chali tillakaratne

Final Project Report

***Context Page:***

# Designing and Developing a Software Solution:

## Log Book:

## Defining and Understanding the Problem:

## Planning and Designing of a Solution:

## Implementation of a Software Solution:

## Testing and Evaluating Software Solutions:

# Logbook:

21/05/2018: @ 5.30-6.35pm

Today I set up my GITHUB online and my GITHUB Desktop. I have learnt that GITHUB Desktop does not work on the school's internet therefore I have decided to make an offline version of my GIT Repository and upload/ Update that when I get home. I have also made a backup on the school's Network Drive where I will Update my Repository whenever I use the school computers.

We also looked at a possible solution to this problem and have decided to work my Gantt Chart and outline my Initial Report. I have also created the Project Diary.

22/05/2018: @ 8.37pm

Today I worked on refining my Gantt chart and Creating the report Context Page. I also worked on what the GUI will look like and planning the layout of my forms.

25/05/2018: @ 8.40-10.27am

Today I worked on my practical part of the project. I developed the different pages involved in the program. I developed the Splash screen which houses two buttons that send you to either the teacher tab or the student tab. I made it so that each page allows access to the Help Page and access to other pages. I am currently implementing a system to be deployed on the teacher page which allows the teacher to be able to open and save text files which contain a short extract to be Edited in the Cloze Generator. I am also currently implementing a system which allows the student to be able to open the file online and complete it, at the end of which they would be able to print off the completed text with the possibility of getting a certificate with their mark. I would also want the teacher to be able to print off the newly made Cloze tests via the Program.

26/05/2018: @ 7.45-9pm

Today I started Development of the Login Page and the Create New User page. I have finished designing them and I have all so figured out how I want the log in too work. For logging in the teacher will need to create a new user. The Username will be the name of a text file and the password will be the only word in the text file. When the user logs in they will enter their username and password. The program will then search a database of usernames to find a matching one, if found the program with check the password against the password in the text file. If the Passwords do not match, then the teacher will have to either create a new username or password. Alternatively, there is the possibility I could have a Hard-Set password that cannot be changed, and the teacher will know this password, but the student will not. If the teacher forgets the password, then the teacher should follow instructions on how to find the password again.

28/05/2018: @ 5.45-7.30pm

Today I finished by Login page, I ended up using a static login so currently the password and username cannot be changed. Eventually I want to be able to have the user create their own password and username. I also created the systems in which the saving would occur. The first system automatically reads everything from the text file into a text file this method uses the Stream Writer method. The second method was implemented when the user clicked save as. This would open a file dialogue to allow the user to save the text as a file with any name they so choose.

30/05/2018: @ 8.40-11.40am

Today I refined my database and removed the automatic ID generator that was giving me troubles when searching for entries. I replaced it with a pre-set Username and Password that is less efficient but still works as directed. If I had more time I would be able to fully develop a login system using Microsoft Access, but I do not have the time requirements to do so which is why I chose to use the faster method of logins using static variables, if the user types the correct username name and password corresponding to two variables they will be given access to the form otherwise they will not be allowed access. The problem with using static usernames and passwords is that I have no way of encrypting them unless I put them into a textbox and encrypt the text box, but I do not understand how to do that, nor do I find it crucial to do so in the period I had been given to complete the project.

08/06/2018: @ 8.40-10.27am

Today I worked on printing the cloze text and I have ended up with three different methods of printing.

The first method of printing was done so using a print dialog and a print document

The document declared that the textbox was to be printed and the dialogue allows the user to select the printer

13/06/2018: @ 8.40am-12.33pm

Today I worked on printing the form. I ran through a few methods of printing including the print form and the print document. Both solutions had major complications and were not complete. The print document is unable to draw the dimensions of the text when printing, this means then when printing to a pdf or any page the words will be read left to right onto the page until it runs out of space on that line and then it stops adding words.

The print form method creates a screenshot of the form exactly how it is and displays it in a print preview and from there I do not know how to print the form.

I have decided to leave this until the end after I have finished the rest of my project.

