**Project Abstract:**

Design a Java GUI to allow entry and retrieval of weekly contributions to a local church. The program should allow the user to print a report listing each donor and the amount and ministry area for the current week and a summary page with total weekly deposit information. The purpose of this program is to allow the counters to communicate efficiently with the treasurer of the church while only allowing them to access the current week’s donor information (a privacy issue).

**Project Team Members:**

Kevin Lancaster

Jedidiah Bird

Dondra Crocker Richards

Brian Hinkle

Timia Davis

**Repository URL:**

<https://github.com/BigDaddyCodes/CSU-CSCI325_FALL2018>

**Detailed Description:**

Kevin: This is a GUI- and menu-driven program for use by the counters of the weekly offerings. When starting the program, the user is presented with a menu that allows them to input the week’s donations, input the miscellaneous data that is not tied to a particular donor, cross-check the inputted data to ensure it is correct, and view/print the reports required for the week.

Kevin & Jed: When the user selects the button to input the week’s donations, they are presented with a form that allows them to select the donor’s name and the donation “ministry” or “area” from lists and input the amount and type (“CASH” or Check #) of donation. The user is not required to scroll through a list of donor’s names, as the combo-box used for entry should auto-fill as they type. This is also true for the Ministry Area box. This entry method is intended to provide data validation. Additionally, the user should be able to add donors and areas, although the program should prompt them that they are trying to add and allow them to cancel the add without changing the validation list. If a new donor is added, the user will be able to input various supplemental information (address or any other notes to the church treasurer). Also, “Cash” should be changed to all upper-case (if not entered that way) prior to printing the reports.

Kevin & Brian: The miscellaneous data must allow the entry of Deposit Bag #, the counters’ names, and the total number of each denomination of currency and coin. It will also verify that a bag number and counters’ names are entered prior to allowing the form to be closed.

Brian: The cross-check will compare several items to ensure the printed report contains accurate information:

The total of the donations entered as “Cash” with the total of the input cash by denomination. This ensures that the total cash counted matches the total donations being marked as cash donations.

The total of all area subtotals with the total donations entered. This ensures that every donation reported has been designated to a specific area.

Count the number of unique donor/check# sets. This ensures that check numbers are accurately reported.

Dondra & Timia: The view/print form will allow the user one last opportunity to verify that all data has been entered correctly and then print the two or three reports required. One report is a list of all donations by donor and area, including the amount and check#/cash for each. It will include a summary of all donation areas with a subtotal of each area and an overall total. It will also list any loose offerings as a separate subtotal, but will include these loose offerings in the overall total. The second report will contain a summary of the first report, which is the area subtotals and overall total. It will also include the denomination breakdown of cash and the deposit bag number. Blanks of the two reports are attached as an Excel spreadsheet, and although the reports are not required to look exactly like these, they must contain the same information. Additionally, if new donors were added, a report should be generated to allow printing of the new donors’ information.

**Division of Labor:**

Kevin will be laying out the GUI, including all forms for data entry, which interfaces with the donor information and the cross-check calculations.

Jedidiah will be writing the donor class, which interfaces with the GUI and the cross-check calculations.

Dondra will be writing the print report class which will interface with the donor and donation classes in order to view and display the Detailed Report, and the Summary Report.

