|  |
| --- |
|  |
| Geometry Quiz |
| Software Systems Development AS Unit 2 Coursework |
|  |
| **Adam McNeill** |
| **Candidate 1099** |

|  |
| --- |
| [Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.] |

Contents

[Introduction 3](#_Toc449342552)

[User Requirements: 4](#_Toc449342553)

[Initial Plans 6](#_Toc449342554)

[Consultation with the Client 14](#_Toc449342555)

[Detailed Storyboards 15](#_Toc449342556)

[MasterForm 15](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342557)

[Loading Screen 18](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342558)

[Splash Screen 20](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342559)

[Log In 22](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342560)

[Sign Up 24](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342561)

[Main Menu 26](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342562)

[Question 1 28](file:///N:\A-Level\AS\Software%20Systems\AS%20Coursework\SSD%20REPORT.docx#_Toc449342563)

[Question 2 30](#_Toc449342564)

[Question 3 32](#_Toc449342565)

[Question 4 35](#_Toc449342566)

[Question 5 37](#_Toc449342567)

[Results Screen 39](#_Toc449342568)

[High Scores 41](#_Toc449342570)

[Test Plan 43](#_Toc449342571)

[Testing 45](#_Toc449342572)

[frmMainMenu 46](#_Toc449342573)

[frmQuestion1 46](#_Toc449342574)

[frmQuestion2 47](#_Toc449342575)

[frmQuestion3 47](#_Toc449342576)

[frmQuestion4 48](#_Toc449342577)

[frmQuestion5 48](#_Toc449342578)

[frmResultScreen 48](#_Toc449342579)

[frmHighScores 49](#_Toc449342580)

[Test Evidence 50](#_Toc449342581)

[EVALUATION 84](#_Toc449342582)

[Beta Testing and Response to Feedback 89](#_Toc449342583)

[Appendix: 91](#_Toc449342584)

# Introduction

The principal of my former primary school has contacted me to tell me that the school is installing a new computer suite in order to introduce a higher level of digital learning within the school. Whilst the computers themselves enable digital learning to take place, the computers need educational software in order to teach the pupils, this is why my principal contacted me; he has asked that I produce an educational and interactive application for the pupils in the form of a quiz game, to teach the topic of basic geometry to prepare pupils for their end of year in-house examinations. As there is a limited amount of computers, the client has asked for the quiz to be timed so that each student has a maximum of 30 seconds to complete each question, he has also asked that the quiz cover topics from quadrilaterals, triangles, and their properties.

The client would like the quiz to be as intuitive and robust as possible, and has insisted that at least one of the questions is randomised somehow to create a new user experience each time the game is played; additionally he would like a unique response system for each question, including text fields, drag and drop, true/false questions and selecting an answer from range of choices in a combo-box. The client also insists on the inclusion of an account system wherein a pupil can log in to their account when playing the game as a means to track their progress, which will be reflected in their ‘score’; another function that the client has requested. Moreover, as this is an educational quiz, the client has requested that an extensive help function be included that is accessible from every page.

In order to give pupils ample time to complete the quiz for revisionary purposes, the client has asked that the quiz be completed by May 1st

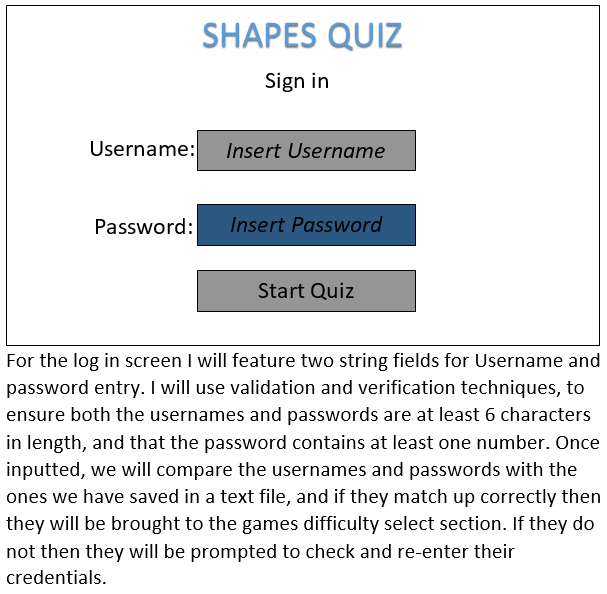
# User Requirements:

* Create an educational quiz game
* Must be playable from a computer running windows 7
* It must teach the topic of Geometry
  + Focus on 2D shapes
  + The properties of quadrilateral and triangular shapes
* It must be intuitive enough for 7-11 year olds to be able to play it unsupervised
  + Help screen accessible from every other screen in the game
  + Tool tips included to explain what intractable properties do
  + Feedback after questions
* It must have sign up/log in functions with user credentials held in a file
* Navigation should be intuitive; appropriately named buttons, exhaustive use of a menu strip.
* Ensure every form bar the loading screen can be accessed from any other screen without needing to reload the game.
* Must feature at least 5 questions
* Each question must have a unique response type
  + True/False
  + Text Input
  + Button input
  + Selecting options from a drop-down combo box
  + Drag/drop
* The application must be completed by May 1st
* Randomisation used at least once to create a new user experience each time the game is played
* A scoring system to track the pupils progress
  + A high score table that compares high scores and shows the names of the best players
  + Scores saved in a file
* Help available on every usable form
* Ability to restart the quiz any time
* To be able to exit the quiz at any point during gameplay
* Menu Strip and toolbar items consistent to include:
  + Logging out
  + Restarting the quiz
  + Exiting the quiz
  + Viewing the high scores
  + Displaying help