16/06/2018: @ 7.45-11pm

I have figured out the bugs in my printing system and have reworked my setup to be more efficient in the way it works. Now when a user wishes to print a form it will take a snapshot of the form screen and then display it onto a print preview page. From here the user can select to zoom in or print the form to the default printer. I have still not found a way to love my problem of printing to pdf or printing via the print dialogue and the print document. I have decided to leave this till very last.

17/06/2018: @ 12.34pm -9.40pm

Today I have started compiling all my information into the corresponding documents to be handed in to class. I have created the Initial report using the information I created before I started the project, this includes a pre-project Gantt chart and screen designs. I am also planning to have a post-project Gantt Chart and to compare them to see where I miss calculated or issues I have missed. During my initial planning phase.

18/06/2018: @

I have also found a major problem with my program in which the program cannot be opened without the visual basic power pack 3 installed. If you open the form that requires the "form.print" such as the student and the generator they are unopenable without the powerpack as the print form is part of the power pack. I am not certain if this error will continue after the program has been compiled.

19/06/2018: @

20/06/2018: @

# **[Defining and Understanding the Problem:](#_Designing_and_Developing_1)**

## My Solution:

### The needs of the client (teachers and students):

The clients and users of the software have the right to

#### Efficiency:

The solution functions by using the computer’s resources effectively. The program uses modules, well-designed algorithms, and structured code. The program has been written in VB.net to help insure that the GUI is clean and effective.

#### Functionality:

The program functions as intended without major bugs, glitches, or errors

#### Compatibility:

The program is compatible with any device that is supported by Visual Studio 2012 and Vb.net

#### Performance:

The program performs effectively without any documented or foreseen errors occurring.

### Social and Ethical Issues:

The software follows high social and ethical standards, it is usable by anyone on a pc freely. There is no malicious or illegal function imbedded in the software. The program included links to websites where code was sourced from and documents that this code was not originally written.

### Specifications or requirements definition:

* Produced on a PC
* Follows good ethical practices
* Solution functions as described
* Full system documentation included
* limited user documentation is included
* Project management tools used where appropriate

## Documentation required to model the system:

-Internal documentation of the system is used to increase the ease of maintainability and to decrease CPU usage. This includes documentation for the modules, how they work and what they do.

-External documentation of Modules, System Diagram, Screen design and algorithms is included as to increase maintainability and error checking