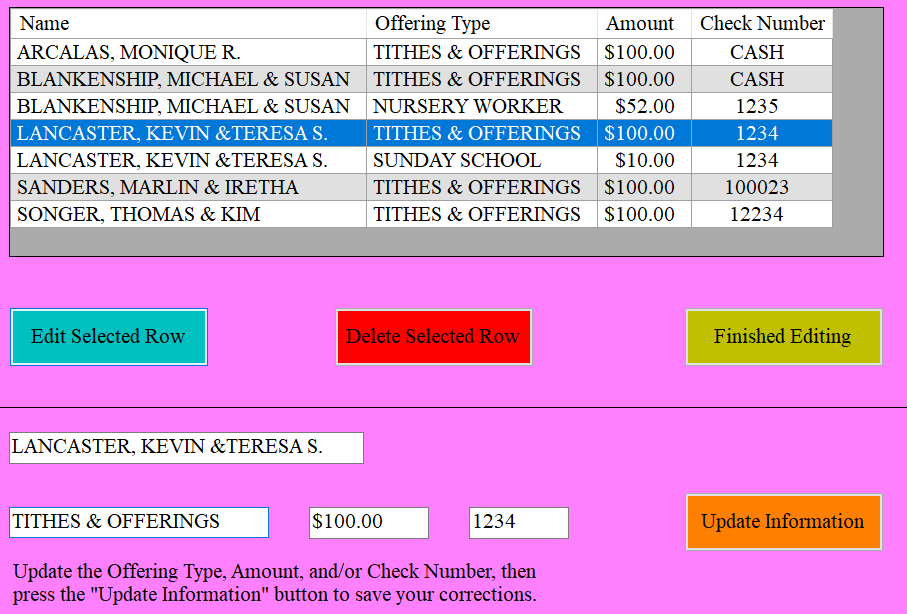
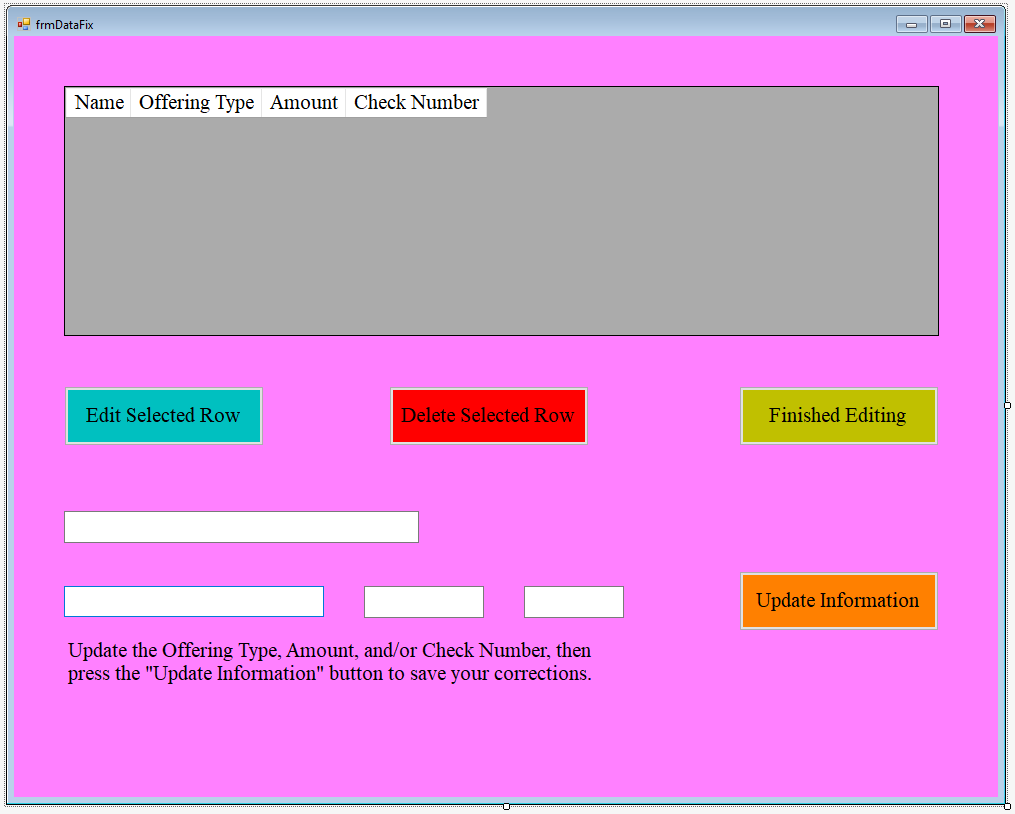
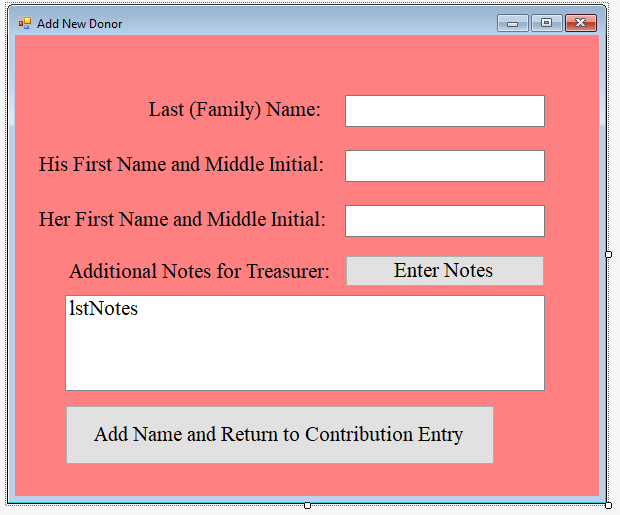
