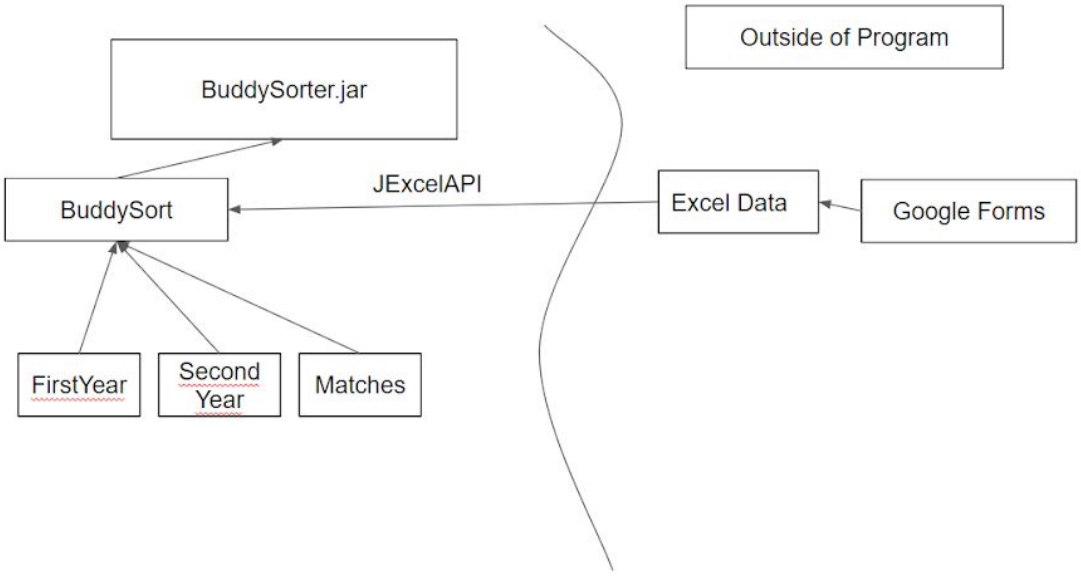
Initial Design:



Input for First Years: (Created using Google Forms -> Excel File):
Needs to get

- First year name
- Region
- Level of Challenges
- an additional Text Field for first Years to comment on specific things that they are uncomfortable with.

Input for Second Years: (Created using Google Forms -> Excel File)
Needs to get

- Second Year email
- Second Year name
- Region
- 10 First Year Buddy Selections

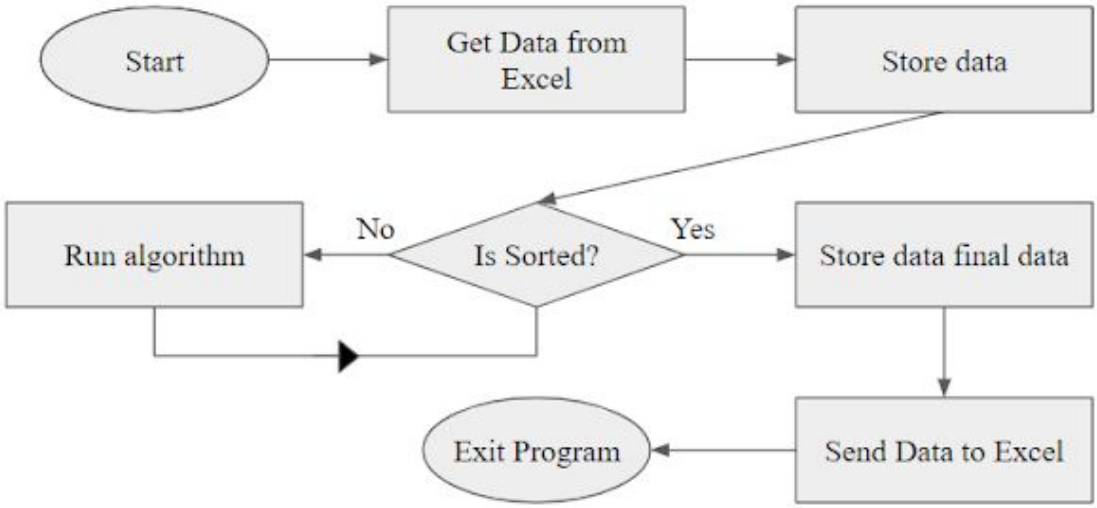
Output: (Excel File)

