

Topic

PAT 2024 Grade 12

Learner Name and Surname

School name

Table of Contents

[Scenario and Scope: Define the task (Task 1A) (**±** 200 words) 1](#_Toc159454338)

[1. Topic 1](#_Toc159454339)

[2. Purpose of program 1](#_Toc159454340)

[3. Possible solution 1](#_Toc159454341)

[4. Scope 1](#_Toc159454342)

[User Requirements (TASK 1B) 2](#_Toc159454343)

[Use a table OR a 'use case diagram' 2](#_Toc159454344)

[Design the Database (TASK 2) 3](#_Toc159454345)

[Database Tables 3](#_Toc159454346)

[Relationship between the tables 3](#_Toc159454347)

[Data Dictionary (TASK 3A) 4](#_Toc159454348)

[Classes and Objects 4](#_Toc159454349)

[Data Dictionary (TASK 3B) 5](#_Toc159454350)

[(Text Files and Arrays) or Advanced Concepts 5](#_Toc159454351)

[ Text File 5](#_Toc159454352)

[ Arrays 5](#_Toc159454353)

[ Advanced programming constructs 5](#_Toc159454354)

[Navigation/Description of Flow Diagram (TASK 4A) 6](#_Toc159454355)

[Option 1 6](#_Toc159454356)

[Option 2 7](#_Toc159454357)

[Design the Graphical User Interface (GUI) (TASK 4B) 8](#_Toc159454358)

[Screen 1: <screen name> 8](#_Toc159454359)

[Screen 2: <screen name> 8](#_Toc159454360)

[Screen 3… 8](#_Toc159454361)

[IPO – Software Design Tool (TASK 5) 9](#_Toc159454362)

[Data Input validation (TASK 5A) 9](#_Toc159454363)

[Screen 1: Screen name 10](#_Toc159454364)

[INPUT 10](#_Toc159454365)

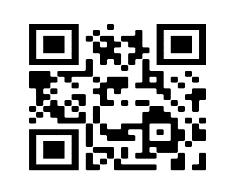
[Screen 2: Screen name 10](#_Toc159454366)

[INPUT 10](#_Toc159454367)

[Data Processing (TASK 5B) 11](#_Toc159454368)

[Data Output (TASK 5C) 12](#_Toc159454369)

Use the **headings** below in your PAT and complete it as described. The scenario and scope are often easier to do after you completed the other sections in Phase 1, but it has to be the first part of your Phase 1 document.



<https://youtu.be/dlMtjyK1m7o>

# Scenario and Scope: Define the task (Task 1A)

# (± 200 words)

Give a brief description (approximately 200 words) in your own words in terms of the problem/task and how the project will solve the problem. Use the subheadings in the numbered list (numbers 1 to 4) below for the description of the scenario and score.

## Topic

(Describe the topic you chose from the examples given in the PAT document or your own related topic.)

## Purpose of program

(Describe the purpose of your program – why does the company need your program?)

## Possible solution

(What will the program do to meet the needs/solve the task/problem. Describe how your program will work. Also include a description of each of your three Tab Sheets /Screens and how the program will interact with your database and text file (if it is going to be used.)