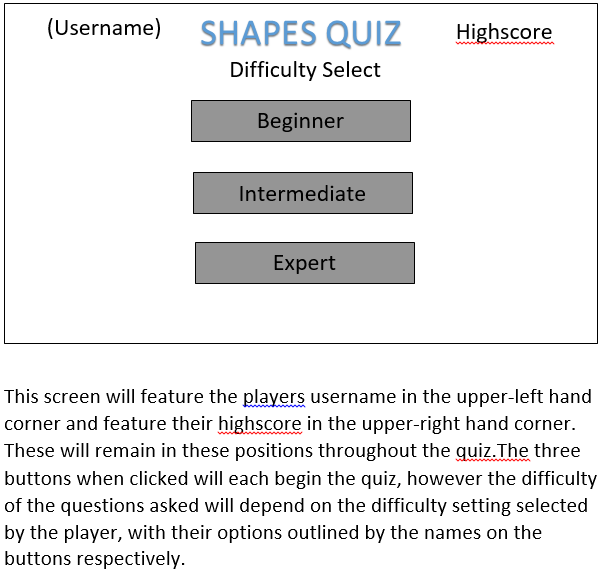
# Initial Plans

Colourful/relevant to theme background

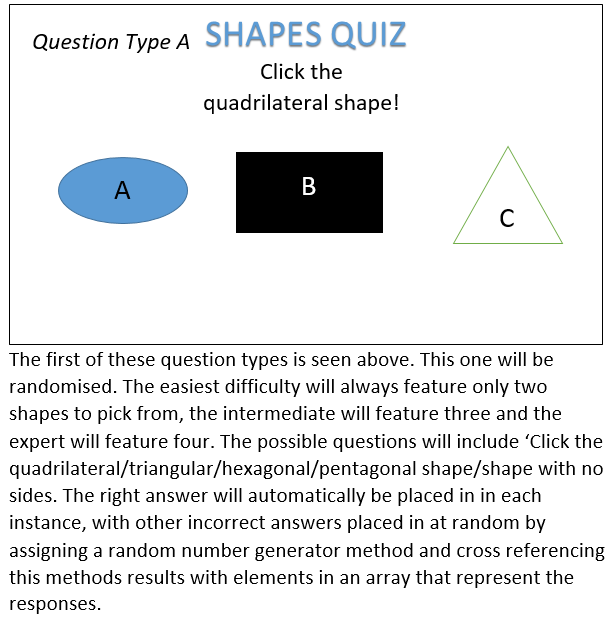
Create A sign up button/form

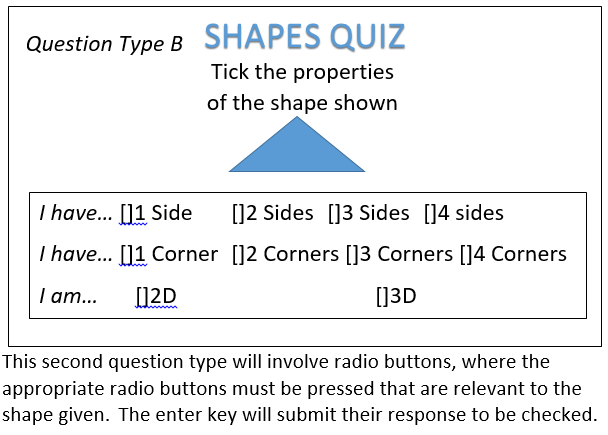


Menu Strip; create a menu strip as outlined in the user requirements

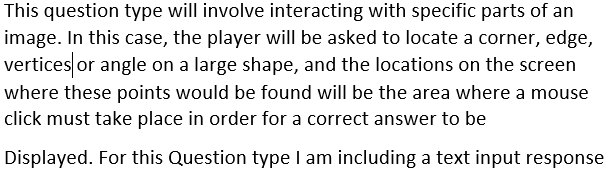
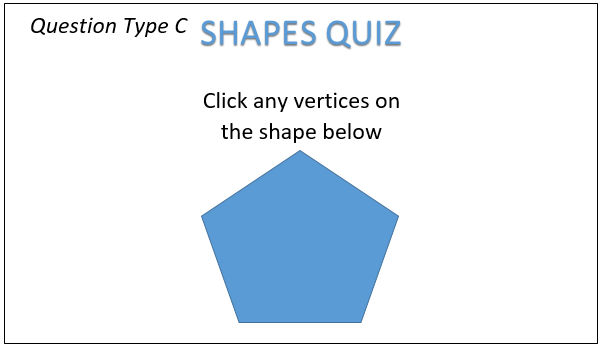


Difficulty options promote elitism in children (Single difficulty)

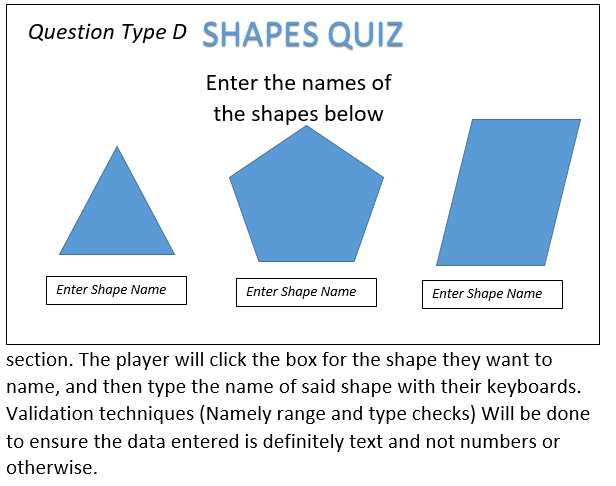


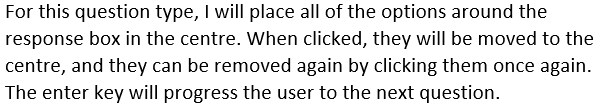
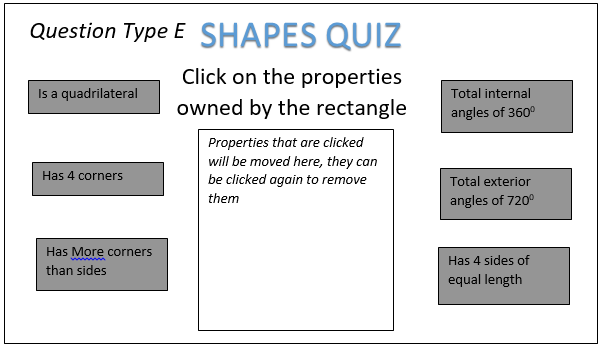


All shapes should be 2D, T/F response.



Vertices not studied at P7





Take advantage of visuals for this question

## Consultation with the Client

I brought my initial designs to the client to gain feedback before the detailed design process took place.

Firstly, the client noted that the log in screen was fine but noted that I hadn’t include a way to sign up in the first place. He noted that my Sign Up screen should have three fields, the third of which being a confirm password field and asked that they were placed on separate forms to prevent confusion on behalf of the children

In terms of question content the client has stated that whilst some question types; namely A and D, the content was exceptional, but that for question types B and C the questioning of ‘Vertices’ is a topic not studied until post primary, and requested that the subject matter of these questions were changed to include angles and exclusively 2D shapes.

The client noted that I had no plans for a menu strip with tool bar items in place and basically no inclusion of efficient help facilities but understood that help is too specific to the final project to begin coding now; nonetheless he requested that the menu and toolbar items outlined in the agreed requirements were satisfied.

The lack of a High Score table was alarming to the client and he requested that attention was drawn to that.

For the drag and drop question (E) the client stated that I missed an opportunity to manipulate images and visuals which would be more effective at teaching the concepts of shapes, and requested that I used visuals for this question.

Specific request to remove difficulty selections as they promote elitism between students that creates a negative learning environment and potential arguments between students

# Detailed Storyboards

picTitle: Picturebox containing an image of the games title, ‘Title.png

Each form will have dimensions 640x480 and feature the background image ‘Background.png

## MasterForm

Game

Restart Quiz

Log Out

View

HighScores

Main Menu

Help

Exit

Exit System

User:

Score:

tspLogOut

tspHighScores

tspMainMenu

mnuExitSystem

lblUser

mnuGame

tspRestart

mnuView

tspHelp

mnuExit

lblScore

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| Masterform | PictureBox | picTitle | 21, 24 | 339, 44 | Has the games name, consistent position on all forms to add consistent branding. | N/A | Title.png | Transparent |
| Masterform | MenuStrip | mnuGame | Standard | Standard | On click, show ‘tspRestart’ and ‘tspLogOut’ | Segoe UI, 9pt | Control Text | Control |
| Masterform | MenuStrip | mnuView | Standard | Standard | On click, show ‘tspExitSystem’ | Segoe UI, 9pt | Control Text | Control |
| Masterform | MenuStrip | mnuExit | Standard | Standard | On click, show ‘tspExitSystem’ | Segoe UI, 9pt | Control Text | Control |
| Masterform | Tool Strip | tspRestart | Standard | Standard | On click, show ‘frmQuestion1’ and change Player.Score to 0 | Segoe UI, 9pt | Control Text | Control |
| Masterform | Tool Strip | tspLogOut | Standard | Standard | On click, show ‘frmLogIn’ and change Player.Score to 0 | Segoe UI, 9pt | Control Text | Control |
| Masterform | Tool Strip | tspHighscore | Standard | Standard | On click, show ‘frmHighScore’ and change Player.Score to 0 | Segoe UI, 9pt | Control Text | Control |
| Masterform | Tool Strip | tspMainMenu | Standard | Standard | On click, show ‘MainMenu’ and change Player.Score to 0 | Segoe UI, 9pt | Control Text | Control |
| Masterform | Tool Strip | tspHelp | Standard | Standard | On click, show a messagebox containing help. | Segoe UI, 9pt | Control Text | Control |
| Masterform | Tool Strip | tspExitSystem | Standard | Standard | On click, close the program. | Segoe UI, 9pt | Control Text | Control |
| Masterform | Label | lblUser | 18, 412 | 43, 16 | To display the players username consistently | Microsoft Sans Serif, 9.75pt | Control Text | Transparent |
| Masterform | Label | lblScore | 12, 428 | 50, 16 | To show the players current score whilst they play. | Microsoft Sans Serif, 9.75pt | Control Text | Transparent |
| Masterform | Background Image | - | Full screen | Full screen | Consistent backdrop to the game. | - | - | - |

Loading…%

**lblLoading**

**pgrLoadingBar**

## Loading Screen

A timer used to increase the progress bar with its increment, not visible

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmLoadingScreen | Progress Bar | pgrLoadingBar | 146, 282 | 339, 44 | Mainly aesthetic; allows the rest of the quiz time to load | N/A | Highlight | Control |
| frmLoadingScreen | Label | lblLoading | 241, 243 | 147, 36 | To display the percentage of the bar that has been filled by | Microsoft Sans Serif, 22pt | Control Text | Transparent |
| frmLoadingScreen | Timer | Timer | N/A | N/A | Increase progress bar by 20 every second in increments of 5 | N/A | N/A | N/A |

