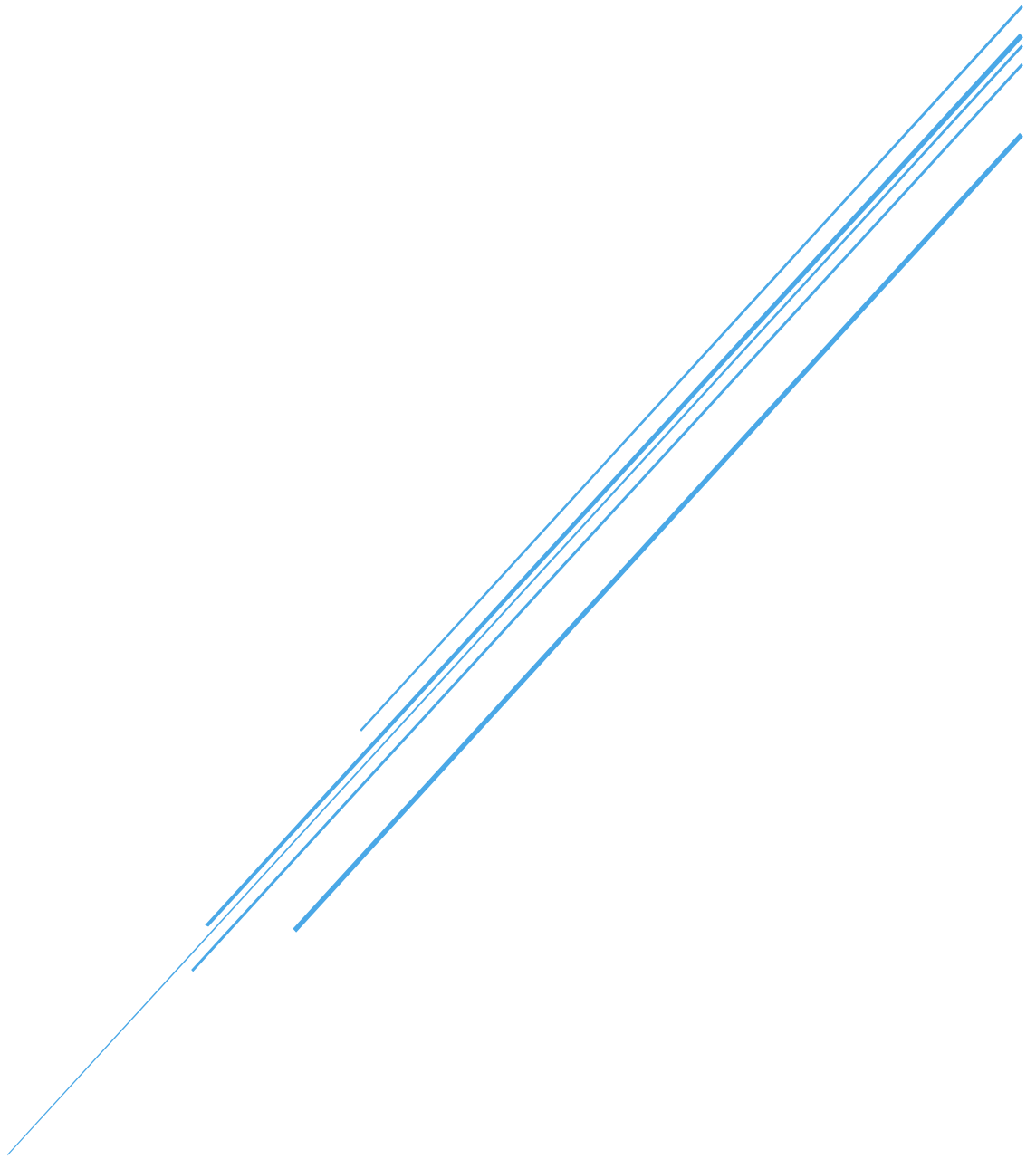


# AM RADIANTS

Police Management System

VB.net Project



ABID MIAH

## **Contents**

### Analysis

- *Background to the problem*
- *Statement of the problem / User(s)*
- *Initial Research*
- *Observation of existing systems*
- *Databases*
- *Objectives*
- *Interviewing the public with experiences with the police*

### Design

- Overall Design Flowchart
- Each form screen designs, prototypes

### Implementation

### Testing

### Evaluation

<https://1000projects.org/crime-detection-through-facial-features-project-in-vb-net.html>

## **Analysis**

### **Background to the problem**

With the rapid advancements of technology over the recent years, a number of manual systems are being replaced by modern technological methods in different fields and sectors. There has been a revolutionary change in many sectors such as teaching-learning, communication, etc. One of these sectors is crime investigation procedure, which is utilising modern equipment such as face identification software and other methods in order to manage and assist in crime management. Moreover, police corruption has been a major issue, especially in developing countries, where there has been many complaints about it by many officers who are not fulfilling the safety and standards of the public.

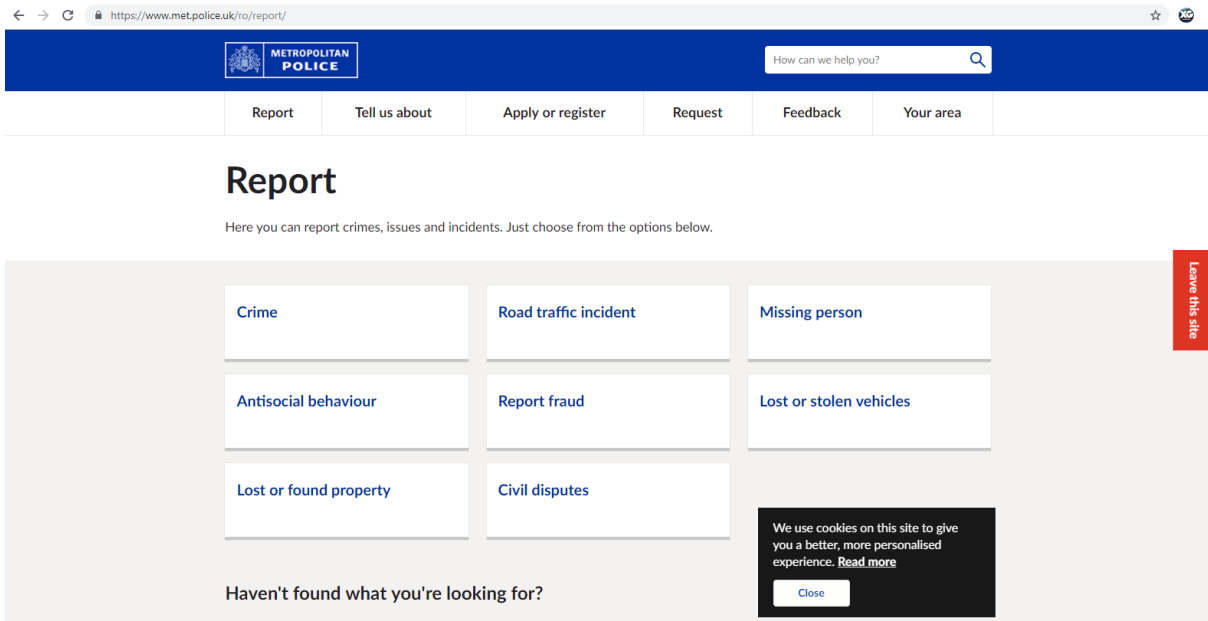
After doing some background research, I have discovered that there are some options to file a crime/report online to mainly to report law enforcement officers in order to get immediate response to violations.

