Our team was given a number of requirements for our system and features that our system would contain. The requirements were:

* Upload a CSV file
* Parse data from a CSV file
* Write sorted students to CSV
* Allow data from the CSV to be viewable
* Display data by Name/Student number
* Allow sorting of the groups manually and randomly
* Allow remain group members to be randomly selected
* The program should be able to terminate

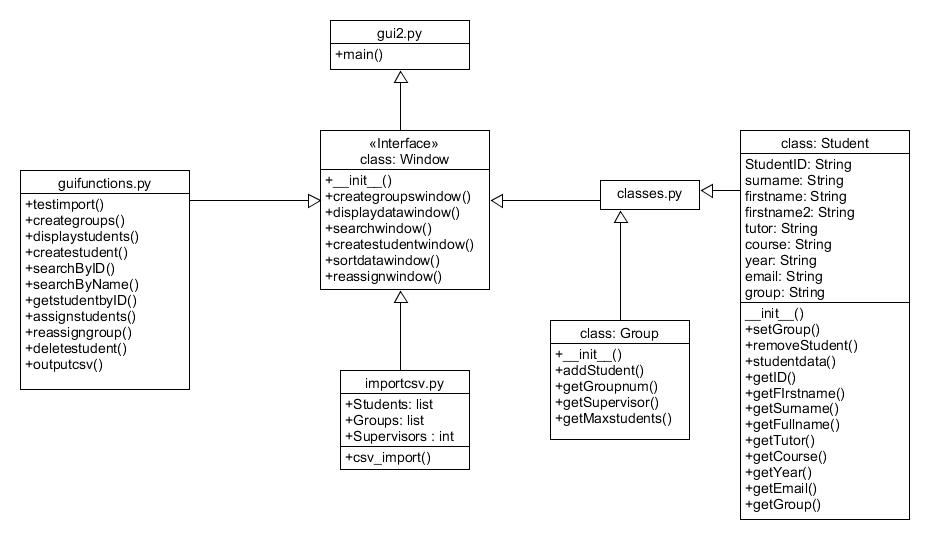
Our team set up an initial meeting so that we could get to know who was on our team and have a look at we were tasked with doing. We split up the requirements between each member of the team; two of us had three requirements to work on and one person had two. Our team was small with only three people which meant there was more to do between each of us but it also meant we could organize meetings easier as there were not that many people to work around when coming up with a day and time. We also set up a group chat which allowed us to keep in contact with each other when we weren’t together and talk about the project.

During our first meeting we decided to use python to code the grouping tool as this language was one that everyone in the team could use and were familiar with. When coding the system, we ended up all working together as working separately was not working very well and was causing more problems and delays. It was when we came together we realized Josh was the strongest coder out of the three of us so he took the lead with the coding and how we were going to go about doing it.

We created the system as a pop up window which launched when the code for it was run. This main pop up window contained buttons which carried out different actions which would result in some other pop up windows coming up. The code to import the csv file through the main menu caused some delay as we weren’t sure how we were going to go about importing it and whether the code to do this would have a hard coded file name in so the system new which file to look for when someone wants to upload a file for use. We created “csv\_import(filenamestudents, header=True)” to import the file and had its code file linked to the main menu file. In the end we didn’t hard code in specific file names.

We were able to start straight away and Josh was given the lead with the coding as he had done something similar before was had an idea of how we could go about creating the tool. He was able to look at what he’d done previously that was similar to what we had to do and get an idea for the code he would have to create. Because of this, many of the smaller features were easy to make and didn’t take much of our team’s time up. Within a few days we had some kind of we were able to use and get an idea for how the rest of the requirements would be implemented.

Our UML diagram for the grouping tool:



When you import a csv file, the system reads in the data it contains and then you can view this using one of the functions on the main menu which will present it in a separate pop up window. You are asked how many groups you want to create when running the create groups function and then to see if these groups have been created, in the command line window which runs alongside the main menu, the groups locations get printed which lets you know they have been created.

The main menu contains a button which allows you to search for students that have been imported in the csv file. You are able to search for the students either by their name or their ID number. The search feature is case sensitive so if you don’t put a capital letter at the beginning of someone’s name when there is one, the results will come back that no student has been found with that name. The delete, randomly assign and manually assign functions are kept within the same branched off pop up menus.

Our main issues were to do with coding and figuring out how we would go about coding the different part of the system. During the team meetings we discussed ideas on how we were going to tackle the different features and the coding they would require. The main feature we struggled with was creating the groups which the students would then be put into either manually or randomly. We had difficulty finding a way for a button to create a number of groups and then have those groups be fillable with students. Our first attempt at this saw the groups being created but we weren’t able to fill them as the auto assign function worked but for some reason was only filling the first group and would never fill the groups after number one. Josh looked into this and was able to get around this by creating a variable to hold the count for how many groups have been created and then putting a maximum number on each group so that one group couldn’t hold all the students so they would then be shared out over the other groups.

The function which deletes a selected student comes up with an error whenever it is used but the function still carries out what it’s meant to do with no other problems. We have looked into why the error continues to come up but whenever we play around with it we end up causing more problems and having the whole function not work. We have thought about creating the function again when we have time but as there is no negative impact from the error we have left the code the way it is.

We currently have an issue with moving the tool from a pop up box menu to an online based tool and then linking our system with the one that group one created. Sean from group one joined our group as he has an understand of how group ones system works and he’s been able to give input into how we can move ours from a pop up menu to an online menu. We are currently in the process of this transition and Sean and Josh have taken the lead with it as they’ve discussed and understand a way they can do it which the other members of the team are happy to let them get on with.

My team worked well together during the process of creating the grouping tool which meets all of the requirements that were set out. We were able to start straight away and from our meetings we have been able to sort out issues early on which meant we didn’t delay when it came to having the tool working for our presentation. Even though we had some issues when making the system, we were able to work together to solves these problems and meet what was needed. Now that we have another member we will be able to progress further and can now focus on moving the tool from a pop up menu to an online usable tool. One thing I would change is when we started off created the system, we should have made it online as this would have saved us time now as we wouldn’t need to do the transition.