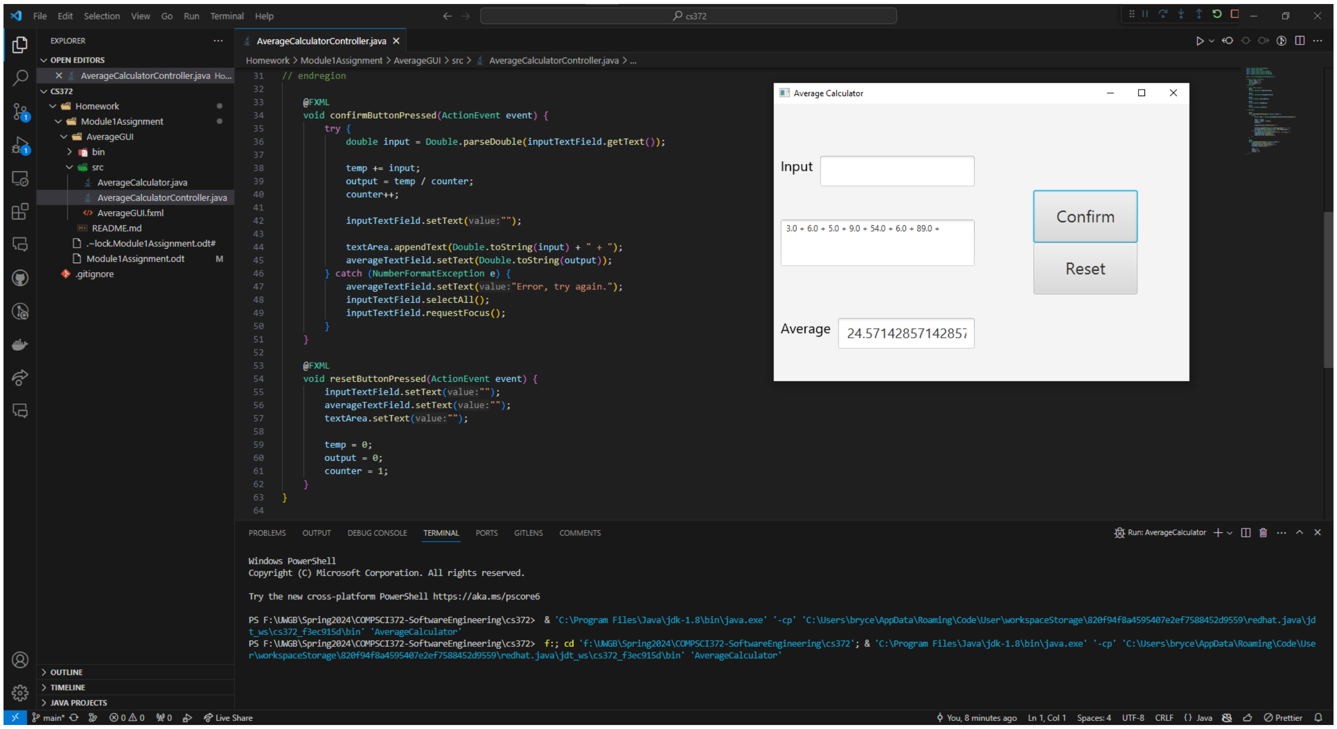
4) Design and implement your solution while keeping track of time. Add screenshot of the running application with the answer. Show the actual time and scheduled time in a table.

Task	Start Time	End Time	Total Time
Research/Design	11:05	11:30	25 minutes
Code	11:35	12:10	35 minutes
Total			60 minutes



```java
// endregion

    @FXML
    void confirmButtonPressed(ActionEvent event) {
        try {
            double input = Double.parseDouble(inputTextField.getText());

            temp += input;
            output = temp / counter;
            counter++;

            inputTextField.setText(value:"");

            textArea.appendText(Double.toString(input) + " + ");
            averageTextField.setText(Double.toString(output));
        } catch (NumberFormatException e) {
            averageTextField.setText(value:"Error, try again.");
            inputTextField.selectAll();
            inputTextField.requestFocus();
        }
    }

    @FXML
    void resetButtonPressed(ActionEvent event) {
        inputTextField.setText(value:"");
        averageTextField.setText(value:"");
        textArea.setText(value:"");

        temp = 0;
        output = 0;
        counter = 1;
    }
}
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS F:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372>  & 'C:\Program Files\Java\jdk-1.8\bin\java.exe' '-cp' 'C:\Users\bryce\AppData\Roaming\Code\User\workspaceStorage\820f94f8a4595407e2ef7588452d9559\redhat.java\jd
t_ws\cs372_f3ec915d\bin' 'AverageCalculator'
PS F:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372>  f:; cd 'f:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372'; & 'C:\Program Files\Java\jdk-1.8\bin\java.exe' '-cp' 'C:\Users\bryce\AppData\Roaming\Code\Use
r\workspaceStorage\820f94f8a4595407e2ef7588452d9559\redhat.java\jdt_ws\cs372_f3ec915d\bin' 'AverageCalculator'
```