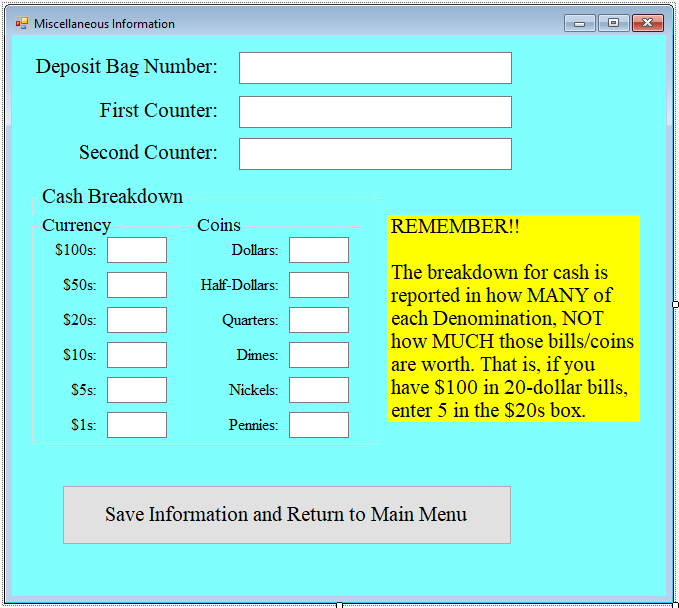
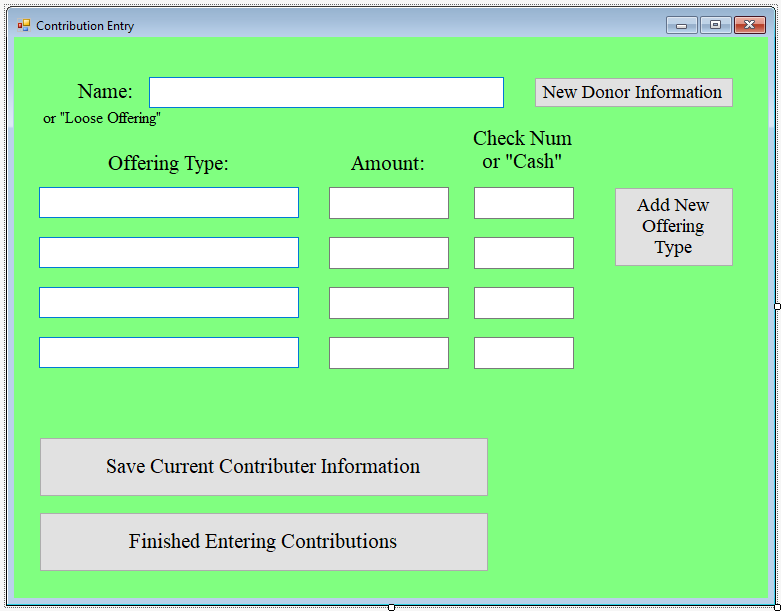
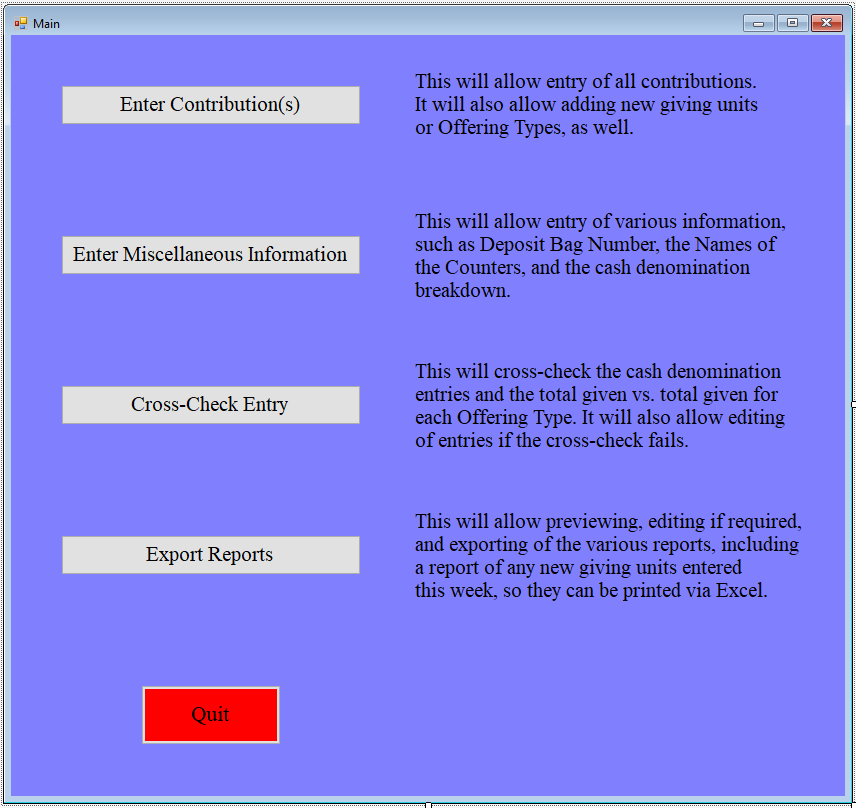
Brian will be taking the donor information and “miscellaneous” information (from the GUI), performing the cross-checks and interfacing with the reports.