Currently, the majority of these issues are generated on personal lifestyles that revolves around the use of technology constantly, in their daily lives.

Therefore, for my A2 project, this gave me the idea to create a system for civilians and police department that would allow reporting of minor crimes or reports, such as missing person online, rather than having to go into a police station or call for emergency. This would be convenient for people to be able to do this and give people an extra option to choose from.

Moreover, my system would help to manage crime details within the police department such as missing persons, stolen vehicle and information regarding it. It

By using my program, you will be able to add and update the crime information with the necessary information. It would be also useful to search information based on the name of the person. Furthermore, it would allow authorities to view a visual crime map of crime data at specific locations in order to identify and help them decide where the main issues are. The main advantage of this project is to easily retrieve the data and prevent from loss of data. To collect a lot of information with minimal time and effort and allow the law enforcement agency to be well planned for major incidences.



*This website helps me decide on the options/menus that the police officers can choose from once logged in, to implement within my project.*



This image also helped me decide on how to login screen would look like, I would also implement a civilian sign up and login system too.

### **Statement of problem**

I am thinking about designing a program that will propose a system that applies to all police stations across the country and specifically looks into the subject of crime records management as well as the ability for law enforcement officers to map crimes. It is well understood that Crime Prevention, Detection and Conviction of criminals depend on a highly responsive backbone of Information Management. The efficiency of the police function and the effectiveness with which tackles crime depend on what quality of information it can derive from its existing record and how fast it can have access to it.

The ability that would have two login options, one for civilians and one for the police agents. Within the civilian login, they will be able to file reports and complaints. File crimes etc. Moreover, within the police login, it will show many categories such as suspects, criminals for different crimes. Missing people, lost/stolen vehicles etc. and the ability to search criminals to find the locations as well as be able to view recent crimes on a map, search locations and density hotspots.

My program would be an application that allows all the paper work that are done in the police station to be stored in one single system with great ease. It can help in handling the records easier. In addition to this, my system would help police officers to efficiently detect and prevent crime where necessary.

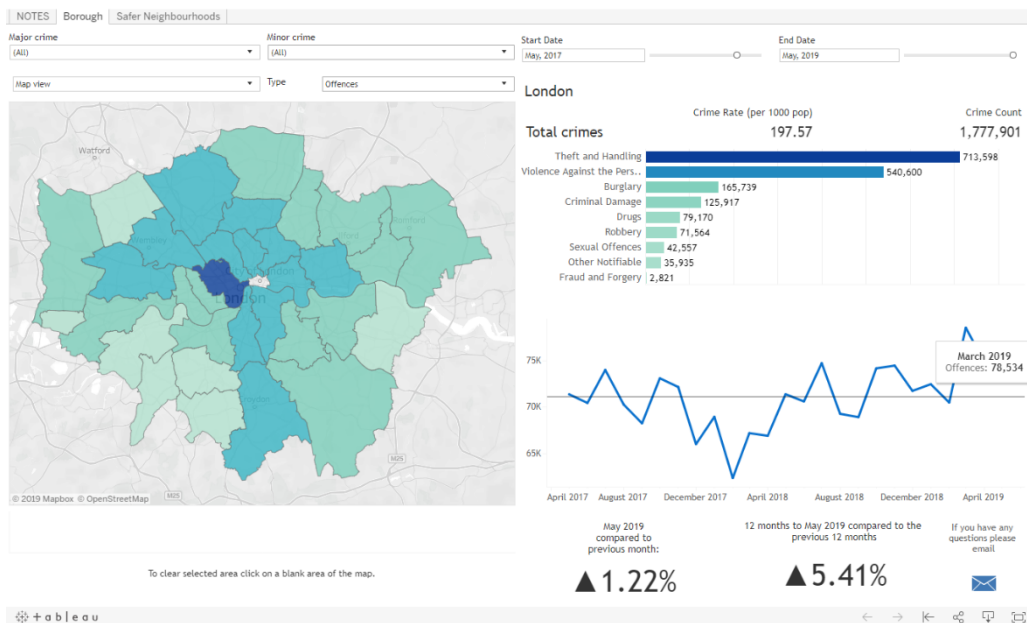
Sometimes if there is a need for some information of a particular criminal then you need to search in huge heap of files where that information is stored in those files. So that will involve in wastage of time and effort. Therefore, to overcome these problems, the police information system can be useful. This will reduce a lot of pen paper work that is involved.

### **User(s)**

The potential users of my program would be various police station / branches and access to civilians of all ages via logging into their own portal. However, for this particular program, it will be based on one specific police department and officials, which I create. This will help many the public and the population to fight down police misjudgement as well as reporting crime with ease of technology. Furthermore, it would also allow sanctioned police departments to keep organised and allow them to view reports and criminals with simplicity making it more efficient.

## Initial Research / Observation of Existing Systems

I observed the metropolitan police database to gather a better understanding of how they use certain equipment to interact and help with crime fighting on a daily basis as well as gathering potential information of each borough and identify the problems they face while doing this.



The crime dashboard that the met police has, has been created to help show the public what crime has been committed in different neighbourhoods across the capital.

This was quite useful as for my project I would show the public the issues we are facing in this generation, making them aware of possible threats they may encounter, and therefore giving the public a way of searching out for help as well as reporting using my program.