AverageCalculatorController.java
Homework > Module1Assignment > AverageGUI > src > AverageCalculatorController.java > ...
Average Calculator
Input
Confirm
Reset
Average

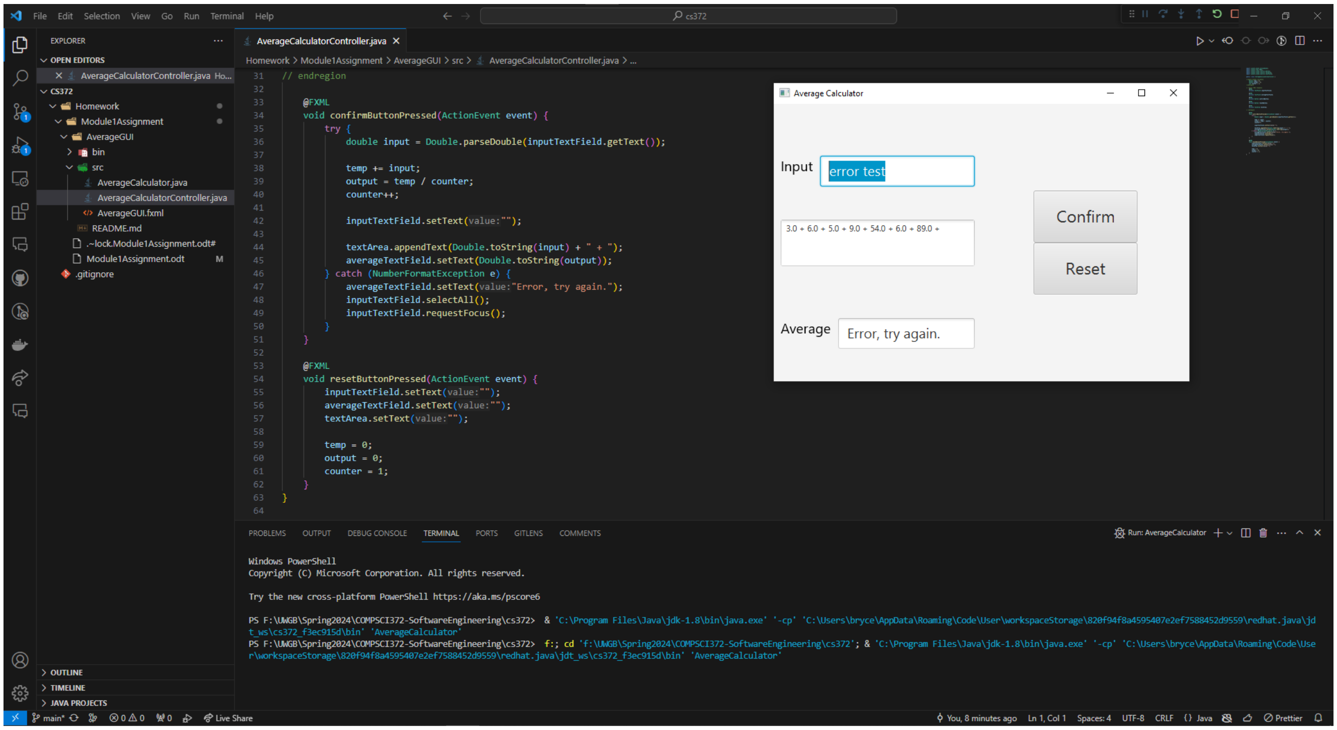
```java
// endregion

    @FXML
    void confirmButtonPressed(ActionEvent event) {
        try {
            double input = Double.parseDouble(inputTextField.getText());

            temp += input;
            output = temp / counter;
            counter++;

            inputTextField.setText(value:"");

            textArea.appendText(Double.toString(input) + " + ");
            averageTextField.setText(Double.toString(output));
        } catch (NumberFormatException e) {
            averageTextField.setText(value:"Error, try again.");
            inputTextField.selectAll();
            inputTextField.requestFocus();
        }
    }

    @FXML
    void resetButtonPressed(ActionEvent event) {
        inputTextField.setText(value:"");
        averageTextField.setText(value:"");
        textArea.setText(value:"");

        temp = 0;
        output = 0;
        counter = 1;
    }
}
```