## Splash Screen

(LogIn.png)

(Help.png)

(Exit.png)

**btnLogIn**

**btnHelp**

**btnSignUp**

**btnExit**

(SignUp.png)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmSplashScreen | Button | btnLogIn | 207,95 | 210,62 | On click, frmLogIn will be displayed | N/A | LogIn.png | N/A |
| frmSplashScreen | Button | btnSignUp | 207,177 | 210,62 | On click, frmSignUp will be displayed | N/A | SignUp.png | N/A |
| frmSplashScreen | Button | btnHelp | 207,257 | 210,62 | On click, frmHelp will be displayed | N/A | Help.png | N/A |
| frmSplashScreen | Button | btnExit | 207,344 | 210,62 | On click, the application will be closed. | N/A | Exit.png | N/A |

Passsword

Username

## Log In

Sign Up

Log In

Log In

**lblLogIn**

**txtUsername**

**lblUserName**

**btnSignUp**

**lblPassword**

**txtPassword**

**btnLogIn**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmLogIn | Label | lblLogIn | 247, 91 | 141, 51 | To display the name of the page | Microsoft sans serif, 32pt | Control Light  Light | Transparent |
| frmLogIn | Label | lblUsername | 128, 186 | 110, 25 | To display ‘Username’ beside txtUsername | Microsoft sans serif, 15.75t | Control Text | Transparent |
| frmLogIn | Label | lblPassword | 132, 263 | 106,25 | To display ‘Password’ beside txtPassword | Microsoft sans serif, 15.75pt | Control Text | Transparent |
| frmLogIn | Textbox | txtUsername | 256, 192 | 219,20 | To allow the user to enter their accounts username to be verified against the saved username they signed up with, on click of btnLogIn provided the password entered in txtPassword is also correct | Microsoft sans serif, 8.25pt | Control Text | Window |
| frmLogIn | Textbox | txtPassword | 256, 268 | 219,20 | To allow the user to enter their accounts password to be verified against the saved password they signed up with, on click of btnLogIn provided the Username entered in txtUsername is also correct | Microsoft sans serif, 8.25pt | Control Text | Window |
| frmLogIn | Button | btnSignUp | 137,337 | 157,54 | Displays frmSignUp on click | N/A | SingUp.png | Transparent |
| frmLogIn | Button | btnLogIn | 344,337 | 131,54 | Verifies Username and Password, displays frmMainMenu | N/A | LogIn.png | Transparent |

## Sign Up

Log In

Sign Up

Sign Up

Confirm Password

Password

Username

**lblSignUp**

**txtUsername**

**lblUsername**

**lblConfirmPassword**

**txtPassword**

**lblPassword**

**btnLogIn**

**txtConfirmPassword**

**btnSignUp**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour**  **/Image** | **Backcolour**  **/Image** |
| frmSignUp | Label | lblSignUp | 229,91 | 175,51 | To display the name of the page | Microsoft sans serif, 32pt | Console Light Light | Transparent |
| frmSignUp | Label | lblUsername | 148,181 | 110,181 | To display ‘Username’ beside txtUsername | Microsoft sans serif, 15.75pt | Control Text | Transparent |
| frmSignUp | Label | lblPassword | 148,238 | 106,25 | To display ‘Password’ beside txtPassword | Microsoft sans serif, 15.75pt | Control Text | Transparent |
| frmSignUp | Label | lblConfirm | 72,294 | 186,25 | To display ‘Confirm Password’ beside txtConfirm | Microsoft sans serif, 15.75pt | Control Text | Transparent |
| frmSignUp | Textbox | txtUsername | 298,181 | 219,20 | Enter a new username which will be saved as the username for the account being made. Validates that the username has only letters/numbers and that said username is not already taken | Microsoft sans serif, 8.25pt | Window Text | Window |
| frmSignUp | Textbox | txtPassword | 298,243 | 219,20 | Enter the password for the account being made. Will be saved to a txt file alongside Username once the password is validated against txtconfirm | Microsoft sans serif, 8.25pt | Window Text | Window |
| frmSignUp | Textbox | txtConfirm | 298,299 | 219,20 | Data entered is compared with the data entered into txtPassword as validation | Microsoft sans serif, 8.25pt | Window Text | Window |
| frmSignUp | Button | btnLogIn | 153,353 | 131,52 | On Click open frmLogIn | N/A | LogIn.png | N/A |
| frmSignUp | Button | btnSignUp | 360,353 | 157,54 | On click execute the sign up code and open frmMainMenu | N/A | SignUp.png | N/A |

**btnLogOut**

**btnHighScores**

**btnPlayGame**

**btnExit**

HighScores

Log Out

Exit

Play Game

## Main Menu

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmMainMenu | Button | btnPlayGame | 62, 178 | 191, 95 | Displays frmQuestion1 on click | Times New Roman, 20pt | ControlText | Control |
| frmMainMenu | Button | btnHighScores | 380, 178 | 191, 95 | Displays frmHighScores on click | Times New Roman, 20pt | ControlText | Control |
| frmMainMenu | Button | btnLogOut | 62, 325 | 191, 95 | Displays frmLogIn on click | Times New Roman, 20pt | ControlText | Control |
| frmMainMenu | Button | btnExit | 380, 325 | 191, 95 | Closes the game on click. | Times New Roman, 20pt | ControlText | Control |

## Question 1

(Picked from array)

(Picked from array)

(Picked from array)

What is the name of the shape below?

**picShape**

**lblQuestion**

**btnAnswer3**

**btnAnswer2**

**btnAnswer1**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmQuestion1 | Label | lblQ | 82, 103 | 465, 34 | Display the question | Times New Roman, 22pt | Control Light Light | Transparent |
| frmQuestion1 | PictureBox | picShape | 149, 153 | 337, 166 | To display the picture the question refers to | N/A | Image from Array | Transparent |
| frmQuestion1 | Button | btnAnswerA | 62, 352 | 134, 79 | To select Answers | Times New Roman, 17pt | Control Text | Control |
| frmQuestion1 | Button | btnAnswerA | 252, 352 | 134, 79 | To select Answers | Times New Roman, 17pt | Control Text | Control |
| frmQuestion1 | Button | btnAnswerA | 436, 352 | 134, 79 | To select Answers | Times New Roman, 17pt | Control Text | Control |

Question 2

Enter the names of the shapes below

(Octagon.png)

(Hexagon.png)

Submit

**lblQuestion**

**picOctagon**

**txtOcatagon**

**txtHexagon**

**picHexagon**

**btnSubmit**

## Question 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmQuestion2 | Label | lblQuestion | 105, 106 | 454, 34 | Display the name of the question | Times New Roman, 22pt | ControlLightLight | Transparent |
| frmQuestion2 | PictureBox | picOctagon | 57, 146 | 210, 153 | Shows a picture of an Octagon | - | Hexagon.png | Transparent |
| frmQuestion2 | PictureBox | picHexagon | 360, 146 | 210, 153 | Shows a picture of a Hexagon | - | Octagon.png | Transparent |
| frmQuestion2 | Textbox | txtOctagon | 57, 332 | 210, 20 | Allows the user to attempt to answer the question by typing Octagon | Microsoft Sans Serif, 8.25pt | Control Text | Window |
| frmQuestion2 | Textbox | txtHexagon | 360, 332 | 210, 20 | Allows the user to attempt to answer the question by typing Hexagon | Microsoft Sans Serif, 8.25pt | Control Text | Window |
| frmQuestion2 | Button | btnSubmit | 231, 371 | 173, 58 | Shows frmQuestion3 and checks if the question has been answered correctly | Times New Roman, 20.25pt | Control Text | Control |