NOTES

**Home Office Crime Categories**

Change note : w.e.f. April 2018, the MPS Crime Dashboard showing the new Home Office Crime Categories can be found here:  
[https://public.tableau.com/views/MonthlyCrimeDataNewCats/Boroughs?:embed=y&:display\\_count=yes&publish=yes](https://public.tableau.com/views/MonthlyCrimeDataNewCats/Boroughs?:embed=y&:display_count=yes&publish=yes)

**Home Office Recording Rule Change: Burglary**

On 1 April 2017 the Home Office recording rules for burglary changed. Instead of the "Minor Crime" categories of 'Burglary Dwelling' and 'Burglary in a Building Other Than a Dwelling', those categories are now 'Residential Burglary' and 'Burglary - Business and Community'.

The main change relates to how crimes where property is taken from sheds and garages are recorded. For example, until 31 March 2017, a burglary from a shed which does not adjoin a dwelling would have been counted as a 'Burglary in a Building Other Than a Dwelling'. From 1 April 2017, if the burglary happens within the curtilage of the property it will count as a 'Residential Burglary', regardless of whether the shed/garage is attached to the dwelling.

This means that comparisons of residential and non-residential burglary volumes after 1 April 2017 to crime recorded prior to that date should not be made.

Further details can be found on the Home Office Counting Rules website here:  
[https://www.gov.uk/government/uploads/system/uploads/attachment\\_data/file/602803/count-burglary-apr-2017.pdf](https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/602803/count-burglary-apr-2017.pdf)

**Additional Information:**  
 Population data - G1 A. Mid-year estimates 2016 released 2017.  
 Sanctioned Detections added to Borough level data.

METROPOLITAN POLICE

Report | Tell us about | Apply or register | Request | Feedback | Your area

Home > Advice and information > Stop and search

## Your feedback matters

Being stopped and searched can be an unpleasant experience, no matter the approach of the police officers involved. But if it's felt that there has been unnecessary force or an inappropriate attitude, then we'd like to hear about it and, where necessary, act upon it. Likewise, we welcome any suggestions or positive comments you'd like to share.

### Tell us about your experience of being stopped by the police

If you feel you've had an unpleasant or unsatisfactory experience, you may wish to make a formal complaint. You can do so if you think a police officer has behaved incorrectly or unfairly. For example, if you think an officer has:

- been rude to you
- used excessive force
- abused your rights
- arrested you unlawfully

All complaints are investigated and this can take time.

To make a formal complaint and for more information about how complaints are investigated, read our [complaints](#) pages. If you've had a positive experience, you can offer us feedback [here](#).

Furthermore, I also have found a place dedicated for the public to report and share about experiences of being stopped by the police. The involvement and empowerment of communities and the public is essential to the success of policing in the country and across the world.

Which I had interest into implementing a civilian's portal where they can report complaints as well.

New Zealand POLICE  
Nga Pirihimana O Aotearoa

Safer Communities Together Kaupapa whai Oranga mō te iti me te rahi

Home | Major Events | Contact Us | Jobs | News | Ten 7 | Do It Online | About Us | Local Police | Services | Reports, Policies, Stats | Missing Persons | Stolen vehicles | Safety Tips | FAQs | Maori

### Stolen Vehicles

Check to see if a vehicle is listed as stolen.

Enter the vehicle's registration number:

Or, enter VIN, Engine or Chassis No:

Use \* for wildcard searches eg. mg\*thg\*46Z. Limit of 5 results

### Download List

Download a file of stolen vehicles from the past 6 months:

- All of New Zealand
- or by individual Police district(s):
- Auckland City
- Bay of Plenty
- Canterbury
- Central
- Counties/Manukau
- Eastern
- Northland
- Southern
- Tasman
- Waikato
- Wairarapa
- Wellington

**Limitations of the Data**

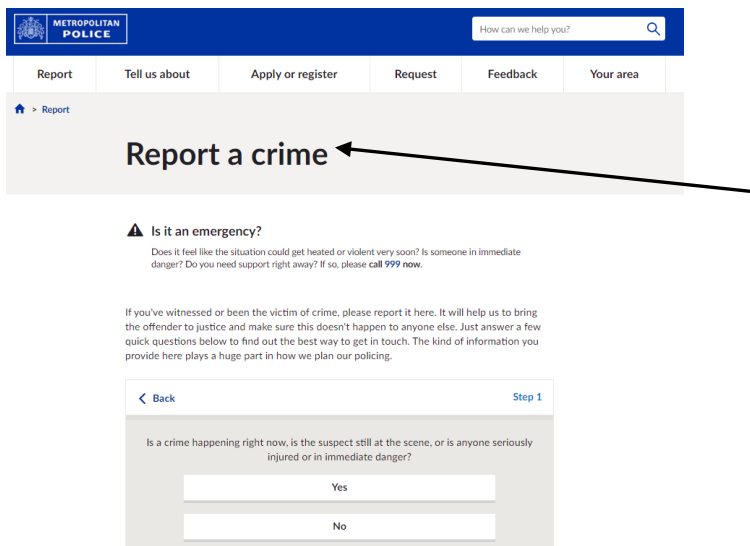
Users of this information should be aware of the following limitations:

- stolen vehicle information listed here is a snapshot of data taken from the Police Vehicle of Interest database and is updated three times per day. There can be a brief delay in stolen vehicles appearing and in recovered vehicles being cleared from the list;
- some vehicles listed as stolen will have been located but Police haven't been advised;
- this data does not include lost or stolen registration plates.

Additionally, this also is more evident of New Zealand police as stated to report stolen vehicles and some crimes online with convenience for the user and benefiting the public in positive way in these times of heavy technological use.

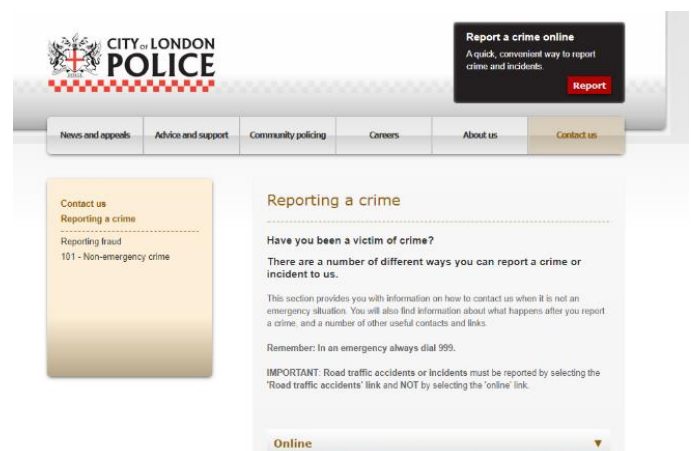


Moreover, after doing some thorough research into actual police databases, it was understandably to be confidential to view them as it would consist of missing persons, crimes, criminal, etc. which perceptibly should not be open to the public. However, I did find this, which is an idea I believe on how an actual police database would look like, clearly simplified.