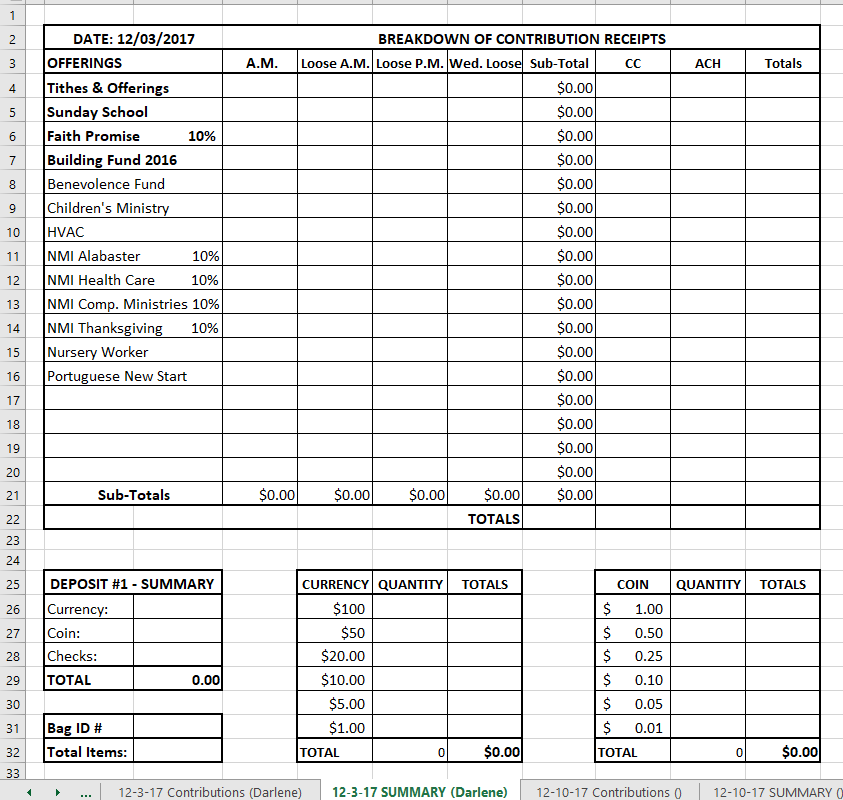
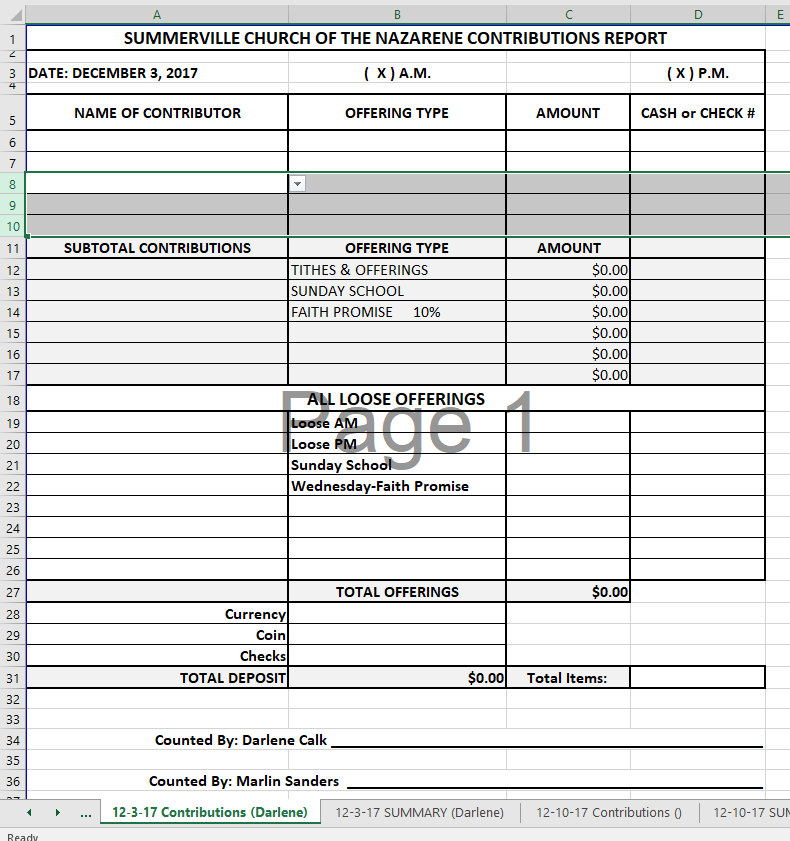
Timia will be creating the view/print class from the GUI which verifies that the data has been entered and will print the reports.

**Attached Pictures & Excel Spreadsheet:**

The picture filenames describe which input form they are. The View&Edit form would list the donors and allow the user to select the line which needs editing (similar to Excel).

The Excel spreadsheet called “DonationsSCN – Reports” is the spreadsheet created to generate the reports from external CSV files. The Excel spreadsheet called “17 DEC” is the spreadsheet that is currently in use by the counters. Both spreadsheets are very similar in layout but differ significantly in the method by which the report gets generated. Pictures of the layouts are below; the spreadsheets are included for reference.



****

**UML Use Case Diagram**

<<extends>>

<<extends>>

<<extends>>

<<uses>>

<<uses>>

<<uses>>

<<uses>>

<<extends>>

<<uses>>

<<uses>>

<<uses>>

<<uses>>

<<uses>>

<<uses>>

<<extends>>

<<extends>>

COUNTER