## Question 3

Complete the statements with the appropriate responses below

1. A quadrilateral shape is one with- - Sides and -Corners

2. A triangular shape is one with - Sides and - Corners

3. A Hexagonal shape is one with - Sides and - Corners

4. An Octagonal shape is one with - Sides and - Corners

lblQuad1

Lbltri3

CbTri2

lblOct3

lblOct2

cbQuad1

lblQuad2

lbltri1

cbTri1

lblQuad1

lblQuad3

cbQuad2

lblTri2

cbHex2

lblHex3

lblOct1

lblHex1

cbHex1

cbOct1

lblHex2

cbOct2

btnSubmit

Submit

SUBMIT

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmQuestion3 | Label | lblQuestion | 17, 150 |  | Display the question | N/A | ControlLightLight | Transparent |
| frmQuestion3 | Label | lblQuad1 | 17, 150 | 301, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblQuad2 | 371, 150 | 91, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblQuad3 | 514, 150 | 74, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | LblTri1 | 17, 210 | 276, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblTri2 | 371, 210 | 91, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblTri3 | 514, 210 | 74, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblHex1 | 17, 270 | 291, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblHex2 | 371, 270 | 91, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblHex3 | 514, 270 | 74, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblOct1 | 17, 330 | 296, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblOct2 | 371, 330 | 91, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Label | lblOct3 | 514, 330 | 74, 24 | Display text | Microsoft Sans Serif, 14pt | ControlText | Transparent |
| frmQuestion3 | Combobox | cbQuad1 | 321, 150 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbQuad2 | 465, 150 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbTri1 | 321, 210 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbTri2 | 465, 210 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbHex1 | 321, 270 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbHex2 | 465, 270 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbOct1 | 321, 330 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Combobox | cbOct2 | 465, 330 | Default | Select Numeric Response (1-9) | Default | Default | Default |
| frmQuestion3 | Button | btnSubmit | 228, 383 | 173, 58 | Check answer, allocate points and open frmQuestion4 | Times New Roman, 20.25pt | ControlText | Control |

## Question 4

1. The internal angles of all triangles add up to 180°

2. The internal angles of all Triangles add up to 360°

3. The internal angles of all squares add up to 270°

4. The internal angles of all squares add up to 360°

TRUE/FALSE

TRUE/FALSE

TRUE/FALSE

TRUE/FALSE

SUBMIT

Which of the Following Are True?

lblQuestion

lblQuestion4

lblQuestion3

lblQuestion2

lblQuestion1

grpQuestion1

rdbQ1T-rdbQ1F

grpQuestion1

rdbQ2T-rdbQ2F

grpQuestion1

rdbQ3T-rdbQ3F

grpQuestion1

rdbQ4T-rdbQ4F

btnSubmit

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmQuestion4 | Label | lblQuestion | 118, 100 | 400, 34 | Displays the general question type | Times New Roman, 22pt | ControlLight  Light | Transparent |
| frmQuestion4 | Label | lblQuestion1 | 17, 145 | 506, 26 | Displays the first question | Microsoft Sans Serif, 16pt | ControlText | Transparent |
| frmQuestion4 | Label | lblQuestion2 | 17, 220 | 506, 26 | Displays the second question | Microsoft Sans Serif, 16pt | ControlText | Transparent |
| frmQuestion4 | Label | lblQuestion3 | 17, 295 | 506, 26 | Displays the third question | Microsoft Sans Serif, 16pt | ControlText | Transparent |
| frmQuestion4 | Label | lblQuestion4 | 17, 370 | 506, 26 | Displays the fourth question | Microsoft Sans Serif, 16pt | ControlText | Transparent |
| frmQuestion4 | ComboBox/  RadioButton | grpQuestion1  rdbQ1T-rdbQ1F | 525, 134 | 76, 37 | Allows the user to toggle between the true and false responses for the first question. | Microsoft Sans Serif, 8.25pt | ControlText | Transparent |
| frmQuestion4 | ComboBox/  RadioButton | grpQuestion2  rdbQ2T-rdbQ2F | 525, 209 | 76, 37 | Allows the user to toggle between the true and false responses for the first question. | Microsoft Sans Serif, 8.25pt | ControlText | Transparent |
| frmQuestion4 | ComboBox/  RadioButton | grpQuestion3  rdbQ3T-rdbQ3F | 525, 284 | 76, 37 | Allows the user to toggle between the true and false responses for the first question. | Microsoft Sans Serif, 8.25pt | ControlText | Transparent |
| frmQuestion4 | ComboBox/  RadioButton | grpQuestion4  rdbQ4T-rdbQ4F | 525, 359 | 76, 37 | Allows the user to toggle between the true and false responses for the first question. | Microsoft Sans Serif, 8.25pt | ControlText | Transparent |
| frmQuestion4 | Button | btnSubmit | 500, 408 | 125, 39 | Executes code to check which answers were right, allocates score accordingly (+50/correct answer, -5/incorrect answer) and shows frmQuestion5 | Microsoft Sans Serif, 18pt | ControlText | Control |

## Question 5

lblPentagon

lblOval

lblParallelogram

lblQuestion

picPentagon

picParallelogram

picOval

(Parallelogram.png)

(Pentagon.png)

(Oval.png)

OVAL

Pentagon

Parallelogram

Drag the Names to Their Respective Shapes

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmQuestion5 | Label | lblQuestion | 49, 103 | 537, 34 | Displays the question type | Times New Roman, 22pt | ControlLightLight | Transparent |
| frmQuestion5 | Label | lblOval | 52, 171 | 78, 40 | When clicked label can be dragged by user to response section | Comic Sans MS, 21.75pt | ControlText | Orange |
| frmQuestion5 | Label | lblPentagon | 48, 265 | 138, 40 | When clicked label can be dragged by user to response section | Comic Sans MS, 21.75pt | ControlText | Orange |
| frmQuestion5 | Label | lblParallelogram | 52, 360 | 201, 40 | When clicked label can be dragged by user to response section | Comic Sans MS, 21.75pt | ControlText | Orange |
| frmQuestion5 | PictureBox | picOval | 467, 152 | 134, 83 | If ‘lblOval’ is dropped here Player.Score is increased by 100 points and both labels disappear, if any other labels are dragged both picOval and lblOval are made invisible | - | Oval.png | Transparent |
| frmQuestion5 | PictureBox | picPentagon | 467, 265 | 134, 63 | If ‘lblOval’ is dropped here Player.Score is increased by 100 points and both labels disappear, if any other labels are dragged both picOval and lblOval are made invisible | - | Pentagon.png | Transparent |
| frmQuestion5 | PictureBox | picParallelogram | 467, 348 | 134, 79 | If ‘lblOval’ is dropped here Player.Score is increased by 100 points and both labels disappear, if any other labels are dragged both picOval and lblOval are made invisible | - | Parallelogram.png | Transparent |

## Results Screen

Thank you for playing, (Username)!

You Scored (Player.Score)!

High Scores

Exit Game

Main Menu

lblThank

lblScore

btnHighScores

btnMainMenu

btnExitGame

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmResultsScreen | Label | lblThank | 127, 160 | 392, 34 | Displays the name of the player in a thank you message | Times New Roman, 22pt | ControlLightLight | Transparent |
| frmResultsScreen | Label | lblScore | 171, 245 | 312, 34 | Displays the players score in congratulatory message | Times New Roman, 22pt | ControlText | Transparent |
| frmResultsScreen | Button | btnHighScores | 64, 348 | 128, 61 | Displays frmHighScore  -Uses CheckScore method to compare current score with previous high score  -If it’s a new highscore, writes it to the score file as a highscore | Microsoft Sans Serif, 8.25pt | ControlText | Control |
| frmResultsScreen | Button | btnMainMenu | 255, 348 | 128, 61 | Displays frmMainMenu  -Uses CheckScore method to compare current score with previous high score  -If it’s a new highscore, writes it to the score file as a highscore | Microsoft Sans Serif, 8.25pt | ControlText | Control |
| frmResultsScreen | Button | btnExitGame | 443, 348 | 128, 61 | Closes the game.  -Uses CheckScore method to compare current score with previous high score  -If it’s a new highscore, writes it to the score file as a highscore | Microsoft Sans Serif, 8.25pt | ControlText | Control |

## High Scores

btnMainMenu

lblScore(1-5)

lblHighScores

Main Menu

1. – UserName – Score
2. – UserName – Score
3. – UserName – Score
4. – UserName – Score
5. – UserName – Score