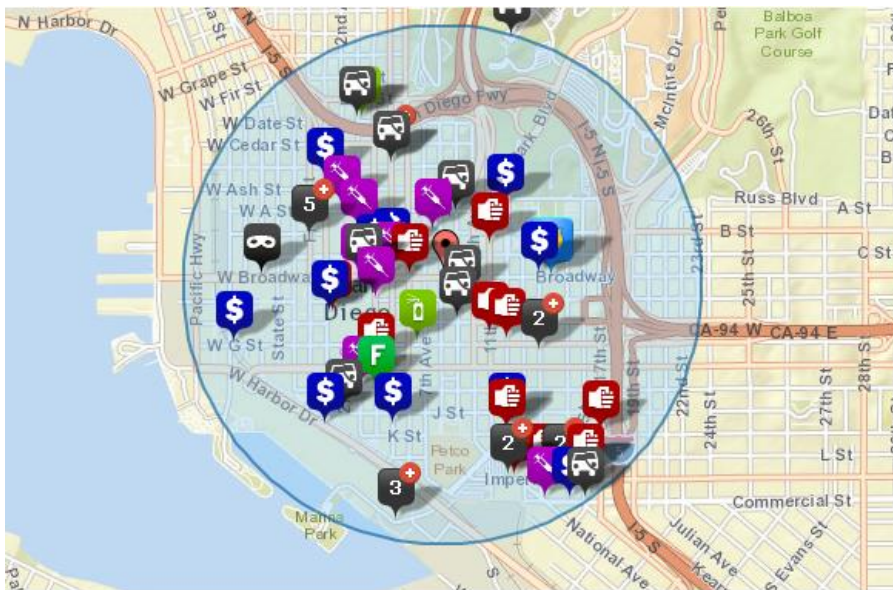
Doing some background research on existing systems, the Metropolitan Police as well as the city of London police developed a way to report crimes online if they have witnesses or even been the victim of the crime.

Adding a way to report crimes into my project would help police departments to bring the offender to justice and make sure that it does not happen to anyone else in the future



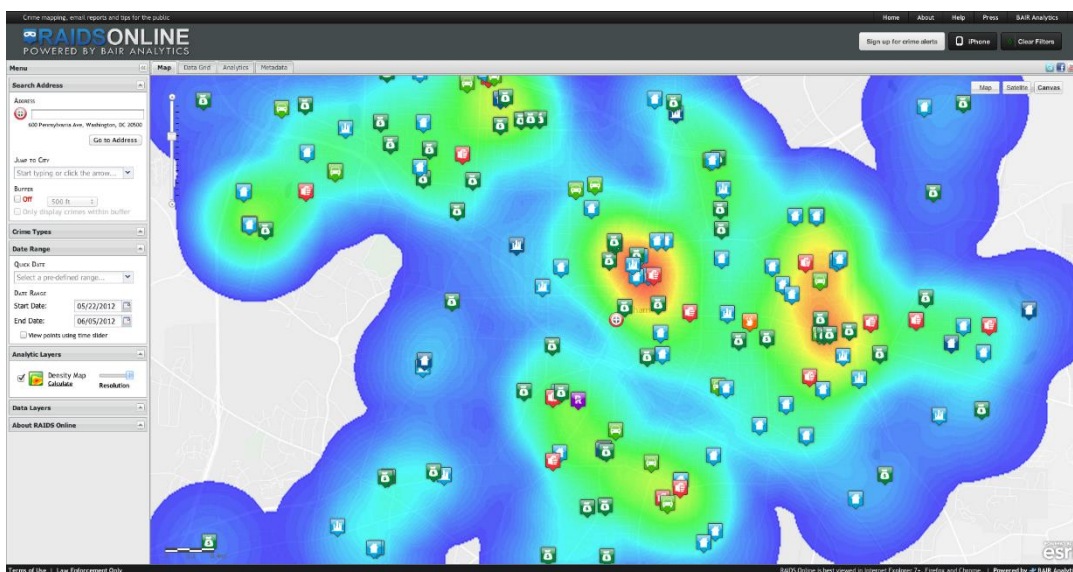


In addition to this, many police departments have crime mapping which allows them to view crime data on a visual map in order to represent and see the specific areas and location where crimes have taken place. Therefore, helping them to create density hotspots, which would allow them to compare different locations, where most crimes happen, thus enforcing the law more extensively in the specific area.



This is an example of crime mapping where there is different icons for many different crimes to be plotted and identified in a specific area.

Implementing the use of Crime Mapping into my project would be significant as it is a good tool in managing a controlling crime in an area. By analysing the spatial and temporal data provided by maps, investigators are able to understand the crime patterns and trend it. Moreover, it would also help in resource allocation and in geographic profiling of criminals and suspicious locations.



In order for the implementation of Crime Mapping, I would need to use an API within Visual Basic. Due to the Google API being reasonable costly and unfortunately not very usable within Visual Basic. I have researched some potential free online APIs in order for this to be successful integrated.

For instance:

OpenStreetMap (<https://www.openstreetmap.org/#map=5/54.910/-3.432>)

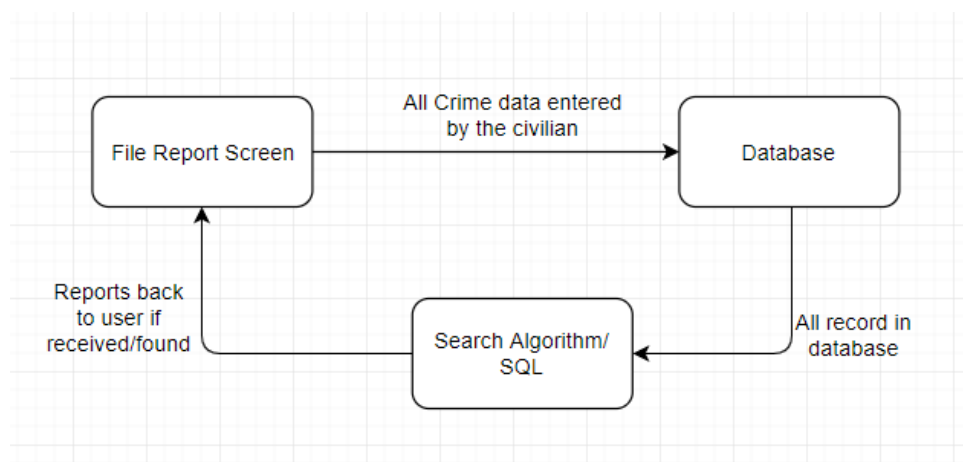
Map Window (<https://www.mapwindow.org/>)

GMap.NET (<https://www.nuget.org/packages/GMap.NET.Windows/>)

These are some potential APIs that I can use in order to implement Crime Mapping data. The ability to search coordinates/postcodes. Buffer analysis as well as a density hotspot map.

