**What are the project’s requirements?**

Phase 3 Final Document

Professional Practice and Employability, Team Based Software Development Workshop Project

Lewis Impey, Jacob Beynon, Nisa Shahril, Ryan Murunga, Shantanu Talukder

Program

The end requirement of this project is to design and create a GUI application and system in Python that has been requested by our employers. Our group has been tasked with the creation of the Personal Information Management section of the system which will allow users to access a database with information on employees of the company, with the option to add, search for, and delete employees from the database if the correct permissions are in use.

Documentation

A quality assurance document is required to be created for the project that will outline the role designation of each of the members of the group, along with a detailed description of how the application will be designed and installed, including information on the chosen database to use for the program, with reasoning why.

A GUI Design is then also required, which will be simple mark-ups of how the actual design of the program will look before starting work on it to help when it comes to the implementation, along with a way to discuss between the group how it should look beforehand.

An LSEPI should also be created to outline all of the ethical and legal aspects of the program which will then be embedded into the log-in page where the user will need to accept the terms and conditions before logging in, to make sure that the company is correctly protected in the event something were to happen where the security of the program was breached.

A software vulnerabilities document is also required to be made, which should outline a few researched software vulnerabilities that could potentially affect the program that is being made, with ways to prevent these software vulnerabilities also provided. Trustworthy sources are also required to be documented to maintain the integrity of the researched information.

A testing document is also required, where the group members should detail numerous tests that have been carried out on the program to see what does and does not work, along with follow up tests to ensure that things that did not work previously that have been fixed have been documented.

Finally, a team portfolio is required to be kept where the person in charge of recording the data will rotate throughout the project allowing every member a chance of overseeing the portfolio writing. Evidence of work being done and collaboration/communication between the group members should also be recorded within this document.

Group Member Communication and Personal Portfolio

Alongside the completion of the program and documentation, the group is also required to provide evidence of communication between the members of the group all throughout the project to ensure that a consistent line of communication has been kept and that the group members have been working alongside each other and that every member of the group has made their contribution. All group members are also required to keep a portfolio where they will detail (with evidence) all of the work that they have done towards the project. Here they should also detail any problems/obstacles that they ran into during the project, with evidence.

Phase 1,2 and 3 Seminars

Another requirement for this project is that for all 3 phases of this project, a seminar should be made (about programming languages or updates to show others the progress on the program). These seminars should be created in collaboration with all the group members and should include all the needed information along with audio recordings from group members with explanations where necessary. The design of the seminars should also be simple yet intriguing for the people watching.

**Use Case for the Program**

1. Program is run and log-in form is opened.
   1. User inputs their username and password to log-in.
      1. Input is checked for SQL injection.
         1. Input is invalid (SQL injection found.
            1. Invalid input message is displayed to the user.
         2. Input is valid.
            1. User reads the terms and conditions and accepts them.
            2. Username and password are checked and verified.
            3. User is logged in.
   2. Main menu form is opened.
      1. User decides to exit the program (program is closed).
      2. User selects personnel records from the drop-down menu.
   3. Personnel records form is opened.
      1. User selects the ‘Add employee’ option.
         1. System checks that user has the permissions to do this action.
            1. User does not have sufficient permissions (nothing is done).
            2. User does have sufficient permissions.

The ‘Add Employee’ form is opened.

User closes the ‘Add Employee’ form

User inputs information required to add new employee and input is checked for SQL injection.

Input is invalid (SQL injection found).

Invalid input message is displayed.

Input is valid.

The new employee is added to the database.

The ‘Add Employee’ form is closed.

* + 1. User selects the ‘Delete Employee’ option.
       1. System checks that the user has the permissions to do this action.
          1. User does not have sufficient permissions (nothing is done).
          2. User does have sufficient permissions.

Selected employee is deleted from the database.

* + 1. User uses the ‘Search’ feature on the database.
       1. User inputs employee ID of the employee they are looking for information on.
          1. Search is unsuccessful (invalid ID input).
          2. Search is successful.

