PRACTICAL ASSESSMENT TASK

PHASE 1

[Jude Wells]

[10CI]

TASK DEFINITION

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

**PAT INSTRUCTION**

Write a brief description (approximately 200 words) in your own words to describe in general terms the problem/task and how the project will solve the problem.

Your explanation must highlight that:

* You understand the needs of the task that you have chosen.
* Your solution will solve the needs of the task.
* The scope of the project is clear and well defined in the format of a simple/brief description of the project.

Primary schools in many countries do not have great math marks. Math is the language of the universe, and is essential not only for your future career but for brain development. This program aids to help primary school kids understand mathematical concepts and have fun while learning and applying math.

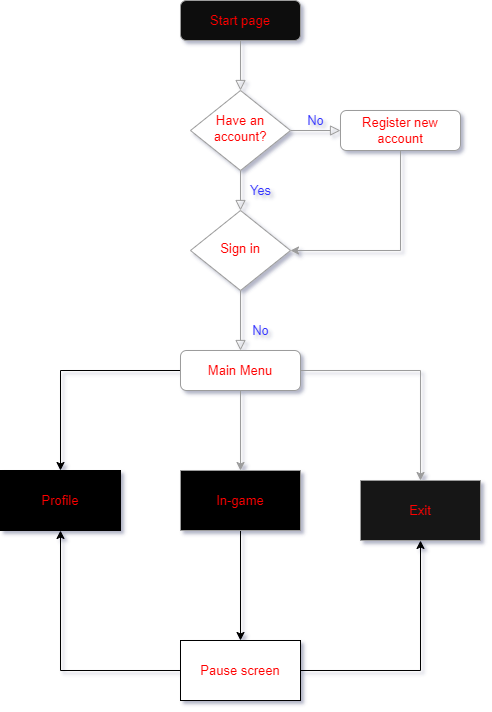
This programme will come in the form of a game. The interface the game will have is similar to a flashcard simulation, with a card with a sum on one side, and the answer on the other. However, the user must calculate the answer and once the answer is typed in, then the flash card will flip around.

This programme will be useful to primary school students as it is engaging and applying active recall to mathematical methods from the classroom. This is therefore then in essence, a game that can be used for studying.

USER REQUIREMENTS

|  |  |  |
| --- | --- | --- |
| **User Role/Who** | **Description** | **Limitations** |
| Primary school student | * May take quizzes * May view other player’s scores | * May not add new quizzes * May not edit existing quizzes * May not edit student stats |
| Teacher | * Edit quizzes * Create new quizzes * View Student stats | * May not edit student stats |
| Administrator  (Manage key parts of the program) | * Edit and delete users * View access log * Can edit, create and delete quizzes | * No limitations |

SOFTWARE DIAGRAM



DATA DICTIONARY

Text Files

**PAT INSTRUCTION**

**Text files**

Your application must use a text file(s) for input and/or output. Explain where a text file(s) can be used in your application so that it adds value to the application.

**NG’s TRANSLATION/NOTES:**

* Write a short paragraph as to how you will be using text files as a part of your software. Be descriptive.

**EXAMPLE OF TEXT FILE USAGE**

Based on the Departure and Destination locations, the program will generate an itinerary that will be printed out and stored in a text file. The text file will be named according to the user’s ticket number.

Other Data Structures

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

**PAT INSTRUCTION**

**Other data structures/Advanced programming constructs**

Your program must use a one--/two-dimensional array/an array of objects OR apply programming concepts, such as inheritance, polymorphism, overloaded methods, method binding, etc.

**NG’s TRANSLATION/NOTES:**

* An array is compulsory (1D or 2D) – you can have parallel arrays.
* Inheritance, Polymorphism, Overloaded methods, Method Binding are NOT compulsory. If you would like to know more, come and speak to me.
* You must write a short paragraph on what the array is that you will use and how you will use it.
* Draw up a table of how your array looks. This must clearly show me how many elements the array has, the data type and name of the array.

GUI DESIGN

**PAT INSTRUCTION**

Design a GUI that considers good human-computer-interface (HCI) principles, that prevents errors occurring from invalid input and that minimises the amount of information a user has to input.