### **Exchange of data with the database**

Below is a rough overview of the nature of the exchange of data with the database.



### **Data Flow**

A graphical tool used to describe and analyse the movement of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also known as a data flow graph or a bubble chart.

## Type of data flow diagrams

Two types of **DFD's**

- a) Physical DFD
- b) Logical DFD

### **Physical DFD**

Structured analysis states that the current system should be first understood correctly. The physical DFD is the model of the current system and is used to ensure that the current system has been clearly understood. Physical DFDs show actual devices, departments, etc. involved in the current system.

### **Logical DFD**

Logical DFD is the model of proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system's structure charts/

The basic Notation used to create a DFD is as follows:

**Dataflow:** Data move in a specific direction from an origin to a destination



**Process:** People, procedures or devices that use or product data. The physical component, which is not identified.



**Source:** External source or destination of data, which may be people, programs, organisations or other entities.



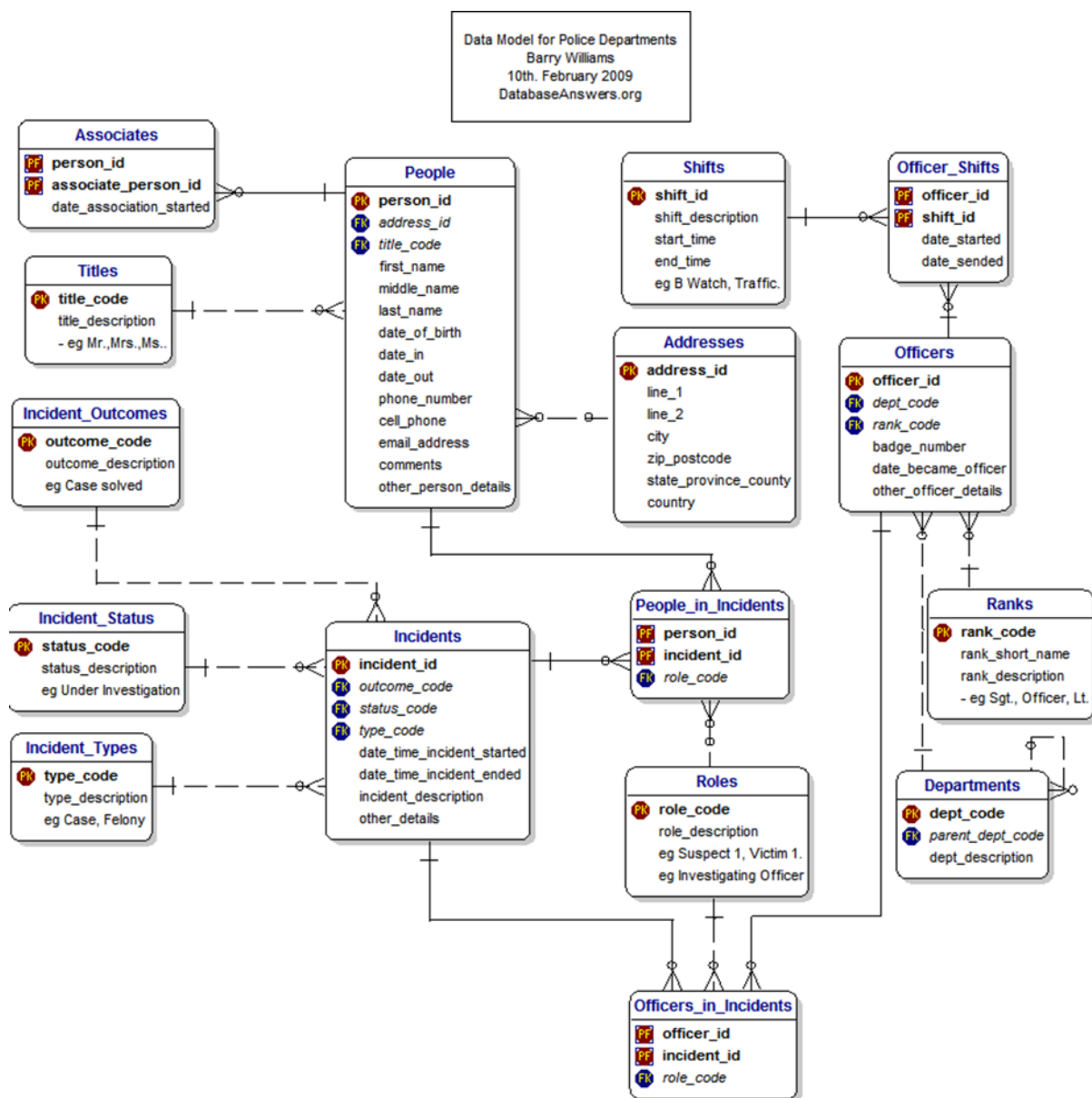
**Data Store:** Here data are stored or referenced by a process in the system.