All saved information about that employee is displayed.

* + 1. User exits from the Personnel records form (form is closed and user is returned to the main menu form.)

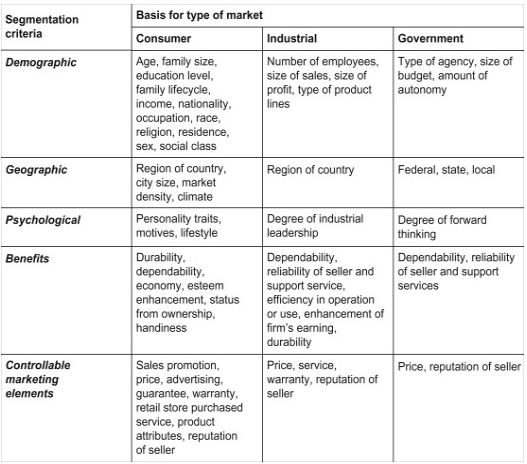
1. User closes the log-in form, closing the program.

**Project Constraints – Nisa**

**Target Market for the product**

The team’s marketing strategy will be based on providing the right service available to the right target customer. One main intention that we hold is to provide the right service to Ufix Ltd. in storing safely employee’s personal information and keeping up to date with all the information, in order to benefit from the advantages offered by our Personal Information Management System. We have considered security-wise by giving access only to the admin of the system.

The process of market segmentation is extremely crucial in identifying a target market. There are a few basic segmentation criteria can be effectively used to define our target market amongst the three overall market which can be consumer, industrial or government. Based on the table below, our product could extend its purposes in serving other target market. We realized that our product can also be targeted to the Industrial Market or the Government Market which gives us new opportunity for our company, Brainvire Ltd.



**Out of scope of end products**

As a team, we have outlined parts of our product that is out of scope. Out of scope is work that is beyond the current agreed scope of the product. There are a few points that were out of scope of our product that we have included:

1. We have included additional packages to the product in order to output pictures on the interface of the system. Images are important to give visualization on important or less important information to the end user which in this case, the admin.
2. We have also considered having readable sized font for our system that could benefit all ages of admin that will be using our product.
3. We have also considered to give a cool colour shade of blue to the interface to provide ease on the eyes when reading through all the information. This colour also matches our group colour that has been used in all our previous seminars as a point of professionalism and identity.

A project has three types of baselines which are scope, schedule and cost. Usually, changes to original scope require adjustments to schedule and cost. However, the ‘out of scope’ points stated above was not in need of further adjustments. The product that was developed were finished ahead of schedule and there is no additional cost added. This does not give us the leniency to not keep track of any further out of scope. We must always reflect against the original scope in order to create a reasonable product and taking consideration of the main scope.



**Implementation of Program – Jacob**

We have coded a Personal management system that allows the logged in user to access a database which contains the employees’ personal details. This section will go through some core parts of the program and how it can be implemented. There Is also a section on how to install packages due to them being vital in running our program

Packages

In order to run the program within an IDE there are some Libraries that are required for the running of the program these are:

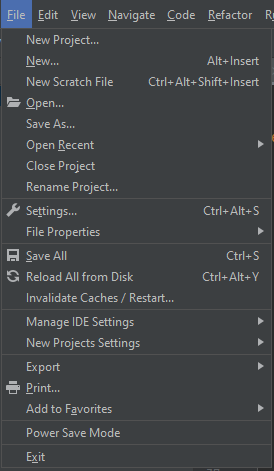
PIP – this allows packages to be installed and used within the program.

Pillow – This library is need for the viewing of any images within the program.