- File should consist of following

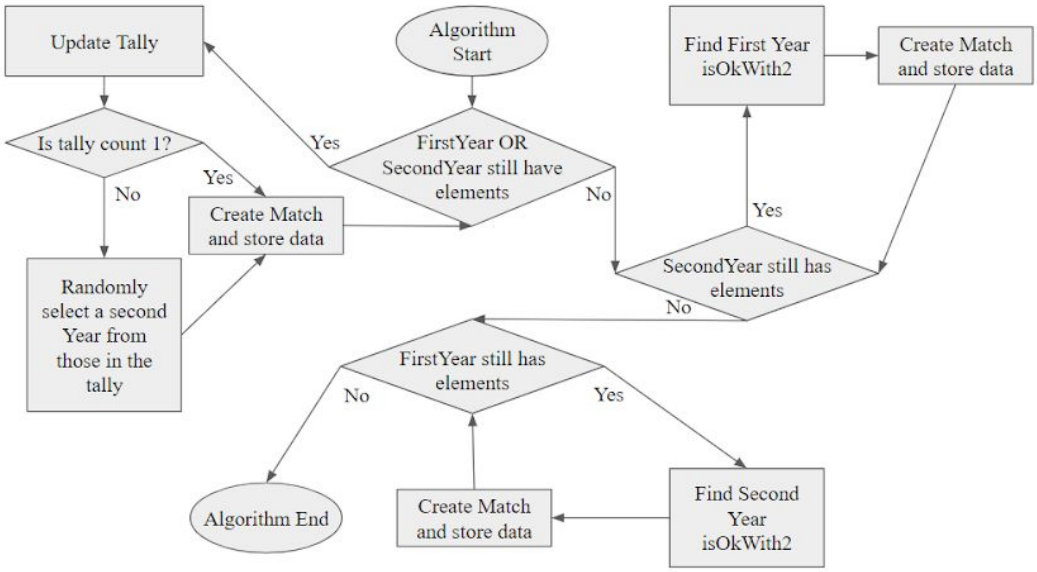
Second year Email	Second Year Name	Second Second Year	Buddy Name	Second Buddy Name	Buddy Challenge Notes	Second Buddy Challenge Notes

- The Second Second Year, Second Buddy Name and Second Buddy Challenge Notes are optional fields that may be filled if the first years and second years set sizes are not the same.
- In the case of the Second Second Years, there should be a different row for each second year. This is so that the row count should always be equal to the number of second years. Additionally this allows for the excel document to be easily prepared for a mail merge sending each second year the appropriate information

Process:



Sorting Algorithm:



Schedule to Complete Project:

Program development will be divided into 3 sections. One section will pertain to the development of the backend code that actually sort through the data. The second section will be the development of the front end google forms, conversion to excel and implementing the JExcel extension. The third section will deal with merging the previous two sections into a functioning program

Front End	Back End
1. Create Google Forms 2. Create Excel File 3. Write Code that reads data from Excel File 4. Write Code that appropriately stores data from Excel File 5. Write Code that writes data to an excel file 6. Test Code for potential errors and add code to prevent errors	1. Create classes for First Years, Second Years and Matches (These classes need to appropriately store the data that is read in by the front end) 2. Write code for the sort which allows Second Years to be assigned a first year buddy 3. Test extensively and add error prevention code.
Merging	
1. Create Interface. 2. Integrate both of the aforementioned parts 3. Test extensively	

Test Plans

Action Plan	Method of Testing
Test Google Forms	Send out the Form to a smaller group of people (~10, this will also provide a sample set of data to test the rest of the program with)
See if data converts into an excel file appropriately	Download the results file and inspect to see if it worked.
Test if program opens correctly	Click on the jar file and see if program opens
Check if buttons work correctly	Click Button and see if appropriate response occurs

Check if data is read in correctly	Print out data during development stage
Check if sort works	See if final data meets the constraints. Create a method that evaluates how well a sorted set meets the constraints
Check if sorted data is stored correctly in program	Print out data during development stage
Check if data is presented correctly in excel file.	See if final data meets the constraints. Compare with stage above.