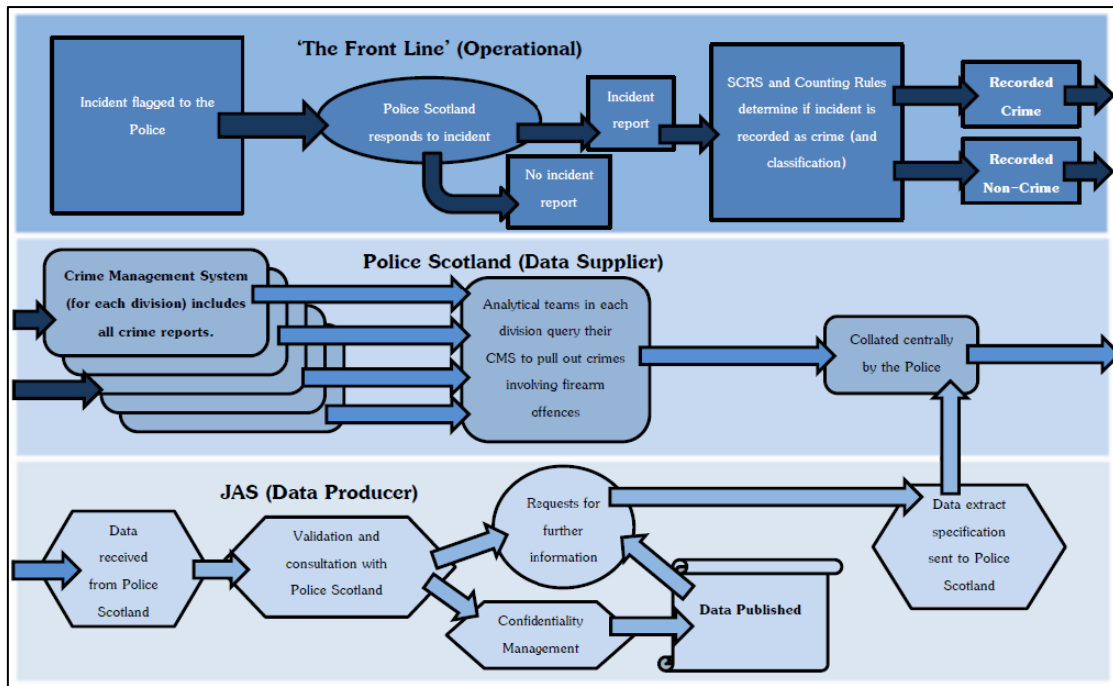


## Dataflow Diagrams

I have found a normalised database on <http://www.databaseanswers.org/> , which shows an example of a data model for police departments. It is useful as it is structured with a relational database with different groups and categories. It is beneficial and designed in a way to minimise data redundancy and improve data integrity. Therefore, meaning that it makes modifying data easier and less complicated.

It also includes primary keys, which identifies the record uniquely. For example, the text in bold are the primary keys, which the values are usually generated by an 'Auto-increment', field and don't have any intrinsic meaning and are simply integers generated automatically one after the other. Overall, there is consistent data within the database and has a much more flexible database design to handle better database security.





This flowchart shows how the police identify problems and how they respond in order to handle the incident. This centralised approach allows the police officers as well as suppliers and producer be efficient in major incidents and resolve them swiftly.

By providing usability with easy use is significant to be implemented in my project, as this would allow police as well as citizens to be comfortable with the program and know how to navigate in order for reports and response.

## ***Interviewing the public with experiences with the police***

1. In an interview with Jack Morton, a public teenager within the Merton borough, I asked how he found the police within his neighbourhood to be and any experiences he had being encountered from the police in the past.

"I live in a quiet neighbourhood it was very rare occasion for police to be around. But, there was one time when a corner shop across the road from me was vandalised and stolen from. And the police showed up and very brutally dragged the shoplifter, hitting him into the police car. As seen from my window in my block."

The key points I drew from this interview was that the police department in his area might not have been efficient in consulting the training of its law enforcement officers. Therefore, may have used the program from my project to report these types of incidences to be sorted.

2. Another interview I had with a police department in East London, they suggested that there was many paper work and in some cases hard to identify potential threats even with the use of technology and their databases they had currently.

I made them aware of my program project for a one single system to store all data with great ease. It can help in handling the records easier and identify certain criminals. They replied by suggested paper work is necessary as well to allow them to be more constraint.

After some thorough discussion with different police departments within the county. I was given the impression that my Police Management System would be most suitable for departments located in rural areas as they are limited with fewer resources and overloaded with paper. This would help them to be well managed and have an impact on the productivity as well as the efficiency of their service to the public and will also allow them to keep organised and maintained for future crises.

## **Objectives**

### **1) To enable a citizen to file various reports.**

The future is here. With this system, a citizen will be able to report various incidents from the comfort of his home. The incidents currently supported by the system include murder, burglary, missing persons, fraud and stolen items. To allow for reporting of incidents not supported by the system, a category called miscellaneous is included. This category is added because it is not possible to imagine all possible categories of crime and incidences. The developers are not criminals or, which is worse, criminologists. Future versions of the system will allow for reporting of additional incidences.

Some of the categories are expounded on below.

#### *i. Category murder*

This category will enable to citizens to report murdered persons. Details such as the name of the dead person and the location of the murder are collected and stored in the database. Police officers using the system will automatically find this report and respond appropriately.

#### *ii. Category burglary*

While it might be faster to simply dial 911, the first responders at 911 are of use only when a burglary is in progress. They are not particularly involved in the apprehension and charging of the criminals. Reporting a burglary through this system will allow for the criminals to be pursued and captured with the long arms of the law.

#### *iii. Category missing persons*

Children, and adults alike, vanish all the time. Sometimes they disappear on purpose, sometimes by accident. Either way, it is the responsibility of every citizen to file a missing person's report when a friend, relative, acquaintance, or anyone else for that matter, goes missing. The system will allow for this, and enable police officers to respond appropriately.

#### *iv. Category fraud*

It is not pleasing, to say the least, to you when someone else pretends to be you. It is even less pleasing when a fraudster pretends to be a multimillion dollar company, selling goods "from" the company, and pocketing colossal amounts of money. Such criminals deserve to be caught and persecuted and punished as appropriate. Reporting a fraud will be easy with the system thus enabling investigation of the report by police officers and detectives.



## **2) To enable a citizen to give feedback about the police system in general.**

Not everyone might like the way things are handled. There will naturally be people who want to bring attention to something they wish to be improved. Also, some people, when they like something, they want to be thankful and congratulate everyone for a job well done. Criticism, whether negative or positive, is necessary if any progress is to be made. People like to talk and be assured that they are being heard, else chaos will arise. The feedback system will allow citizens to air the views thus giving feedback to the developers and to the police officers.

## **3) To enable a citizen to view criminal reports that have been filed by other citizens.**

A citizen will be able to view reports that other citizens (including himself) have filed. For instance, if I have filed a missing person's report for person A, person B who knows that person A is lost will log into the system and find out that his report has already been filed. Person B will therefore not file a missing report for the same person. This saves redundancy, and improves knowledge sharing.

Naturally, a citizen will not be able to edit information about an already filed report. This is the work of police officers and administrators of the system who can access the database. If a citizen were allowed to edit a report that has already been filed, the police officers will not respond to the appropriate report. For instance, if someone filed a burglary report that occurred in Washington DC, and another person edited the location to Westminster, there will be pandemonium, with crime units banging at doors of innocent, peace-loving, law-abiding citizens, barraging them with a fusillade of questions.

## **4) To enable officials (administrators) to add police officers to the system.**

Only administrators are allowed to add police officers to the system. A new police officer is assigned an ID and a password. The police officer is not expected to share this password with anyone since he will use it to login into the system.

The first few administrators are added by the developers so that the system has a starting point. If there were no first administrators, then there would be no police officers, and citizens filing criminal reports will spend precious time barking up the wrong tree. The first administrators can then proceed to add new administrators to the system who will add new police officers. The administrators have full access to the system and the database and therefore can do pretty much anything.