## Quality Assurance:

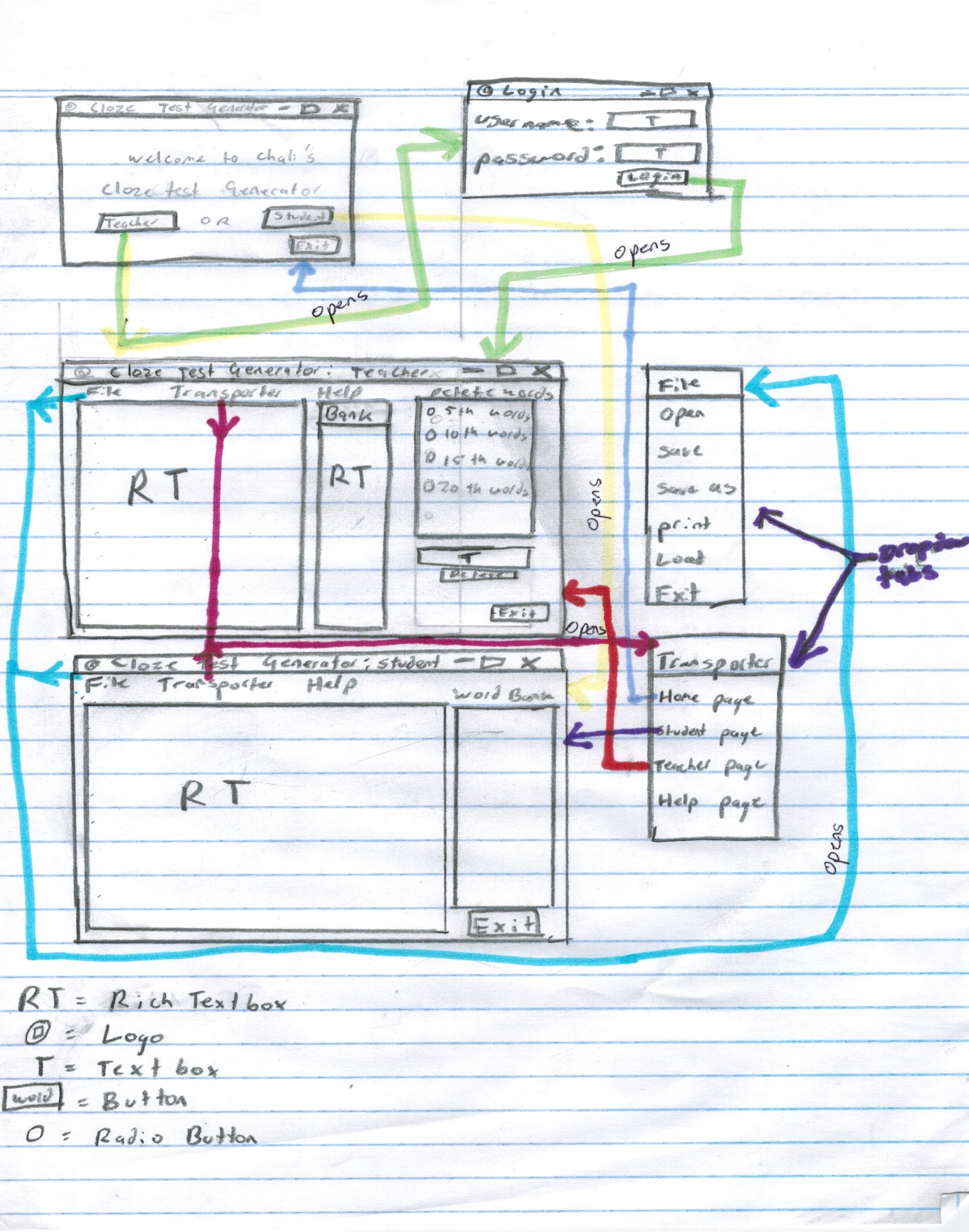
-The software is reliable and will not crash mid-ways through the generation of a cloze test