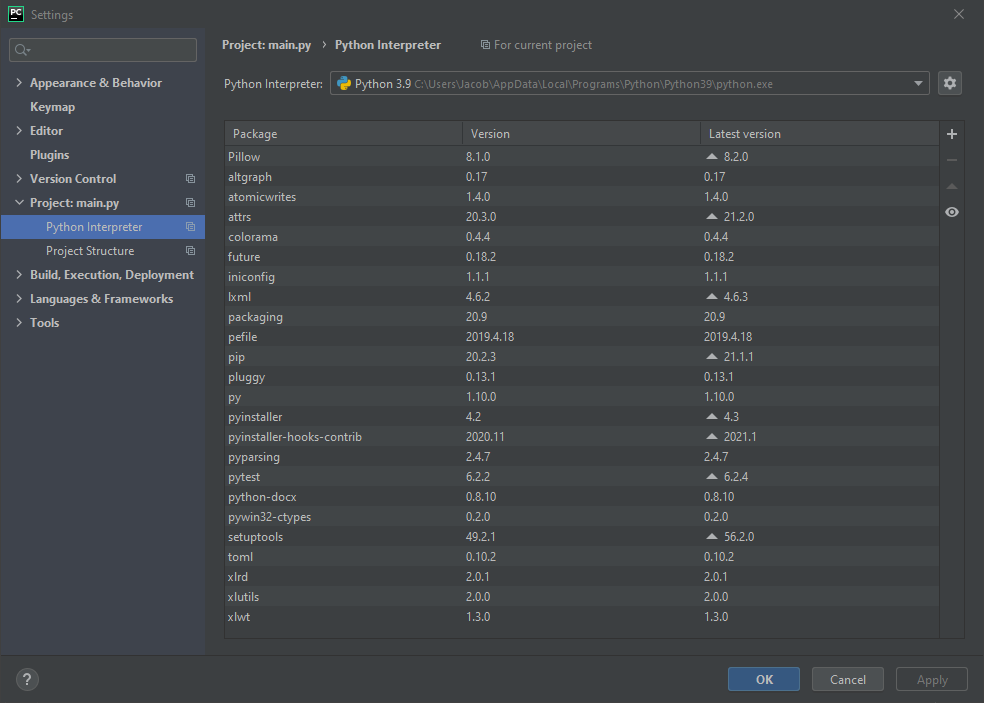
Xlrd – The program requires this to access the customers data if there is no database present.

Here is how to install a package when using PyCharm as an IDE:

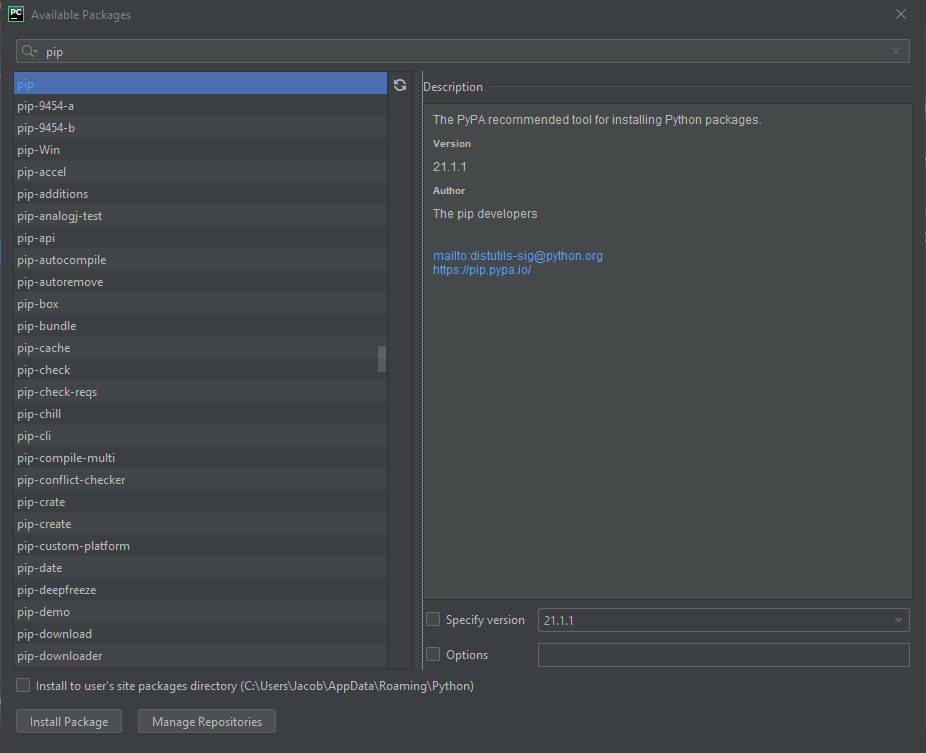
Step 1) go to file then settings.