HIGH SCORES

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Form** | **Property** | **Property Name** | **Position** | **Size** | **Purpose** | **Font** | **Forecolour/**  **Image** | **Backcolour/**  **Image** |
| frmHighScores | Label | lblHighScores | 212, 116 | 214, 34 | Display the name of the screen | Times New Roman, 22pt | ControlLightLight | Transparent |
| frmHighScores | Label | lblScores(1-5) | 212, 160 | 139, 31 | Draws from the SortedScores and SortedNames arrays to list the top 5 scoring players. | Microsoft Sans Serif, 20pt | ControlText | Transparent |
| frmHighScores | Button | btnMainMenu | 492, 380 | 128, 61 | Brings the player to the main menu | Microsoft Sans Serif, 8.25pt | ControlText | Control |

# Test Plan

|  |  |  |
| --- | --- | --- |
| # | Form | Intended Result |
| 1 | frmLoadingScreen | The percentage outlined in ‘lblLoading’ will correspond with the percentage of ‘pgrLoadingBar |
| 2 | frmLoadingScreen | Once the progress bar fills the user should automatically be brought to ‘frmSplashScreen |
| 3 | frmSplashScreen | When the user clicks on ‘btnLogIn’ they should be brought to ‘frmLogIn’ |
| 4 | frmSplashScreen | When the user clicks on ‘btnSignUp’ they should be brought to ‘frmSignUp’ |
| 6 | frmSplashScreen | When the user clicks on ‘btnExit’ the application should terminate. |
| 7 | frmSignUp | When the user types a different password in ‘txtConfirm’ than in ‘txtPassword’ an error prompt is displayed and the user is urged to check that their passwords match. |
| 8 | frmSignUp | If a username is already saved in users.txt then the user is prompted to pick another username. |
| 9 | frmSignUp | When btnSignUp is clicked by the user, provided that ‘txtPassword’ and ‘txtConfirm’ have the same contents and that the username is unique, the username and password will be saved to users.txt |
| 10 | frmSignUp | When btnLogIn is clicked, ‘frmLogIn’ is opened. |
| 11 | frmLogIn | When the user clicks ‘btnSignUp’ they are brought to ‘frmSignUp’ |
| 12 | frmLogIn | If the data entered in to ‘txtUsername’ and ‘txtPassword’ is not also in ‘users.txt’ the user is prompted to check their log in credentials. |
| 13 | frmLogIn | If the data entered in to ‘txtUsername’ and ‘txtPassword’ is also in ‘users.txt’ the user is presented with a textbox reading ‘Welcome Back, (txtUsername.text)!’ and ‘frmMainMenu’ is opened |
| 14 | frmMainMenu | When btnPlayGame is clicked, the user should navigate to frmQuestion1 |
| 15 | frmMainMenu | When btnLogOut is clicked, the user should navigate to frmLogIn |
| 16 | frmMainMenu | When btnHighScores is clicked, the user should navigate to frmHighScores |
| 17 | frmMainMenu | When btnExit is clicked, the game should be closed. |
| 18 | frmQuestion1 | Upon opening of the form, a random number will be generated and used to select one of 5 possible questions. |
| 19 | frmQuestion1 | lblUsername should have the players username following ‘User:’ |
| 20 | frmQuestion1 | When a correct answer is chosen based on the coding of the question. ‘Player.Score’ should be increased by 100, and a message box should appear congratulating the user, and they should automatically be navigated to ‘frmQuestion2’ |
| 21 | frmQuestion1 | When an incorrect answer is chosen based on the coding of the question, a message box should appear telling the user that their answer was incorrect, and they should automatically be navigated to ‘frmQuestion2’ |
| 22 | frmQuestion1 | When ‘tspLogOut’ is clicked the user should be brought to ‘frmLogIn’ |
| 23 | frmQuestion1 | When ‘tspHighScores’ is clicked the user should be brought to ‘frmHighScores’ |
| 24 | frmQuestion1 | When ‘tspMainMenu’ is clicked the user should be brought to ‘frmMainMenu’ |
| 25 | frmQuestion1 | When ‘tspHelp’ A Messagebox should pop up containing advice on answering the question |
| 26 | frmQuestion1 | When ‘tspExitSystem’ is clicked the program should close. |
| 27 | frmQuestion2 | On Click of ‘btnSubmit’ a messagebox appears with a potential of 3 different messages depending on response to question which will be further explained in tests 28-30. ‘frmQuestion3’ is shown. |
| 28 | frmQuestion2 | If both answers are correct, a messagebox containing the words ‘Correct Answers, Well Done!’ will appear. Player.Score will be increased by 200. |
| 29 | frmQuestion2 | If only one answer is correct, a messagebox containing the words ‘1/2 Answers Correct, Good Effort!’ will appear. Player.Score will be increased by 100. |
| 30 | frmQuestion2 | If neither answer is correct a messagebox containing ‘Incorrect answer’ will be shown. |
| 31 | frmQuestion3 | When ‘btnConfirm’ is selected a messagebox will appear reading “(answers)/8 Answers Correct!” filling in the number of correct answers |
| 32 | FrmQuestion3 | For each right answer, Player.Score should increase by 50. For each incorrect answer, Player.Score should be reduced by 5. |
| 33 | frmQuestion4 | When ‘btnSubmit’ is selected a messagebox will appear reading “(answers)/4 Answers Correct!” filling in the number of correct answers |
| 34 | frmQuestion4 | For each right answer, Player.Score should increase by 50. For each incorrect answer, Player.Score should be reduced by 5. |
| 35 | frmQuestion5 | All three labels should be dragable, however if they are dropped outside of any of the picture they should remain visible and in their initial position. |
| 36 | frmQuestion5 | When a label is dragged to its matching image and dropped, both the image and the label should disappear and Player.Score should increase by 100 each time |
| 37 | frmQuestion5 | If A label is dragged to the wrong image, the image it is dragged to and its match should vanish and the label being dragged should return to its neutral position |
| 38 | frmQuestion5 | Each time a label is dragged to an image an unseen variable called Answers increases by one. Once three answers have been attempted, an if statement should be fulfilled that opens ‘frmResultScreen’ |
| 39 | frmResultScreen | lblScore and lblThanks should have the variables ‘Player.Score’ and ‘Player.Username’ shown to read “You scored (Score) Points!” |
| 40 | frmResultScreen | When btnHighScores is clicked, frmHighScores Should open. |
| 41 | frmResultScreen | When btnMainMenu is clicked, frmMainMenu Should open. |
| 42 | frmResultScreen | When btnExit Is clicked, the game should be closed. |
| 43 | frmResultsScreen | When any of the buttons are clicked the players score must be compared with their highest score and if it is greater, written to the scores files as a High Score |
| 44 | frmHighScores | Scores should be ordered highest to lowest and the names of the player should be displayed beside their scores when drawn from their arrays. |
| 45 | frmHighScores | When btnMainMenu is clicked, frmMainMenu should be shown |

# Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Form | Intended Result | Pass/Fail | Evidence | Corrective Action |
| 1 | frmLoadingScreen | The percentage outlined in ‘lblLoading’ will correspond with the percentage of ‘pgrLoadingBar | PASS | pg50 | N/A |
| 2 | frmLoadingScreen | Once the progress bar fills the user should automatically be brought to ‘frmSplashScreen | PASS | pg50 | N/A |
| 3 | frmSplashScreen | When the user clicks on ‘btnLogIn’ they should be brought to ‘frmLogIn’ | PASS | pg51 | N/A |
| 4 | frmSplashScreen | When the user clicks on ‘btnSignUp’ they should be brought to ‘frmSignUp’ | PASS | pg52 | N/A |
| 5 | frmSplashScreen | When the user clicks on ‘btnExit’ the application should terminate. | PASS | pg52 | N/A |
| 6 | frmSignUp | When the user hasn’t filled each field when the sign up button is clicked an error message should appear | PASS | pg53 | N/A |
| 7 | frmSignUp | When the user types a different password in ‘txtConfirm’ than in ‘txtPassword’ an error prompt is displayed and the user is urged to check that their passwords match. | PASS | pg54 | N/A |
| 8 | frmSignUp | If a username is already saved in users.txt then the user is prompted to pick another username. | PASS | pg55 | N/A |
| 9 | frmSignUp | When btnSignUp is clicked by the user, provided that ‘txtPassword’ and ‘txtConfirm’ have the same contents and that the username is unique, the username and password will be saved to users.txt | PASS | pg56 | N/A |
| 10 | frmSignUp | When btnLogIn is clicked, ‘frmLogIn’ is opened. | PASS | pg57 | N/A |
| 11 | frmLogIn | When the user clicks ‘btnSignUp’ they are brought to ‘frmSignUp’ | PASS | pg57 | N/A |
| 12 | frmLogIn | If the data entered in to ‘txtUsername’ and ‘txtPassword’ is not also in ‘users.txt’ the user is prompted to check their log in credentials. | FAIL (Corrected | pg58 | Whilst entering incorrect credentials did not allow the user access to frmMainMenu, no feedback was given to the user. I rectified this by making a messagebox show reading ‘Error: Check Your Credentials and Try Again; |
| 13 | frmLogIn | If the data entered in to ‘txtUsername’ and ‘txtPassword’ is also in ‘users.txt’ the user is presented with a textbox reading ‘Welcome Back, (txtUsername.text)!’ and ‘frmMainMenu’ is opened | PASS | pg59 | N/A |
| 14 | frmMainMenu | When btnPlayGame is clicked, the user should navigate to frmQuestion1 | PASS | pg60 | N/A |
| 15 | frmMainMenu | When btnLogOut is clicked, the user should navigate to frmLogIn | PASS | pg60 | N/A |
| 16 | frmMainMenu | When btnHighScores is clicked, the user should navigate to frmHighScores | PASS | pg61 | N/A |
| 17 | frmMainMenu | When btnExit is clicked, the game should be closed. | PASS | pg62 | N/A |
| 18 | frmQuestion1 | Upon opening of the form, a random number will be generated and used to select one of 5 possible questions. | PASS | pg63 | N/A |
| 19 | frmQuestion1 | lblUsername should have the players username following ‘User:’ | PASS | Pg64 | N/A |
| 20 | frmQuestion1 | When a correct answer is chosen based on the coding of the question. ‘Player.Score’ should be increased by 100, and a message box should appear congratulating the user, and they should automatically be navigated to ‘frmQuestion2’ | PASS | Pg65 | N/A |
| 21 | frmQuestion1 | When an incorrect answer is chosen based on the coding of the question, a message box should appear telling the user that their answer was incorrect, and they should automatically be navigated to ‘frmQuestion2’ | PASS | Pg66 | N/A |
| 22 | frmQuestion1 | When ‘tspLogOut’ is clicked the user should be brought to ‘frmLogIn’ | PASS | Pg67 | N/A |
| 23 | frmQuestion1 | When ‘tspHighScores’ is clicked the user should be brought to ‘frmHighScores’ | PASS | Pg68 | N/A |
| 24 | frmQuestion1 | When ‘tspMainMenu’ is clicked the user should be brought to ‘frmMainMenu’ | PASS | Pg68 | N/A |
| 25 | frmQuestion1 | When ‘tspHelp’ A Messagebox should pop up containing advice on answering the question | PASS | Pg69 | N/A |
| 26 | frmQuestion1 | When ‘tspExitSystem’ is clicked the program should close. | PASS | Pg70 | N/A |
| 27 | frmQuestion2 | On Click of ‘btnSubmit’ a messagebox appears with a potential of 3 different messages depending on response to question which will be further explained in tests 28-30. ‘frmQuestion3’ is shown. | PASS | Pg71 | N/A |
| 28 | frmQuestion2 | If both answers are correct, a messagebox containing the words ‘Correct Answers, Well Done!’ will appear. Player.Score will be increased by 200. | PASS | Pg72 | N/A |
| 29 | frmQuestion2 | If only one answer is correct, a messagebox containing the words ‘1/2 Answers Correct, Good Effort!’ will appear. Player.Score will be increased by 100. | PASS | Pg73 | N/A |
| 30 | frmQuestion2 | If neither answer is correct a messagebox containing ‘Incorrect answer’ will be shown. | PASS | Pg74 | N/A |
| 31 | frmQuestion3 | When ‘btnConfirm’ is selected a messagebox will appear reading “(answers)/8 Answers Correct!” filling in the number of correct answers | PASS | Pg75 | N/A |
| 32 | FrmQuestion3 | For each right answer, Player.Score should increase by 50. For each incorrect answer, Player.Score should be reduced by 5. | PASS | Pg75 | N/A |
| 33 | frmQuestion4 | When ‘btnSubmit’ is selected a messagebox will appear reading “(answers)/4 Answers Correct!” filling in the number of correct answers | PASS | Pg76 | N/A |
| 34 | frmQuestion4 | For each right answer, Player.Score should increase by 50. For each incorrect answer, Player.Score should be reduced by 5. | PASS | Pg76 | N/A |
| 35 | frmQuestion5 | All three labels should be dragable, however if they are dropped outside of any of the picture they should remain visible and in their initial position. | PASS | Pg77 | N/A |
| 36 | frmQuestion5 | When a label is dragged to its matching image and dropped, both the image and the label should disappear and Player.Score should increase by 100 each time | PASS | Pg78 | N/A |
| 37 | frmQuestion5 | If A label is dragged to the wrong image, the image it is dragged to and its match should vanish and the label being dragged should return to its neutral position | PASS | Pg79 | N/A |
| 38 | frmQuestion5 | Each time a label is dragged to an image an unseen variable called Answers increases by one. Once three answers have been attempted, an if statement should be fulfilled that opens ‘frmResultScreen’ | PASS | Pg79 | N/A |
| 39 | frmResultScreen | lblScore and lblThanks should have the variables ‘Player.Score’ and ‘Player.Username’ shown to read “You scored (Score) Points!” amd | PASS | Pg80 | N/A |
| 40 | frmResultScreen | When btnHighScores is clicked, frmHighScores Should open. | PASS | Pg80 | N/A |
| 41 | frmResultScreen | When btnMainMenu is clicked, frmMainMenu Should open. | PASS | Pg81 | N/A |
| 42 | frmResultScreen | When btnExit Is clicked, the game should be closed. | PASS | Pg81 | N/A |
| 43 | frmResultsScreen | When any of the buttons are clicked the players score must be compared with their highest score and if it is greater, written to the scores files as a High Score | PASS | Pg82 | N/A |
| 44 | frmHighScores | Scores should be ordered highest to lowest and the names of the player should be displayed beside their scores when drawn from their arrays. | PASS | Pg82 | N/A |
| 45 | frmHighScores | When btnMainMenu is clicked, frmMainMenu should be shown | PASS | Pg83 | N/A |

# Test Evidence

|  |  |
| --- | --- |
| # | Evidence |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| A. |  |
| B. |  |
| C. |  |
| D. |  |
| 19 |  |
| 20 |  |
| 21 |  |
| 22 |  |
| 23 |  |
| 24 |  |
| 25 |  |
| 26 |  |
| 27 |  |
| 28 |  |
| 29 |  |
| 30 |  |
| 31 |  |
| 32 | 50 x 7 = 350 – 5 = 345  180+345 = 525 |
| 33 |  |
| 34 | 2 Right = 100+  2 Wrong – 10-  100-10 = 90  525 + 90 = 615 |
| 35 |  |
| 36 |  |
| 37 |  |
| 38 |  |
| 39 | l |
| 40 |  |
| 41 |  |
| 42 |  |
| 43 | -Once New Score is entered |
| 44 |  |
| 45 |  |

# EVALUATION

The time bracket given for the creation of the game is coming to a close and I have been asked to extensively evaluate my progress both in relation to personal development and in terms of the satisfaction of user requirements; manifested most prominently in my response to criticism from beta testers.

Personal Evaluation

When I was first commissioned with this project I had no real prior experience in programming; at least on a practical level. Whilst I had limited experience using the Visual Studio Console to make programs with C#, this was my first time using windows form design to produce a user friendly visual interface with my program. Initially I found the new format somewhat difficult. The biggest aid in my production was my detailed designs, they helped me to visualise what the functioning quiz would look like when finished and gave me a chance to explain how exactly the quiz would work and ensure that my ideas made logical sense before attempting to produce them in the unfamiliar environment of windows form design. Having these detailed designs as a blueprint, my first course of action was to recreate each of the designs with the windows forms. This meant that I had a fully functional interface under which I could begin to code.

