**Login**

Username:

Password:

\*\*\*\*\*\*\*\*\*

Login

Sign Up

**Login Screen UI Design**

**X**

# incorrect UN or Pass

User name input field: allow for validation of the user.

‘txtUsername’

Password input field to ensure only the user can gain access to their info.

‘txtPasswd’

Notify the user if the UN or Password is incorrect/ doesn’t exist

‘lblErrorMsg’

Login Button: compares inputted data to that of which is in the DB

‘btnLogin’

Sign up button, allows the user to add their account to the database and store their data

‘btnSignUp’

Input:

The Login Form requires that the user inputs their username and password for validation in the Database when the login button is clicked. The username will be checked against those within the database login table – ‘tblLogin’. The password which is linked to this username will be compared against the password entered into the password field on the login form, after the entered password has been encrypted (simple encryption). If the password then matches, the user is validated and signed into the application, where they can then see their information on their progress.

When the sign up button is pressed the system will take the data from within the username and password fields and add this data to the database, creating an account for the new user. The system will ensure that the fields are not empty and contain valid characters (e.g. no spaces in UN), the system will next add the username to the database and then encrypt the password and add that to the database, the program will then continue by automatically logging in the new user.

Output:

If the password entered is wrong for that account or the account does not exist when the login button is pressed; the program will display an error message in the error label – ‘lblErrorMsg’ saying “Incorrect username or password”

If the username is already in use or contains invalid characters when the Sign Up button is pressed the program will display the error message in the Error label saying “Invalid Username”

**Home Screen UI Design**

Home Screen

**\_**

**X**

**\_**

Answer Questions

Modify Questions

My Progress

Welcome <username> to <application name>, Select what you would like to do.

Answer Questions button allows the user to answer questions from the DB, and select which questions to do from various categories.

Modify Questions button, takes the user to the modify questions form where they can edit each individual topic of questions.

This button will take the user to the form which will display their progress and will also show them their progress over time and predictions on their progress.

Logout

Logout button allows the user to sign out of their account

The user’s username will be displayed here – ‘lblWelcomeMsg’ & ‘userName’

Input:

The home screen will receive two types of input, clicks and data from the login screen. There are various buttons around the home screen which will take the user to other pages/forms, these include: ‘Answer Questions’, ‘Modify Questions’, ‘My Progress’, and the ‘Logout’ buttons. The ‘Logout’ button will take the user back to the login screen and clear any used variables. The Home form will take information from the login screen, and use that (‘userName’), it will be used in the welcome message (‘lblWelcomeMsg’).

Outputs:

Depending upon the button clicked the home screen will take the user to one of ~4 different forms. Other outputs include things such as the welcome message being displayed on the home screen including the user’s ‘userName’.

**My Progress Screen Design**

**My Progress**

**🡨Go Back**

This page displays your current progress…

<Information\_goes\_here>

This is the label/s which will present the user’s information to the user so that they can see their progress, and how they are doing.

Graph to show the user’s progress over time and then predict where the user will be after a length of time of using the program.

Inputs:

There are very few inputs on this screen, as this page is mainly just to show the user their progress and how they are doing over time. The only input on this page from the user will be the ‘btnBack’ or the ‘Go Back’ Button. This form will however take the data from the database and use that within the same form.

Outputs:

There are a few outputs on this form, these include the user’s data being presented to them in a graphical form, as text in labels and a graph which will show the user their progress over time and where they are predicted to be after a certain amount of uses.

**Question Modification Screen**

**Question Management**

**🡨Go Back**

Modify Questions

Create Questions

Manage your questions here…

‘Modify Questions’ Button – allows the user to go to the page to edit the questions that they have already made and are stored in the database.

‘btnMod’

This Button will allow the user to add questions to the database

‘btnCreate’

Inputs:

This page only has 3 inputs which are all buttons, the go back button; which is the same as before, there is also the ‘Modify Questions’ and ‘Create Questions’, which will take the user to the corresponding pages.

Outputs:

This page will take the user to one of 3 possible forms as mentioned above.

Inputs:

**Create Questions Screen**

**Create Questions**

🡨 **Go Back**

Next

Subject

Topic

Question

Answer

The ‘Create Questions’ screen will allow the user to type in a subject, topic, question, and answer for every question which will then be stored in the database. The next button will add the previous question to the DB, and then will clear the page and allow for another question to be created while the previous one is being processed and added to the database.

Output:

The inputs from the textboxes will be stored as variables and then the textboxes will be cleared when the next button is pressed, this data will then we processed (while the user is adding the next question to improve efficiency), and placed into the database question tables. If the next button is pressed without anything in the textboxes the system will close the creation window and save all of the previous questions.

Inputs:

**Modify Questions Screen**

Subject

Topic

\/

\/

Two possible layouts for the textboxes to be in.

Dropdown menus to select which questions to edit

The questions can be modified using textboxes on this page, the topic can be selected then a list of all of the questions from that topic will be presented (extracted from the database) to the user which they can edit, and then save these changes to the database using the save button - 'btnSave'.

Outputs:

The outputs will be the saving of the changes to the database, and the outputs will be taken from the textboxes and then put back into the database.

Inputs:

**Answer Questions Screen**

**Test!**

**🡨Go Back**

**1.**

<Question\_Goes\_Here>

<Answer\_Question\_Here>

🡪

A\_N\_S\_\*\_E\_R\_/\_G\_O\_\*\_S\_/\_H\_\*\_\*\_E\_.

Time Remaining: 0:00

<Previous\_Answers\_Are\_Displayed\_Here>

<Previous\_Answers\_Are\_Displayed\_Here>

<Previous\_Answers\_Are\_Displayed\_Here>

<Previous\_Answers\_Are\_Displayed\_Here>

-

The Answer Questions Form will take inputs from a button and a textbox, the textbox is where the user would type on their answer to the question and then press the enter key or the ok button which will send the answer to the system.

Outputs:

The system will display the question, and hidden version if the answer, it will also display the time remaining to answer the question, and the user's previous answers. It will also use these previous answers to display the letters which the user has already got correct in the correct position. The system will also take an output from the timer which will reveal a random letter every five seconds.

Inputs:

**Select Questions Screen**

Random 20 Questions from DB

20 Questions from Subject

Choose Topics (Max 20)

Subjects

\/

**Start**

Subjects

\/

/\

\/

-

-

This form will contain a checkbox and dropdown menu lime systems to allow the user to choose a which questions they would like to answer from a choice of topic/s, random all questions from a subject, or Random all subjects. The form will also take inputs from the database to display the topics, and subjects which the user can pick between; this data must remain up to date and contain a list of all of the subjects, and topics. There will also be a button which will finalise the choice and open the next form: the answering page.

Outputs:

The start button on the page will setup the questions for answering and will open the next form. The other outputs will be from the checkboxes and dropdown boxes witch will tell the system which will category of questions will be answered by the user and which ones to setup.

Inputs:

**Score Screen**

**Your Results**

Score: <\_\_\_\_\_\_\_\_\_\_\_\_>

Percentage: <\_\_\_\_\_\_%>

**Done**

the button 'btnScoreDone' will be the only user input on this screen. The other inputs will come from the other screens and the user's answers to those questions, the results will be calculated.

Outputs:

The button 'btnScoreDone' will close this form and the previous form then take the user back to the home form. Other outputs include the presenting of the score to the user from variables.