Step 2) in settings go to Python interpreter. Then press on the + next to Latest version.

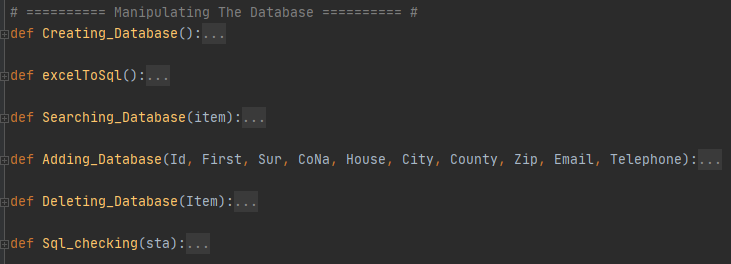


Step 3)A window will pop up, search for the library that you want and press install.



Once you have done this it will pop up in your installed packages area. For our program to work the three listed packages are required.

The program makes uses of many functions and classes. Each function has its appropriate name to it so there easier to understand. The first set of functions are all to do with the SQLite databases that have been given to the program.



**Creating\_Database()** – this function is ran when running the program and is what checks to see if the two databases that we will be accessing are there if they are not then it creates them and populates then with data.

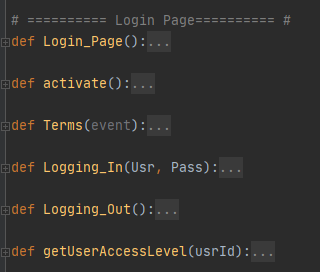
**excelToSql()** – this is only used if there is no accounts database in where it takes a excel document and converts it to be SQLite 3 compatible.

**Adding\_Database()** – Here is where we add information into the accounts database. Its needs data passed into it to add it to the database. If this works successfully then the data is passed into the database which updates the main viewer with the current data.

**Deleting\_Database() –** Here is where we delete the selected user. This requires that an option has been selected and that the current logged in user has admin level access.

**Sql\_checking()** – when a user tried to access the databases this function gets called checking if the user is trying to enter in some sql specific characters. This prevents sql injection within the program.

These next functions are all to do with logging into the program.



**Login\_Page()** – This is for the creation of the log in page, here we defines all the buttons labels and which functions will be called when the buttons get pressed

**Activate()** – This is used for showing the login button.

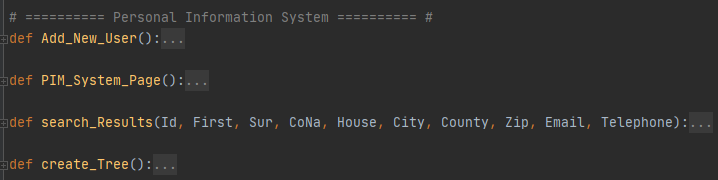
**Terms()** – This will be called when the user wants to view the terms and service as it displays them in a pdf document.

**Logging\_In()** – Here we check the username and password entered by the user are within our user database if it is then we allow the user to be logged in and we get there access Level of the logged in user

**Logging\_Out() –** Here we just remove the menu options associated with being logged in

**getUserAccessLevel() –** This just gets the users access Level.

Theses functions are for the Main PIM system:



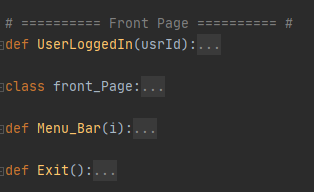
**Add\_New\_User()** – Here we take the information inputted into the text boxes and add them to the database which also updates the database viewer as well

**PIM\_System\_Page() –** This is the creation of the PIM page which displays all the options. Labels and data grid viewer needed for the page

**Search\_Results()** – Here we get the search results from the Searching textbox and we display that in the search area within the Pim System

**Create\_Tree()** – This will be called when every we need to re construct the viewer of the data base normally after we add or remove a account.