On the topic of code, I found that two main areas caused the most difficulty; the writing to and reading from of files, and the randomisation of question 1. Whilst the basic concepts of randomisation were relatively easy to grasp my original intention was to change the picture in the picturebox to a picture in an array, and to change the tag on said picture to reflect the name of the pictured shape. I couldn’t figure out how to make this work, so I switched instead to having 6 separate invisible picture boxes made invisible, and using a random number generator to generate a number which would make any one case take place in a switch statement. I found that C# was a suitable language to use for the product at hand as elements of encapsulation and inheritance were used throughout and hence I felt an Object Oriented language was necessary; with this in mind, Visual Studio does not support the use of Java and hence C# seemed like the optimal choice.

Coding however was not the only obstacle that I encountered during the creation of this application; another new concept to me was developing within the guidelines of a client. Previously I was able to freely create what I wanted, and if I encountered difficulty I merely left it out. As I had specific user requirements to meet and a target demographic to satisfy a large standard of communication was required between developer and client. If there was something I was unable to do – such as the randomisation of frmQuestion1 – It was my job to learn how to do it and to implement it on time. Communicating with the client was done through short meetings wherein they described what they wanted and I wrote them as a list of requirements that I had to satisfy. I also had to present the client with my ideas for initial designs as the requirements could be satisfied through coding but the actual aesthetic of the forms were subject to change hence feedback on the visuals using initial designs were necessary. The client also advised me on the types of questions to be asked on the quiz, a close co-operation between client and developer on this particular project because as this is educational the client needs to ensure the final product is suited to the specifications of the pupils attending their school and they have to use their knowledge of teaching and education to advise me on the best ways to present each question; this was mainly outlined in the question type section of the requirements, nonetheless however the quiz is not perfect and help from the client as to the language form to use for the ‘help’ sections may have made the quiz more accessible to the target audience.

Quality of design was a big area for me when it came to this; coding is quite a binary and logical process; as long as the coding works then your job has been done as far as the player is concerned. What isn’t quite as objective however is the design process which is entirely subjective and hence, never perfect. I had the title of the quiz in a metallic stylized font and had it on every form in a constant location, the background image of each form was also a constant; a variety of engaging and mentally stimulating colours in a consistent geometric pattern to coincide with the quiz’s theme. In terms of font I used white font with times new roman for the individual titles of each form and I stylised the buttons on the splash screen form. Due to time constraints I was unable to stylize the rest of my buttons but I maintained consistent house styles outlined in my property tables. Given more time, I would like to further develop the stylistic themes of the quiz to encompass every question type being presented with unique, stylised response sections. Moreover, I would have liked to put some research into font stylings that are most effective with learning in youths and then have implemented them in my work. For the most part the non-stylised buttons had the control background and contained ‘controltext’ font; these are largely just defaults and I would have preferred to have made it more fit for my target audience, even if they were universally intuitive and self-explanatory.

Time management was another area that I developed over the course of the project; Coding, designing or otherwise, the application had to be completed by May 1st, and regardless of computing competence or artistic talent, if I wasn’t efficient then I wouldn’t have been able to complete the application on time. I was undoubtedly pushed for time at the end - as I will explain in my User Requirement evaluation, - however I do feel that I have improved my time management over the course of this project; the constant thought of the deadline underpinned my thoughts, and once I had the coding knowledge confirmed I created a schedule of how much I had to complete each week, and then I ensured I followed this.

I also had to be accustomed to file handling; both the placement of files within the program (putting the ‘users.txt’ and ‘scores.txt’ files into the debug folder) and the process of backing up my program during development to ensure that not all data would be lost in the event that files were lost; something which I handled effectively. I had a copy locally, on the cloud, on two USB flashdrives and on my schools network. On my school network, even deleted files are still accessible by staff for up to a month after deletion, so regular backing up ensured a high degree of security.

The final area that I have developed considerably is my ability to format word documents; this document is 94 pages long and I had to make reference to specific pages for my testing. As of typing, this document contains 15 tables, a lengthy contents page and nearly 10,000 typed words. Over 50 images were inserted using the snipping tool and use of bullet points and indentation are also featured in this document. Page numbers, front covers and constant headings are all part of the huge development in my word processing skills that I earned from this project, however given more time I would have liked to implement a more efficient way to present testing evidence.

## Evaluation of satisfaction of User requirements:

Here I will evaluate in detail each requirement I had been issued, discussing their weaknesses, strengths and potential improvements. Consulting the testing section will show evidence for points raised, specific examples with singular instances for example file handling and randomisation will have specific references to testing

* Create an educational quiz game –

This application uses positive/negative feedback in the form of a ‘score’ as well as the use of message boxes between questions to ensure players know where they have succeeded and failed, hence they are able to learn from their mistakes and be rewarded for their strengths. This makes it an appropriate educational game

* Must be playable from a computer running windows 7

By using Visual Studio the finished product is playable on all windows Operating Systems between XP and 10, and this includes windows 7.

* It must teach the topic of Geometry
  + Focus on 2D shapes
  + The properties of quadrilateral and triangular shapes

The game contains questions on various 2D shapes with a focus on quadrilateral and triangular shapes, however it does not include the differences between equilateral and isosceles triangles. Had I been given a larger time bracket I would have introduced a question about the differences between these shapes.

* It must be intuitive enough for 7-11 year olds to be able to play it unsupervised
  + Help screen accessible from every other screen in the game

Rather than having one single help screen, I opted to create a custom help message box for each screen with help specific to each screen. This is accessed from the menu list at the top of each screen.

* + Tool tips included to explain what intractable properties do

I was able to use the tool tip function however due to time constraints I was unable to include tool tips for every property (as across the 14 forms of the quiz I had a total of over 100 properties). Given more time I would ensure to include a tool tip for every property to make the software as understandable to the user as possible.

* + Feedback after questions

This application uses positive/negative feedback in the form of a ‘score’ as well as the use of message boxes between questions to ensure players know where they have succeeded and failed, hence they are able to learn from their mistakes and be rewarded for their strengths. In the message boxes, it is specified through use of a variable how many questions the player answered correctly i.e.; “N/8 answers correct” with N being equal to the amount of correct answers for that question. Despite this however, to make it more educational I would ideally want question-specific feedback wherein for each incorrect answer the player is presented with help for that specific question, so that they can learn from their mistakes. This could be easily done by introducing a string variable to each incorrect answer that is added to the feedback Message Box when incorrect answers are selected, however it would have taken an extensive amount of time. Provided a longer time frame I’m confident that I could implement this useful function to the quiz

* It must have sign up/log in functions with user credentials held in a file

Usernames and passwords used in the ‘Sign Up’ form are saved to a file called ‘Users.bin’, (Test 9) the code also verifies that the username is unique and will present an error if said user already exists. The fields in the ‘Log In’ screen verifies that the credentials entered exist in the file and are accompanied by a valid password, in which case the user will sign in.

* Must feature at least 5 questions
  + Each question screen must have a unique response type

Each of my five question screens had a unique response type from the list outlined in the requirements.

* + True/False

“frmQuestion4” contains a question on the properties of shapes with a ‘true/false’ radio button response section provided. Through the use of the ‘Group Box’ property I was able to have only true or false selected at any one time, and by using a separate Group Box for each question. To improve screen however I could have held a large selection of questions in a file and had these questions read from the file and written to the labels in order to randomise the true/false questions the user is asked to answer, this would improve versatility.

* + Text Input

“frmQuestion2” Asks the user to type the names of the shapes shown into their respective textboxes. I understand some users will or won’t use capital letters in their responses, so I used a “ToUpper” method that automatically changes the text entered to all capitals, and I made the correct answer it is compared to in capitals also.

* + Button input

This was the most interesting question that I developed. I didn’t want to have distinct buttons with fixed text and one image on this question because the large sizes of the picture and buttons made it take up a lot of room on the screen. To increase versatility and usability, I decided to use a random number generator to create a variable that chooses which ‘case’ in a switch statement was executed; there were 5 options, and each one changed the text on the buttons and made a different image visible. To make the code as robust as possible, rather than setting one button as the consistent ‘right answer’ I created a method which compared a variable with the name of the shape shown (set within the switch statement) with the text on the button selected. This allowed one screen to be a vessel for a countless number of questions in this format. Whilst I’m very happy with how this question turned out, there are two improvements which I was unable to implement myself;

Firstly I wanted to remove the string variable ‘Answer’ and instead compare the text on the button with the selected images tag, however I couldn’t figure out how to do this. I was aware of how to do this with one picture, but unsure as to how I could change the image used in the ‘if’ statements to fit the image selected during randomisation.