## Scope (what the project cannot do)

# 

# User Requirements (TASK 1B)

## Use a table OR a 'use case diagram'

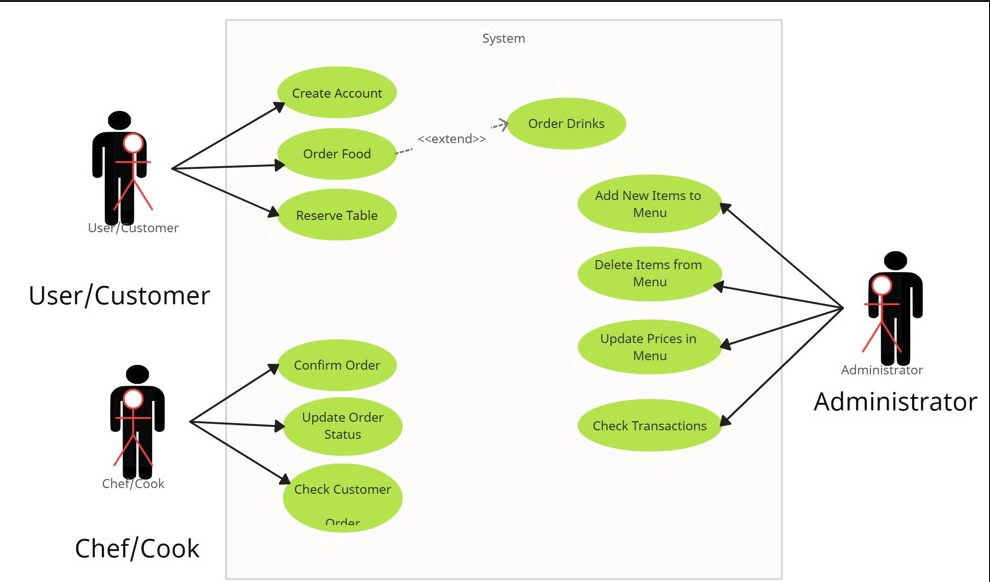
Complete the tables in detail to **clearly** describe the users (target audience) that will use your program. You need at least TWO different TYPES of users. Each of your users must use the program differently, although some of the tasks may be the same.

1. User1**:** (Give the user a "name" such as client, so it is clear who the user/users is/are)

|  |  |
| --- | --- |
| Role | What role will they perform in the program?  For example: Keeping the data in the database up to date. |
| Activity | What will they be able to do in the program? List the tasks they will be able to perform.  For example: add a new product, change the prices of products, delete a product, view the profit of the company. |
| Limitations | What can’t they do? List the tasks they will not be able to perform.  For example: Reset a user’s password or change the login report. |

1. User2: (Give the user a "name" such as administrator, so it is clear who the user/users is/are)

|  |  |
| --- | --- |
| Role | (What role will they perform in the program) |
| Activity | (What will they be able to do) |
| Limitations | (What can’t they do) |



|  |  |  |  |
| --- | --- | --- | --- |
| Task | Administrator | Staff | User |
| Log in to the system | Yes | Yes | Yes |
| Add a new User | Yes | Yes | No |
| Delete a User | Yes | No | No |
| Match the User with a property by allowing scrolling through the property pictures | No | No | Yes |

# Design the Database (TASK 2)

Show the design of the database, including the tables, relationships, field names, field types and field sizes.

## Database Tables

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Field Name*** | ***Field Type*** | ***Field Size*** |
|  | field\_name | Text/Integer/ Boolean/Currency/  Real | Max\_Size |
| PK | Example: Username | Text | 25 |
|  |  |  |  |
|  |  |  |  |

*Table Name: tblXXXXXX*

*First column PK for primary key fields and FK for foreign key fields*

*Table Name: tblYYYYYYY*

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Field Name*** | ***Data Type*** | ***Field Size*** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Relationship between the tables



<https://youtu.be/ZWEPU019Nmw>

Relationship Description: *<Explain the link between relational tables>*

tblStudents

tblCourses

tblLitter

tblLitterCollected

take

has

Example of the tables:

|  |  |  |
| --- | --- | --- |
| **Parent table name** |  | **Child table name** |
| Primary Key Field (PK)**1** |  | Primary Key Field (PK) |
| Field |  | Field |
| Field |  | Field |
| Field |  | Field |
| Field |  | Field |
| Field |  | **∞**Foreign Key Field (FK) |