Lastly here are the functions for the front page:



**UserLoggedIn() –** All this function gets used for is created the Label that displays the current logged in user.

**Font\_Page –** This a class which creats the main page displaying all of the menue options and images neeed.

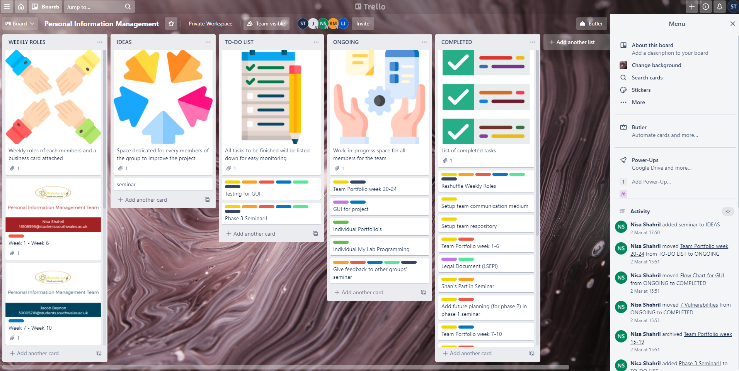
**Menu\_Bar() –** This takes in a number passed into it which determines the options displayed to the user

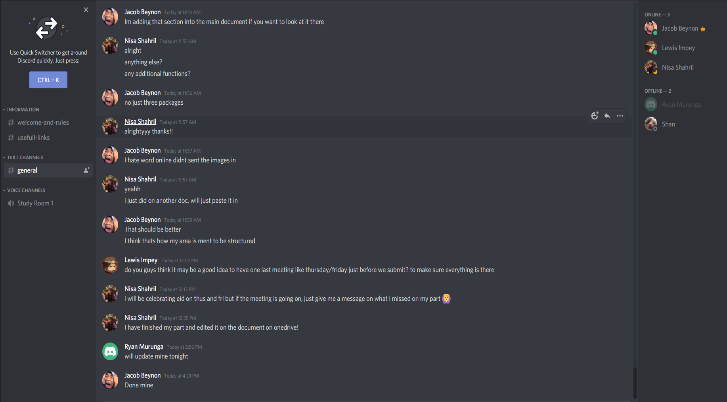
**Exit() –** Used when the program is being closed down

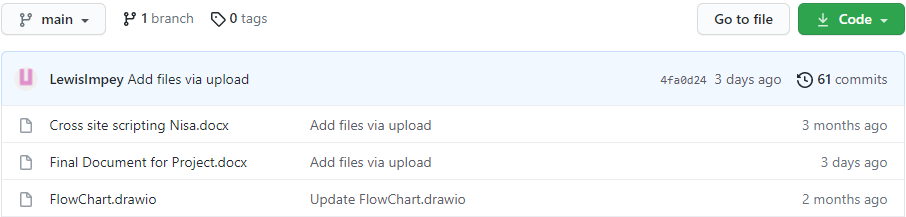
**Components of Project – Shan**

The team has used several products to aid us with our project. Some of them were essential while others were for quality-of-life improvements and streamlining the development process.