* + Selecting options from a drop-down combo box

‘frmQuestion3’ has 4 sentences each featuring two drop-down combo box response sections. The player selects the correct response from the selection provided and they are checked at the end by comparing the int data selected with the correct answer. I felt that I used this function effectively and that I could apply it to other questions seamlessly.

* + Drag/drop

Undoubtedly the most difficult question type to learn how to use initially, I decided on using a drag and drop question type for ‘frmQuestion5’. I had to first make the labels drag-able in the events panel and then make the three images drop zone. If the label is dropped outside of a drop zone I assumed this would mean the player was reconsidering his choice, and hence I chose to have it return to its initial position should this happen. When all three drop zones have something dropped in them, the results screen is shown, points are also allotted for correct answers, causing a relative increase/decrease in score. I felt this question went relatively successfully considering the complexity of the coding behind it, however I would have liked to implement a randomisation function for the question wherein a cases from a switch statement choose pairs of labels/pictures to make the question replay-able.

* The application must be completed by May 1st

The application was finished on time, however there are various aspects outlined in this evaluation that I felt could have been enhanced provided more time to complete them. Time constraints were definitely an issues and part of that was due to the extensive amount of testing and pre-production involved in the project which I underestimated, despite this however the product does fulfil the agreed user requirements and thus I am happy with the job I have done.

* Navigation should be intuitive; appropriately named buttons, exhaustive use of a menu strip.

The Menu strip contains working buttons that when clicked, can take you to the high score table, the log in screen, the main menu screen and the first question. The game has a constant progressive feel; the log in screen will lead the player directly to the main menu, whose buttons each take you to a new form; each of these buttons’ destinations are outlined in the text on them. Once any question is answered the next question is automatically shown, and eventually the results screen follows suit; speaking of which, the results screen can be used to exit the game, access the main menu or view the results screen. I felt I satisfied this requirement sufficiently.

* Ensure every form bar the loading screen can be accessed from any other screen without needing to reload the game.

Selecting the ‘Main Menu’ menu item on the Sign Up/Log In forms will bring the player to the splash screen menu. The log in screen can be accessed from any other screen in the game past the Log In screen by use of the Log Out menu button, meaning that the game is fully intuitively navigatable.

* Randomisation used at least once to create a new user experience each time the game is played

Randomisation was used in frmQuestion1 (Test 18 A-E) by having a random number generator choose between 5 different cases in a switch statement that would change the question accordingly, however with this method I could have randomised all of my questions for an entirely unique gameplay experience every time. Due to time constraints however I was only able to randomise one question.

* A scoring system to track the pupils progress

The player class contains a score variable that is always displayed in the ‘lblScore’ property, this is added to or subtracted from during gameplay based on the players performance. I decided eventually on a system that saves high scores in a separate file names ‘scores.txt’ (Test 43) and saving the players username with their score to be placed on a highscore table, rather than saving the score within ‘users.bin’.

* + A high score table that compares high scores and shows the names of the best players

‘frmHighScores’ contains the top 5 highest scoring players along with their scores. (Test 44) This number can be extended to include more scores as required by adding more labels, but I stuck with 5 as I felt the form accommodated that number well.

* Help available on every usable form

Clicking the help option from the toolbar on any screen will present the player with a message box with help specific to that screen.

* Ability to restart the quiz any time

Clicking the restart button will bring the user back to frmQuestion1 and will set Player.Score to 0

* To be able to exit the quiz at any point during gameplay

Clicking Exit System followed by exit allows the player to exit at any point. I placed this function alone underneath its own menu option to prevent accidental clicking and loss of progress.

* Menu Strip and toolbar items consistent to include:
  + Logging out

I have a log out function included in the tool bar and it is active on every form after the log in form. It resets score to 0, and brings the player to the log in screen.

* + Restarting the quiz

Clicking the restart function in the tool bar will bring the user back to frmQuestion1 and will set Player.Score to 0

* + Exiting the quiz

Clicking the Exit System menu item followed by the Exit System tool will close the application

* + Viewing the high scores

frmHighScores is accessible from every file once logging in takes place, however doing so resets score to zero and doesn’t allow the user to return to the form they came from if they did so in the middle of the game. They can choose to directly restart the quiz however. Provided more time I would ensure that a variable was included as part of the player class that counts how many questions have been answered, then I could use this to bring the player back to the question they were on ie; ‘Answered = 3’ would activate a case on the switch statement that automatically opens the fourth question.

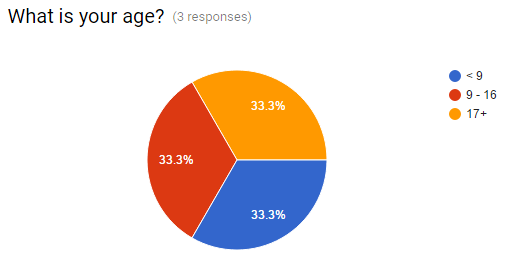
* + Displaying help

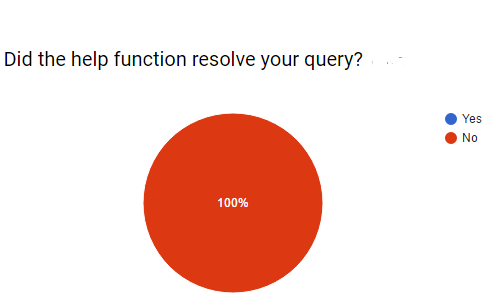
A question-specific help facility is available in the tool bar under the game menu, it can be used on all forms bar the loading screen, where no interaction takes place.

# Beta Testing and Response to Feedback

Having read the responses of the Beta testers I can now see the general opinion of the game so far.

I created a survey on google docs ( <http://goo.gl/forms/XH58gQ7kRF> ) and issued it to several players across three age-ranges; 9 and younger, 9-16, and 17+. I requested that the client would test one of their students on it too, found in ‘Emma McNeill’ the 9 Year old player.

Firstly, 66% of all players enjoyed the game; 100% of under 17 year olds enjoyed the game too, so the game appeals its demographic effectively. Older people enjoyed the game a lot less, meaning that the game isn’t nearly as enjoyable to older people. This narrows its potential audience, and If I wasn’t being commissioned to create an application specifically for a primary school I would more than likely create a more mature and universal aesthetic for the quiz.

100% of those surveyed found no difficulties whatsoever in completing the quiz even without the use of the help function. Adversely however, the target demographic of <12 Year olds found difficulty in answering at least one question, these difficulties were successfully addressed by the help function 100% of the time. This would indicate that the quiz is definitely useable by primary school children without the aid of teachers, however clearly there is some difficulty in interpreting the question based on the question itself without the use of a help function, so this is something that I could improve upon in future, perhaps by including lengthier subtitles.

Out of all beta testers, none of them encountered any problems which were not solvable by the help function, which was pleasing to hear. Younger audiences learned something new 100% of the time, but again older audiences found the content too easy, and learned nothing. One player however, needed a teacher’s assistance with logging in but this was most likely a human error wherein they forgot their own password.

The average rating for the game remains at 3.5/5 in its beta form which I’m currently comfortable with, I am aware of ways in which the application could be improved given more time, also outlined in this evaluation, and with these changes made I am confident that this rating could increase.

## Player Feedback: (Template)

Geometry Quiz Beta Testing

Having Completed the Quiz, Please complete this survey on your experience with it.

\* Required

What is your name? \*



Your answer

What is your age? \*

< 9

9 - 16

17+

Did you Enjoy the Game? \*

Yes

No

Were You Able to fully grasp the games controls without help? \*

Yes

No

If not, were you able to locate the help function?

Yes

No

Did the help function resolve your query?

Yes

No

Did you encounter any problems during game play that need amended? \*

Your answer



Did you feel this game taught you anything effectively? \*

Yes

No

Overall, how would you rate this game out of five? \*