# Data Dictionary (TASK 3A)

## Classes and Objects

* Application must contain at least ONE object class.
* How is the class going to be used?
* ***Class description and class diagram*:**
* **Class diagram** Use UML here. NOTE: UMLs are sometimes asked in **theory exams**, this is your time to practice.
* **Class description**: **Clearly explain** the purpose of your class in your program and how it will be used to add value to your program.
* Explain where objects can be used in your program application so that it adds value to your application

|  |
| --- |
| **Class Name:** TUser |
| ***Attributes*** |
| * fFirstname : string * field\_name : data\_type * field\_name : data\_type |
| ***Methods*** |
| + Constructor Create ( parameters : data\_type )  + function FuncName : return\_data\_type //Explain function  + procedure ProcName ( paramters : data\_type ) //Explain procedure |

- 🡪 private

+ 🡪 public

# Data Dictionary (TASK 3B)

## Text Files and (Arrays or Advanced Concepts)

### Text File

Use text file(s) for input and/or output. Explain the following:

* + - **Purpose:** Describe what you will be using it for in your program.   
      HINT: text files are used to store data.
    - **Describe** how & when the program will:
      * **read** from the file.
      * **write** to the text file.
    - **Format*:*** Include the format of the text file. For Example: **<Name>#<Surname>#<DOB>**

**NOTE**: You may **NOT** place data from your database in your text file. You can use one field in the file, to link a field of a table, to be able to extract data from the text file, according to data from your database.

For Example: If one of your tables in your database contains a primary key for each client, you can use that value in the text file and next to it place extra information about the client that does not exist in your database.

* + - **Extract of data:** Place three lines as an example of your text file here.

**I’m not sure if this is right, but I try to use a Text File for “NON ESSENTIAL” things that would not need to be stored on the database**

**e.g. Audit Trail file, storing a record of ALL SQL transactions done on the database (This is quite an easy thing to do)**

**A Log in Trail file, while stores the USERNAME#REF#Date#Time#UserType(e.g. Admin) every time someone logs in. This file can be filtered to show e.g. all people who logged in after midnight**

**Complaints/Suggestions Text file. Every time the user wants to send a complaint / suggestion, this can be stored in one long text file that the Admin could browse through**

### Arrays

Your application must use a one-/two-dimensional array.

If you have a Login File, you could make a LOG Object, and then read the text file into an array of these Log Objects. They can now be sorted and Filtered to produce stats of the people who log in, and the times etc

OR

### Advanced programming constructs

Must apply programming concepts such as inheritance, polymorphism, overloaded methods, method binding, array of object etc.

# Navigation/Description of Flow Diagram (TASK 4A)

Clearly indicate the logical program flow and/or navigation between screens. Use a UI flow diagram or any other form of illustration to present a global overview of the project/system navigation.

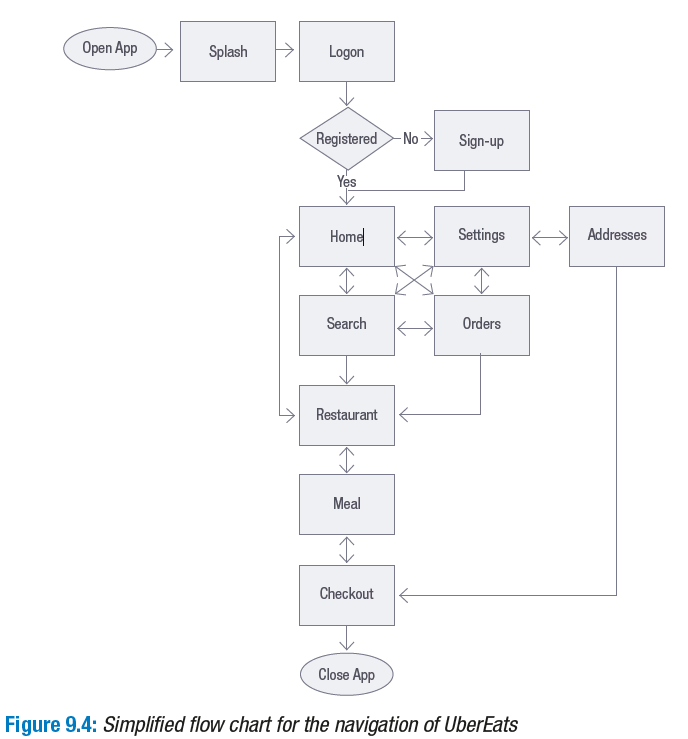
Do the following:

* + - Create a *flowchart* of your program.
    - Do **NOT** include the help and reset buttons in your flowchart.
    - Go to **draw.io** website to create your Flowchart.
    - Watch the video on Dandel10n Delphi YouTube channel in the PAT paylist named: **PAT: Flowcharts on draw.io**

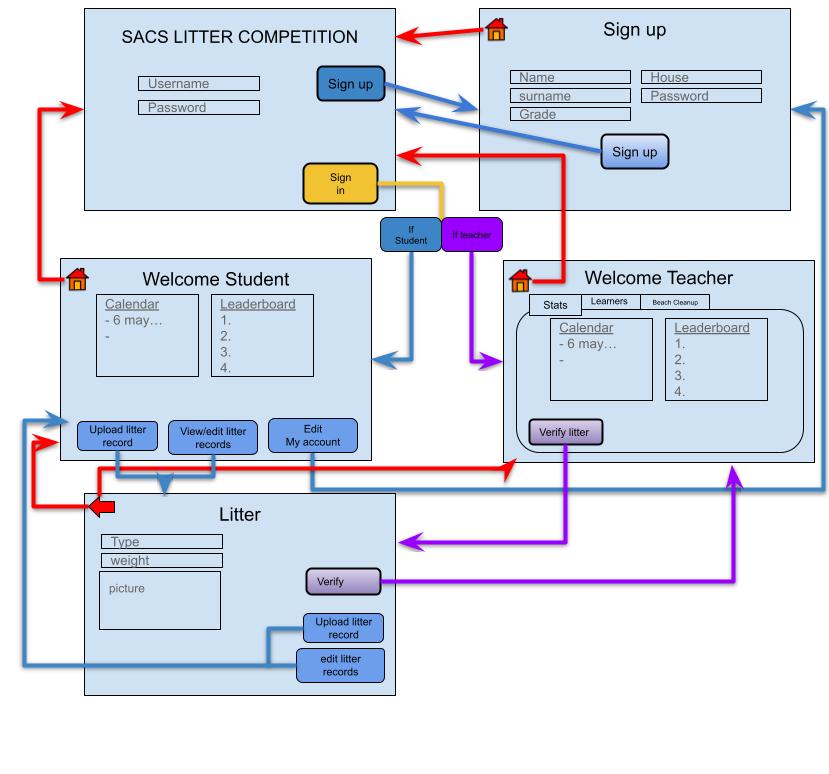


<https://youtu.be/eNRe5Ofrbgo>

## Option 1



## Option 2

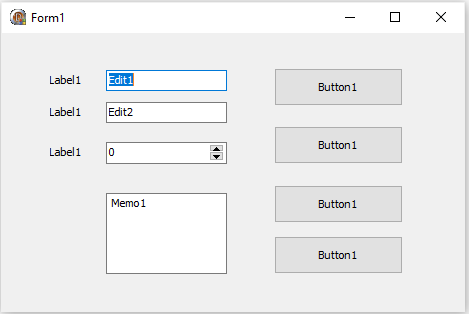


# Design the Graphical User Interface (GUI) (TASK 4B)

## Screen 1: <screen name>



<https://youtu.be/16Qb8ZOPMIA>



## Screen 2: <screen name>

#### <Paste screenshot here>

## Screen 3…

#### <Paste screenshot here>

# IPO – Software Design Tool (TASK 5)

## Data Input validation (TASK 5A)

Input interfaces and validation.