Use HCI design principles and design a GUI that considers the following:

• The user – type and context

• User requirements/needs, usability

• Dialogue – must be relevant, simple and clear

• Icon usage and presentation – well selected and relevant, well placed with a clear purpose

• Colour – use and combination of colour

• Feedback – neat, clear and well presented

• Helpful error messages

• Exits – clearly marked, placed correctly

• Shortcuts

• Flow of information on the screen – top to bottom and left to right

• Sensible usage of space on the screen

Provide sample(s) of planned data capture and data entry designs (prototype screen dumps may be used but must be annotated) and of planned valid output designs.

Show the GUI design following HCI principles of interface(s), excluding introductory screens.

**NG’s TRANSLATION/NOTES:**

* I have created a table on the next page. You must complete the table for each form/tab/GUI that you have in your software.
* Copy and paste the table below one another for however many times you need it.
* Remember to be consistent across your various forms.
* Don’t forget to clearly indicate how you will be providing help, navigation (closing the form, going back to main menu etc.) as you get marks for this.
* Once again, check the rubric to make sure that you cover everything!

**GUI TEMPLATE**

|  |  |
| --- | --- |
| **Design:** | |
| **Purpose/Use:** |  |
| **Components:** |  |
| **Data Validation:** |  |
| **Additional Info:**  **(Optional)** |  |

IPO

**NG’s TRANSLATION/NOTES:**

I decided not to copy the instructions for the PAT document as it is very lengthy. Make sure you read it first before proceeding. Also look at the rubric for this section. The IPO counts a whopping 20 marks out of the possible 48 marks you can get for Phase 1. Make sure you do this section properly! **CHECK YOUR RUBRIC TOO!**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task:** | **Input** | | | | | | **Processing** | | **Output** | | |
| **General** | | | **Validation** | | |
| **Source (GUI Component)** | **Data Type** | **Format** | **What** | **Method** | **Feedback** | **What** | **How** | **What** | **Format** | **Object** |
| User Registration | Keyboard (edtName) | String | Sentence Case | Numbers/  Special Characters | Loop through and check each char. | “Error! Your name cannot contain….” | Store in variable  Preferably Pseudocode, but Delphi code & SQL accepted. | sName 🡸 edtName.Text | Confirmation Message: “Your details have been stored successfully” | Text | Dialog Box |
| Keyboard (edtSurname) | String | Sentence Case | Numbers/  Special Characters | Loop through and check each char. | “Error! Your surname cannot contain….” | Store in variable | sSurname 🡸 edtSurname.Text |
| Mouse (dtpDOB) | TDate | Regional Settings  (dd/mm/yyyy) | None  (built-in) | N/A | N/A | Store in variable | sDOB 🡸 DateToStr(dtpDOB.Date) |
| Keyboard (edtCell) | String | Digits without any spaces in between e.g.  0789419712 | Has 10 Chars | if(length(edtCell) <10) then…. | “Your cell phone number must have 10 chars” | Store in variable | sCell 🡸 edtCell.Text |
| Calculate Total Price | Mouse (spnQty) | integer | Numeric | Range (no more than 5 items) | MaxValue Property | “Error! You cannot have more than 5 items!” | Calculate Price | quantity \* price (constant) = total price | “Your total comes to <price>” | Text  Currency | Price Label |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task:** | **Input** | | | | | | **Processing** | | **Output** | | |
| **General** | | | **Validation** | | |
| **Source (GUI Component)** | **Data Type** | **Format** | **What** | **Method** | **Feedback** | **What** | **How** | **What** | **Format** | **Object** |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

Annexure A – Declaration of Authenticity

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Learner's name** | Jude Wells | **ID Number** | | 0208065349085 |
| **Grade** | 10 | **Year** | | 2022 |
| **Subject** | Information Technology | | | |
| Practical Assessment Task (PAT) | | **Teacher** | | Mr A. Alberts |
| I hereby declare that the contents of this assessment task are my own original work (except for items listed below or where there is clear acknowledgement and appropriate reference to the work of others) and have not been plagiarised, copied from someone else or previously submitted for assessment by anyone.  **List of assistance received:** | | | | |
| Nature of assistance | | | Person who provided assistance | |
|  | | |  | |
| \_\_\_ / \_\_\_ / 2022  SIGNATURE OF LEARNER DATE | | | | |