### *i. Addition of a new police officer*

The administrator will add a police officer by collecting his (the officer's) details. These details include the officer's name, age, date of birth, gender and ID number. The officer is then assigned a password and an ID to use while logging into the system.

ii. *Deleting an officer*

It is a simple matter to delete an officer. When an officer is deleted, for example he has chosen to hang up his boots, or has died, all his activity is deleted from the system. This approach is not as good as it might sound since the officer disappears with his data. In future, deleting an officer will simply prevent an officer from logging in, and his activity in the system will be retained.

**5) To enable police officers to view and respond to the filed criminal reports.**

There would be no need for the police system if officers were not able to view and respond to the reports that the citizens file. The citizens might as well take the law into their own hands, and dish out justice the way they see fit. A police officer, therefore, will log into the system and view the reports that have not been responded to and take appropriate action. If the officer marks the report as responded to, the citizens will be able to see this and rest assured that something is being done about the matter.

**6) To enable police officers to view feedback from the citizens.**

Police officers will have the ability to view the feedback given to them by the citizens, whether positive or negative. The developers of the system will have access to this feedback and therefore will know whether to improve the system or leave well enough alone.

**7) To enable police officers to add criminals to the database, or delete existing ones, or modify the data and search for a particular criminal.**

Police officers will be endowed with the power to add criminals to the database. Data such as the ID of the criminal, his first and last name, age, among others are saved to the database. An officer can search for a criminal using the data collected. Also, officers will be able to delete criminals from the database, for example if they mend their ways, or if they die from gunshot wounds.

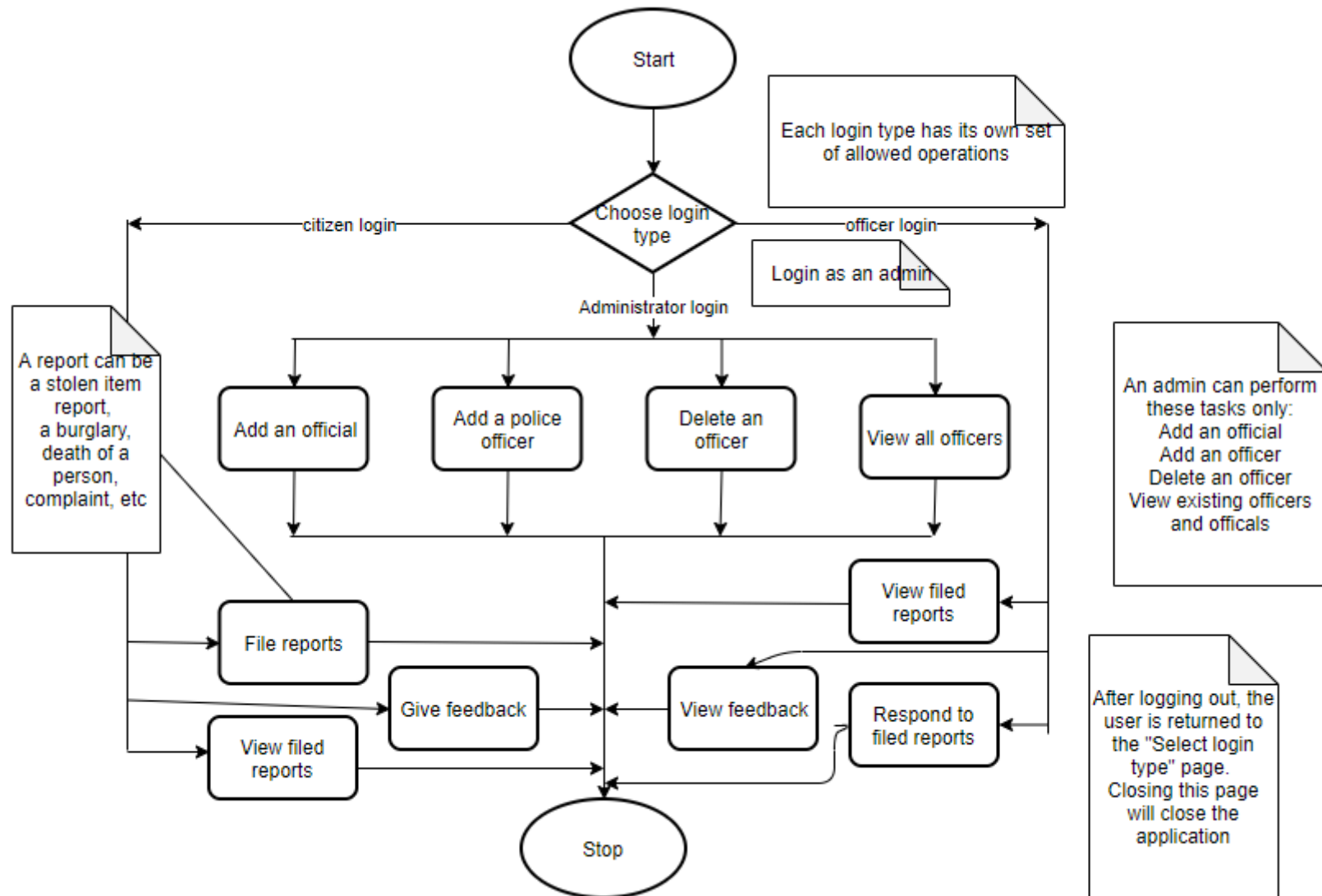
i. *Adding a criminal*

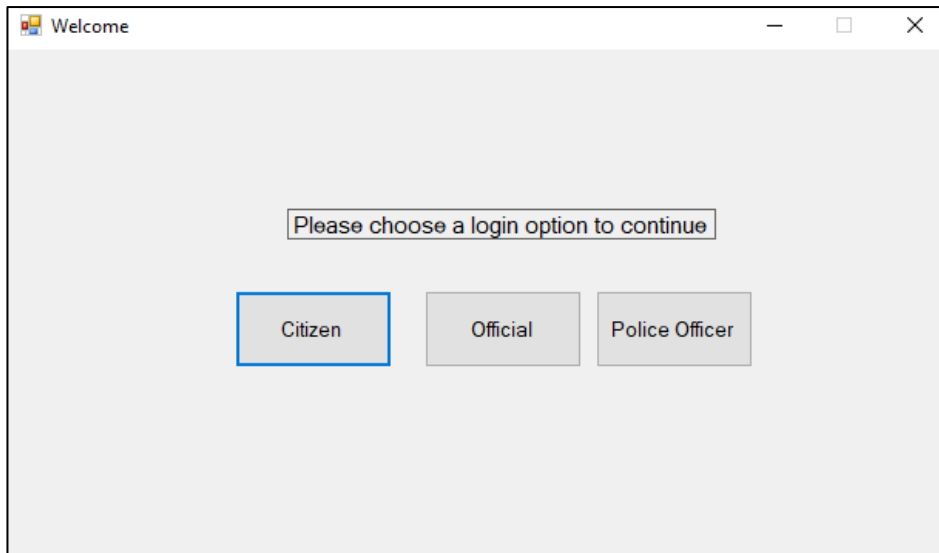
This is pretty easy. The criminal's information is collected (assuming the criminal is not a John Doe) and stored in the database.