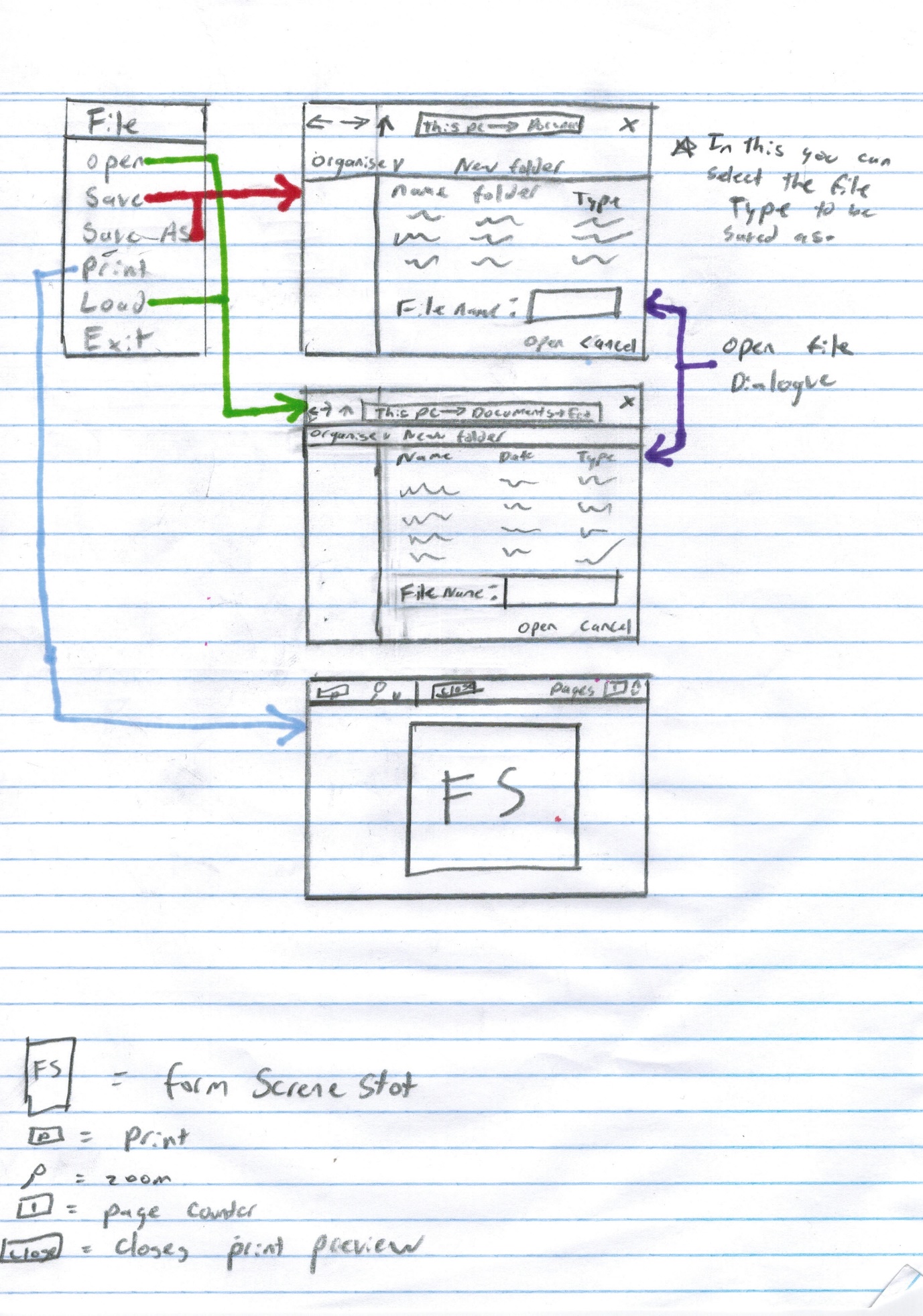
-The software has high maintainability by the use of external and internal documentation of modules and algorithms

