Virtual Keyboard development notes

Ver 1.0 – Create three class files in addition to mainProgram – Key, Keyboard and Keyboard manager. Also a class called texture, a single instance of which will be used to texture all of the keys on the keyboard. Partially Complete.

Ver 1.1 – Rendering of a single key now functional, load in the other keys to be rendered and create button pressing functionality. Partially functional – keyboard rendering with all keys.

Ver 1.2 – Implement key pressing functionality, re-arrange the Keyboard class a bit so that it incorporates functionality required to change the position of the keys according to their useage. Partially completed – button presses detected, however coordinates wrong as it is the coordinates of the monitor NOT the SFML window.

Ver 1.3 – Move main function functionality into the Keyboard class so that Keyboard functions have access to the window as a variable – necessary for window coordinates being passed into the key press functions. Complete – pointer now inputs the window coordinates of the pointer as opposed to the screen coordinates.

V1.4 - Resolved the error Where all keys were were being detected as being pressed, now only the key that is actually pressed is detected as having been pressed - major milestone reached.

V1.5 - Adequately write out the letters and frequency of their occurance to a text file - this will be used to reload the keyboard in its new form the next time the program is started up. Also program the read function to read in the data saved to the text file to initialise the new keyboard. Added button class to project to detect when to write out the keys and their popularity to a file. First Part complete – writing out letters and their frequency to a text file.

V1.6 – Implement read function to read in the letters and their frequency to a text file. Read function almost functional program will read in characters but there is an issue with numbers appearing in front of letters in the read function. So the program dows not find the correct index. Sort function does no work properly as the indices are not being read properly in the read function - so the keys are not being sorted into descending order of precedence and hence do not appear at the correct position on screen.

V1.7 - Resolve above outstanding errors. Errors with reading from the file resolved – Keyboard now optimises as it should.