You need to validate at least

* + - 4 different data types
    - At least FOUR inputs validated, including:

- Validate for NULL/empty field AND

- Test if value was selected in a selection component

* + - Associated error messages

For example:



<https://youtu.be/5fjW_AtGGXM>

* Test if an Edit has been left empty.
* Test if the user selected something from the ComboBox / RadioGroup / ListBox.
* Use Try .. catch StrToFloat for real input (real data type).
* Test the length of a string entered (string data type).
* Test if an integer number is in a certain range. (Integer data type).
* Test if the file exists (Text file data type).

For the *DateTimePicker*, you can test if they selected a date before or after a certain date depending on your scenario. This will give you another data type check should you need one.

Specify the format, data types, source of input, validation of input and error checking mechanisms of at least TWO INPUT interfaces.

Use the following format:

### Screen 1: Screen name

### INPUT

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Input*** | ***Source*** | ***Data Type*** | ***Format*** | ***GUI Component*** | ***Validation*** |
| Example: sUsername | Keyboard  *Other options: text file, database, array* | String | General text | edtLogInUsername | *<Explain validation process>*  Error Message:  *<Text displayed in error message>* (mtError)  Message type if using a messageDlg |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### Screen 2: Screen name

### INPUT

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Input*** | ***Source*** | ***Data Type*** | ***Format*** | ***GUI Component*** | ***Validation*** |
| Example:  dateofBirth | Keyboard  *Other options: text file, database, array* | Date | dd/mm/yyyy | dtpDateofBirth | *n/a* |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Data Processing (TASK 5B)

**WHAT** and **HOW** the processing will need to be done

Specify the processing that needs to be done and provide algorithm(s)/formulae to show how the processing will be done.

You need to **list** a total of **8** processes in your program (You need to write **algorithms** for **4** of these processes. Ensure that the FOUR algorithms you write have **significant** processing. Reading from the text files, string handling, your array manipulation planned earlier, or Database manipulation using Delphi code, should provide you with the proper algorithms to write.



<https://youtu.be/oakEThuYQ6Y>

**NOTE:** (SQLs are NOT considered as algorithms).

PROCESSING

|  |  |
| --- | --- |
| **What processing needs to be done** | **How processing will be done** |
| *<List AT LEAST EIGHT processes (****brief explanation) over 2 (or more) IPO tables>*** | *<Give Pseudo code or example code/algorithm of FOUR out of the EIGHT processes listed>* |
|  |  |
|  |  |
|  |  |



<https://youtu.be/oakEThuYQ6Y>

## Data Output (TASK 5C)

Provide a clear description to indicate the output requirements of the system for at least TWO of the main interfaces.

You need to describe **all** the output of **TWO** of your screens / Tab Sheetswith **SIGNIFICANT** output. Create a table as below to describe 2 of your screens / your Tab Sheets.

**NOTE**: when you describe the output in the DB grid, make sure you add the field names and format of the output of everything that will be displayed.

|  |  |  |
| --- | --- | --- |
| **Name of Screen 1 / tab sheet** | | |
| **Output** | **Format (type, size)** | **Output Component** |
| Cost | You owe: <Cost as a Currency, with two decimal places> | Label |
| Name, Surname and Total Cost | In neat columns with the headings Name, Surname and Cost. Cost will be displayed in Rand currency, with two decimal places. | RichEdit |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Name of Screen 2 / tab sheet** | | |
| **Output** | **Format (type, size)** | **Output Component** |
| Cost | The label shows "You owe the following" and the edit displays the value in Rand Currency | Labelled edit |
| Date of Sale | Date – dd/mm/yyyy | RichEdit |
|  |  |  |

**Resources**:

* Text to Speech by Mr Long: <https://youtu.be/TqCZUioQDeg>
* Add a video by Mr Long: <https://youtu.be/lbPIwgzhWCE>
* All QR codes extracted from: ***PAT 2023*** by Lilané le Grange & Karen Andersen
* <http://www.delphibasics.co.uk/>
* <http://festra.com/>