-The software is free to be used by any user

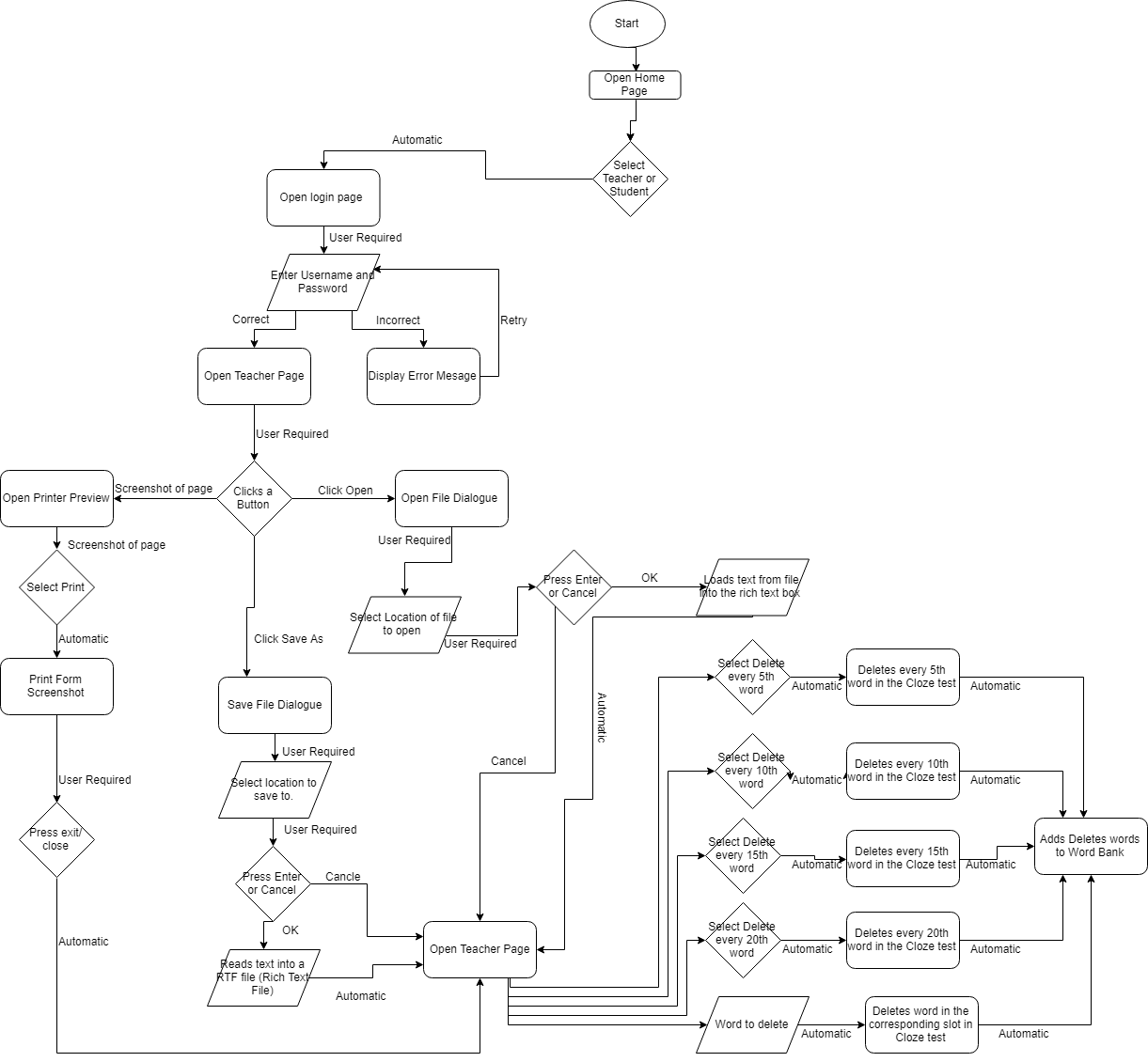
-The software is easily portable and can be run on any PC supporting VB.net and Visual Studio 2012

# Planning and Designing Software Solutions:

Story Board:



## Flow Chart:



## Testing Algorithms:

Test any algorithms with appropriate test data. Detail why you have included each of the test data items.

## Standard modules or OS Routines:

## Choice of Programming Language:

The programming language that has been selected for use during the creation of this software solution is VB.Net via Visual Studio. The reason that vb.net has been selected is due to flexibility of the language and the fact that the GUI is premade. The programming language in Visual Studio allows for further customisation and makes it much easier when it comes to programming as it runs a debugger to find bugs and it will point out where it is and possible solutions to fix it. The language is also very straight forward and easy to understand. The programming language contains many functions that allow large amounts of code to be progenerated when implemented in the solution.

## Sorts or Searches used:

|  |  |  |
| --- | --- | --- |
| Searches: | Where was it used? | How was it used? |
| Binary Search |  |  |
| Linear Search |  |  |

|  |  |  |
| --- | --- | --- |
| Sorts: | Where was it used? | How was it used? |
| Bubble Sort |  |  |
| Insertion Sort |  |  |
| Selection Sort |  |  |

## Data Types and Structures used:

|  |  |  |
| --- | --- | --- |
| Data Types | Where was it used? | How was it used? |
| Integers |  |  |
| Booleans |  |  |
| Floating Point |  |  |
| Strings |  |  |

|  |  |  |
| --- | --- | --- |
| Data Structures: | Where was it used? | How was it used? |
| Array |  |  |
| Record |  |  |

# Implementing Software Solutions:

5. Code/develop the solution using an appropriate design method. A choice of language, development approach and user interfaces should be justified.

6. Write an appropriate user manual with some online help that assists program users.

7. Include any other documentation you see as relevant for this task. But must include all of the modelling tools and test data previously mentioned.

# Testing and evaluating Software Solutions:

8. Test the completed solutions with appropriate test data using a range of testing methods. Desk check and use of test data in your program