

Justification of the proposed solution

There are many different methods available to me to create a solution for Greenparks schools problem. The first and most obvious method available is to use a programming language, such as visual basic. This programming language provides numerous options, and has a wide range of commands that would be useful for this project. However, this programming language is not as flexible as a programming language such as C or C++. It provides a constricting although simple user interface creator, and does not have the wide range of commands available in C. Also visual basic is more abstracted from the machine code, and is therefore less efficient. However, it is far easier and fast to programme in visual basic, as it is closer to human language. Also, I have done a lot of programming in visual basic, and so I am more familiar with visual basic code. I have programmed multiple things using visual basic and have interfaced with data structures before, and thus, for this project, using visual basic may be a good idea.

However, I could, instead of using visual basic use HTML. This would mean that I could create a web based system that could be accessed over the internet. This would mean that pupils and staff could access the system from home. However, using a web based system would require access to the internet, of both the pupils and the staff. Some people do not have access to the internet, and this type of system would discriminate against them. Also, HTML is rather limited in its available functions, when compared to a programming language such as visual basic.

I could use flash action script to create a solution to the schools problem. This could look very nice, and it would be very easy to create an aesthetically pleasing user interface. Flash action script can also interface with a database, and thus allows for different logins and the storage of points totals. However, the system requirements for running flash exceed those of running a basic visual basic application due to the additional graphical elements. Also, flash is not designed for a task such as this and thus lacks some basic functions that visual basic has (such as interfacing with data structures, and form design). Thus implementing the proposed system using flash action script would be very difficult, if not impossible.

The final alternative I have considered is using a spreadsheet programme. A spreadsheet programme can easily handle formulae and equations to produce a result. However, spreadsheets are not designed for the more complex decision making skills requires by the project, such as repeating loops, and interfacing with external data structures. Because of the lack of interfacing with external data structures, if I used this method as a solution, it would be unsecured and easy to hack. Also, the creation the login system itself would be difficult.

Overall, I have decided that visual basic is the best choice for this project. I have a lot of experience working with visual basic, and it provides the complex commands necessary for a project of this type. This includes decision making and loop functions, along with repeatable timers (which provide for an easy method to constantly refresh information). In addition, visual basic provides an intuitive UI designer. While this may be slightly limiting, it is very versatile, and for a project like this, I believe it will be very effective at achieving my goals. It provides an easy way
to implement simple things like buttons, text boxes, massage boxes, and various windows (called forms). Also, visual basic can easily interface with other data structures, which is ideal for storing login information, point’s totals, and comments. I have used visual basic to interface with data structures before, such as Microsoft access, ideal for storing complex 2 dimensional arrays, which will be used commonly in this project. Also, interfacing with text files is even easier than access, which could be used for storing large amounts of text. Visual basic provides inbuilt temporary data structures, for easy manipulation, called arrays. These are very useful. Finally, visual basic has in built error detection, for easy management of the project identifying syntax errors for me, and showing me where any execution errors are when the project is being tested. Because of these reasons, I believe visual basic is the optimum choice for creating this project.