Input:

3.0 + 6.0 + 5.0 + 9.0 + 54.0 + 6.0 + 89.0 +

Average: 24.5714285714285ׂ7

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS F:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372>  & 'C:\Program Files\Java\jdk-1.8\bin\java.exe' '-cp' 'C:\Users\bryce\AppData\Roaming\Code\User\workspaceStorage\820f94f8a4595407e2ef7588452d9559\redhat.java\jdt_ws\cs372_f3ec915d\bin' 'AverageCalculator'
PS F:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372>  f:; cd 'f:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372'; & 'C:\Program Files\Java\jdk-1.8\bin\java.exe' '-cp' 'C:\Users\bryce\AppData\Roaming\Code\User\workspaceStorage\820f94f8a4595407e2ef7588452d9559\redhat.java\jdt_ws\cs372_f3ec915d\bin' 'AverageCalculator'
```


AverageCalculatorController.java
Homework > Module1Assignment > AverageGUI > src > AverageCalculatorController.java > ...
Average Calculator
Input
Confirm
Reset
Average

```java
// endregion

    @FXML
    void confirmButtonPressed(ActionEvent event) {
        try {
            double input = Double.parseDouble(inputTextField.getText());

            temp += input;
            output = temp / counter;
            counter++;

            inputTextField.setText(value:"");

            textArea.appendText(Double.toString(input) + " + ");
            averageTextField.setText(Double.toString(output));
        } catch (NumberFormatException e) {
            averageTextField.setText(value:"Error, try again.");
            inputTextField.selectAll();
            inputTextField.requestFocus();
        }
    }

    @FXML
    void resetButtonPressed(ActionEvent event) {
        inputTextField.setText(value:"");
        averageTextField.setText(value:"");
        textArea.setText(value:"");

        temp = 0;
        output = 0;
        counter = 1;
    }
}
```

Input: error test

3.0 + 6.0 + 5.0 + 9.0 + 54.0 + 6.0 + 89.0 +

Average: Error, try again.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS F:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372>  & 'C:\Program Files\Java\jdk-1.8\bin\java.exe' '-cp' 'C:\Users\bryce\AppData\Roaming\Code\User\workspaceStorage\820f94f8a4595407e2ef7588452d9559\redhat.java\jdt_ws\cs372_f3ec915d\bin' 'AverageCalculator'
PS F:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372>  f:; cd 'f:\UWGB\Spring2024\COMPSCI372-SoftwareEngineering\cs372'; & 'C:\Program Files\Java\jdk-1.8\bin\java.exe' '-cp' 'C:\Users\bryce\AppData\Roaming\Code\User\workspaceStorage\820f94f8a4595407e2ef7588452d9559\redhat.java\jdt_ws\cs372_f3ec915d\bin' 'AverageCalculator'
```

5) How good is your code/design/documentation/testing? Explain.

The XML design is hard-coded, as I do not remember how to make it more dynamic. I chose to hard-code sizes as this is a small program that does not require anything too complex. There are 5 FXML element types that make this work. There are two TextFields for inputting numbers and outputting the average. There are two buttons, one confirms the input and allows the code to add the input number into a variable. The other button resets all of the values to their defaults. The final element is a TextArea, which shows the user their previous inputs.

The code is fairly simple, there are just three global variables keeping track of the numbers that come in. The “temp” variable is of a double type and it saves the current and all previous numbers by just continuously adding the input to the temp variable. The “output” variable takes the temp variable and divides it with the “counter” variable. This “counter” variable is incremented each time the confirm button is pressed, allowing the program to keep track of how many numbers have been input. There are then two methods that make up the program: confirmButtonPressed(…) and resetButtonPressed(…). The method confirmButtonPressed(…) handles the logic of adding numbers to the temp variable and dividing the temp variable with the counter for the output. It handles setting the TextFields when pressing the button. The method resetButtonPress(…) will reset the global variables, TextFields and TextArea to their respective defaults.

There is no documentation for the code as I think it is fairly simple and self explanatory. I would happily go back and provide some if it were necessary.

Testing included inputting numbers both as integers and as doubles. Both work flawlessly. Inputting any other string of text will be caught by the try-catch block that I added into the confirmButtonPressed(…) method. This will show the user that they input an incorrect input and prompt them to try again, without adding the error text to the variable.

Overall, I over-estimated how much time this would take. Only taking an hour seems short, but looking over the times I wrote down, it makes sense now. I could definitely take more time to make the GUI look better and theme it or something but it was not necessary.

If you want to look at any code for any reason, it is in a repository at:
https://github.com/dupergg/cs372