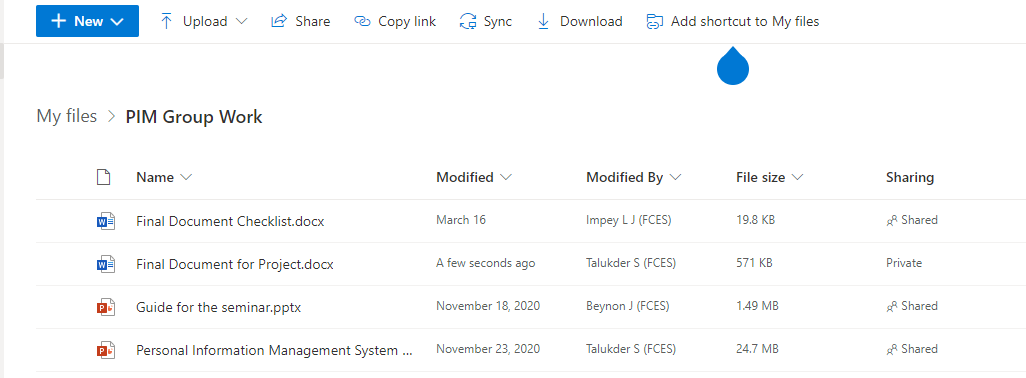
The first product we have used was PyCharm. As the group used Python as our coding language, we have decided to use a better IDE than the default one. PyCharm has many different plugins and libraries available, and it also comes with a built-in dictionary. Additionally, it shows the user which variable, function, classes are called and when. Not to mention that it also has better error handling than the default Python IDE.

The next product we have used was Trello. Trello has been a great help to us in keeping track of our individual progresses and keeping track of our deadlines. This product was very visual friendly as everything could be colour coordinated and labelled, which made it noticeably clear who was handling which task at what time.

Another product which the team has used was Discord. It has been our primary method of communication. We have set up a server for the team where we can discuss about project during our weekly meetings or talk any time on chat. We have also made different channels in our server to keep the topic of our chats organised. Additionally, Discord also comes with features like screensharing, which allows us to better work on projects remotely. 

We have also used GitHub extensively. Making a GitHub repository has been greatly beneficial to us because we have used it for all our work except for seminars. The team has gravitated to it because of is plethora of useful features. Some features for example would be to allow everyone to upload, edit and download files. Its selling point is that it allows us to view previous versions of uploaded files. This is especially useful because in cases where newer versions of the files may contain error or bugs, we can revert it much easier. It also allows us to merge different versions of a file, which has made streamlined the process of testing new codes on the GUI. 

The last product we have used is OneDrive. It was primarily used by the team to create seminar presentations together. It has streamlined the process of working on the seminar extensively, since we were all able to work on our own parts of the seminar simultaneously on one PowerPoint without any hassle.



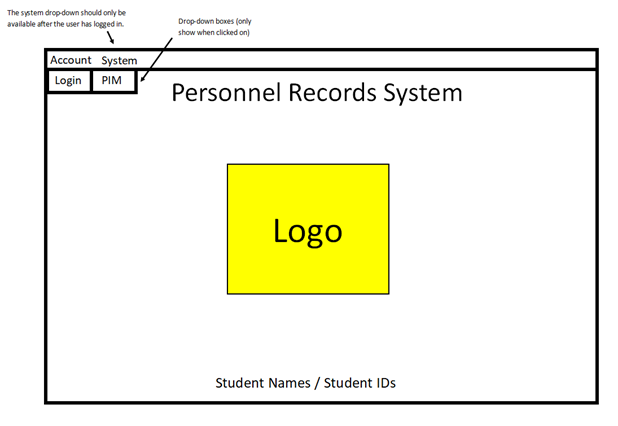
**Desired Qualities of Project – Ryan**

As per one of the requirements, a quality assurance document was created to provide a framework of the expected final program product. It is this document that would be used to mirror the actual program to the expected product, and a comparison deduced whether the expectations put were met.

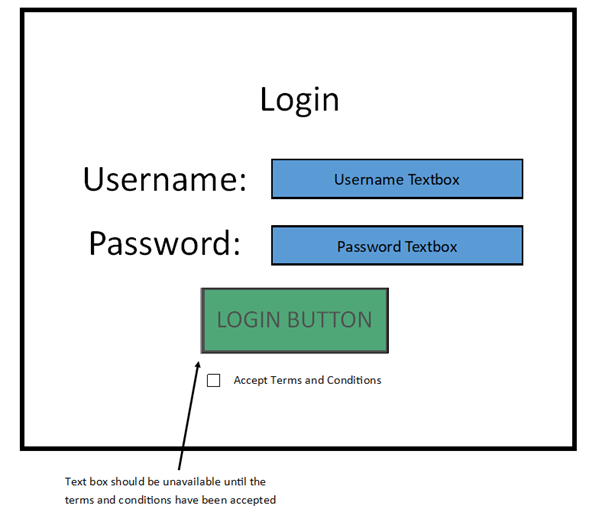
The methodology used to complete the project was Agile, each task completed as the developer tasked.

