**Task 2 Evaluation – Tom H**

**Success Criteria:**

The program must be made in Visual Basic as it is a reliable programming language and it must include a user friendly design:

Yes, my program was made in visual basic and has been coded efficiently so that is easily readable in case I needed to make an edit to it and complex enough to work flawlessly. The design looks stylish and the colour complements the design well. The buttons and list box are clearly displayed for the user to see and are the right size for any screen size. The design can be seen in my development document as I have taken screen shots of the code and program design.

Be able to save a file that contains text, which has been entered by the user from an input box:

My program shows a save file dialog box for the user to enter a file name and will then display an input box that will allow you to enter a sentence. The use of an input box is much better than a textbox because it saves up space on the form.

The program must be able to find all the unique words in the text and locate their original position by putting them in a list:

The program can do the successfully and well by building up the information in an array and the joining this together at the end to form a string which will be outputted to the list box. My testing shows this through the screenshots of 2 – A.

The program must be able to replace the words in the sentence with the positions that will be linked to the unique words:

The program can loop through the array of unique words that I built up and will take this position and put it in a separate array called unique positions. Like the unique words, I join this as a string and save in to a list box. The screen shot of this can be seen in my formal testing under figure 2 – D.

The program must store all this information within a single text file:

As you can see in screenshot 2 – G my program takes the two joined together arrays that were outputted to the array and then saves them into a file. The user saves a file, before the compression process, which stores the sentence, unique words and unique positions. This can also be seen in figure 1 – A.

**Summary:**

In my opinion, the code and design for my program are very efficient as I took time to make is simple and readable for the user. The process of saving and building the two arrays is quick and simple whilst using code that is easy to read for any amendments I had to make during this task. Whereas I initially started with two buttons I managed to simplify it down to just one. I avoided taking up space by using input boxes instead of textboxes and a menu strip also.

If I was to improve my program I would definitely spend more time on perfecting my pseudocode and flow chart. To begin with, I had no idea how I was going to approach the task and struggled to get my head around the logic. When I did get it, I found myself rushing against the clock to get it all done well before I started task 3. Overall, I think I am pleased with my finished result in terms of how polished it is.