Application design patterns drawn before coding were implemented correctly and provided us with a map to work towards to.

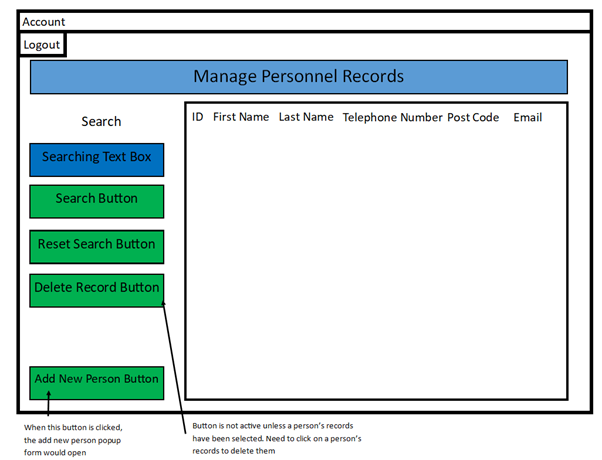
Main form:



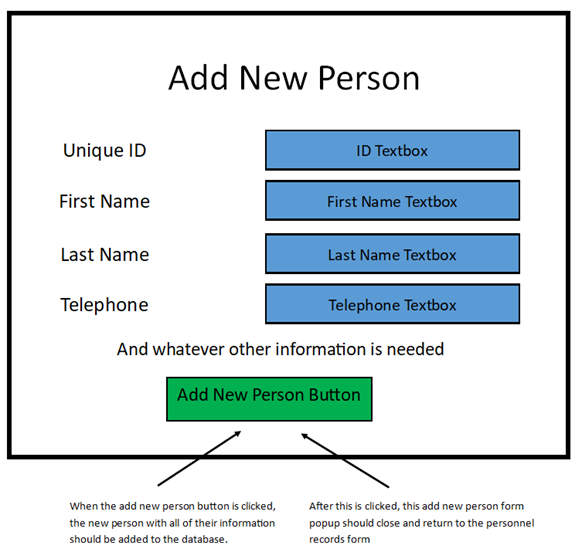
Log in form:



Personnel records form:



Add new person form:



SQLite was the database used to store the data generated by the GUI as it is free, open source and easily accessible. Its compatibility with different platforms was also considered in its implementation as the appropriate database to use in the GUI.

Text on the GUI was made big enough to read, including people that may have visual impairment problems. Font type used and colour was consistent throughout.

When the program is loaded, it is expected to show a main form with a title that states what the program is as well as showcasing the students’ names and numbers. On the top left, a drop-down box that provides the user with an option to log in or exit is present. The log in button is unavailable till the terms and conditions checkbox is ticked.

After a successful log in, the log in page closes but the main form remains open. Its drop-down box now includes an option to view the personnel records within the database. Clicking the drop-down box opens the Personnel records form that allows the user to view the existing records in the database.

A search button lets the user find a record within the database and a corresponding reset button that returns the form to its original order. A delete button that is unavailable till the user clicks on an intended record removes the selected from the database.

The button Add popups a new form that allows the user to add new records in the database. When the relevant details are added in the textboxes available, the form “Add New Person” closes and the database is updated with the new records.

If the user wishes to leave the program, an option to do is available in the top left side of the GUI. Logging out closes the Personnel records form and presents the main form again. The drop down gives the user an option to leave the program or log in again.

The GUI design and programming is appropriately aligned with the desired expectations of the program and greatly manifests a user-friendly interface.