ii. *Searching for a criminal*

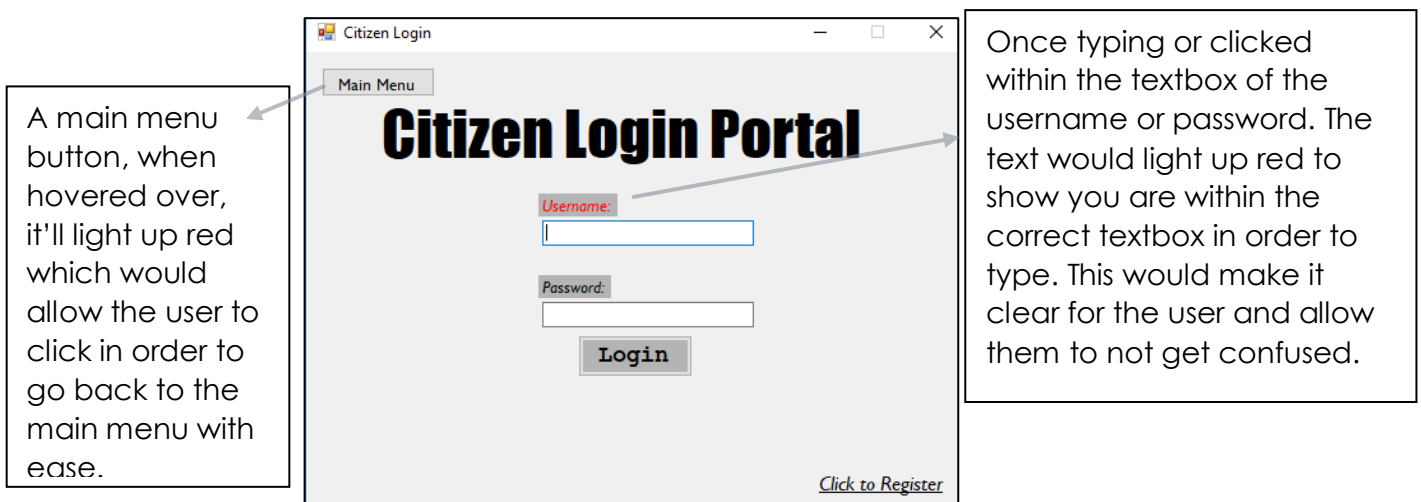
The search process is quite efficient, if I may say so myself. A criminal can be searched for by ID or by first name or by last name. Also, typing a character, for example in the first name field, will bring all the criminals whose first names start with the character, or contain a substring with the typed character.

# Overall System Design





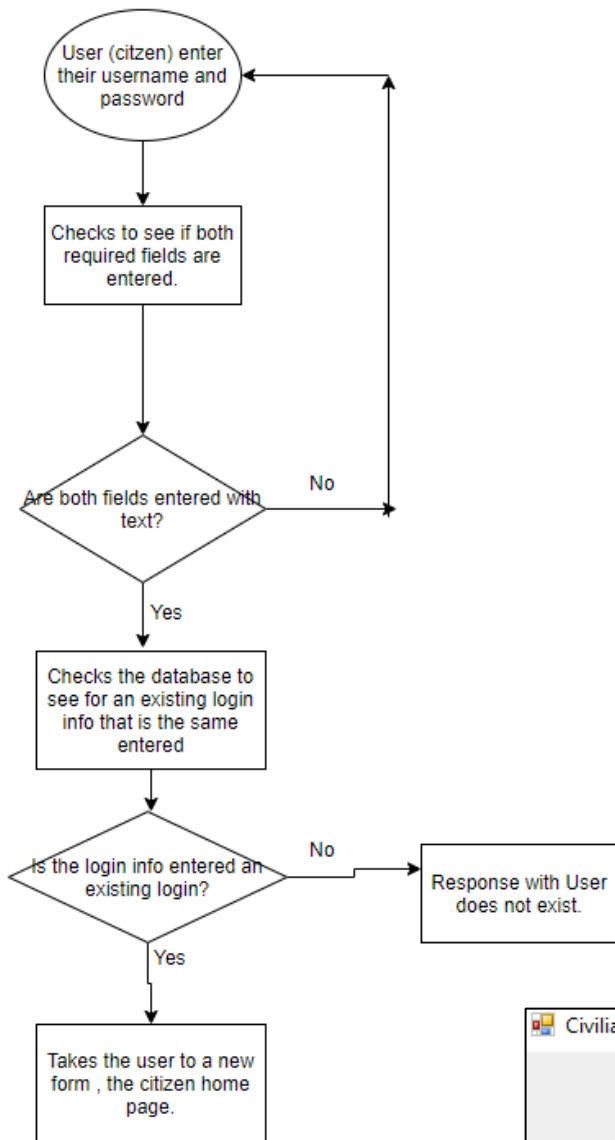
Prototype number 1, this is the main form that will first be displayed once first starting the program. As shown, it will have three login options for the user to direct themselves to whichever login portal they click on. This is beneficial to have as the first main screen because it will give the users the ability to choose which part they are at and can go from there to access their desired needs.



A main menu button, when hovered over, it'll light up red which would allow the user to click in order to go back to the main menu with ease.

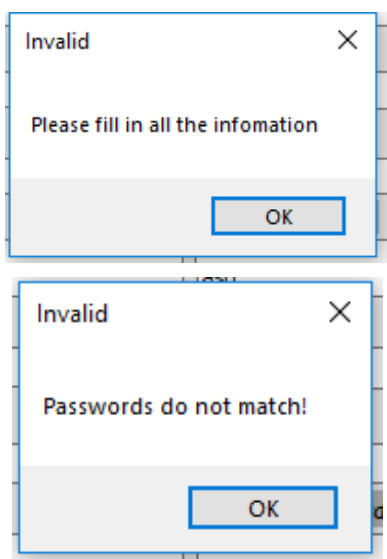
Once typing or clicked within the textbox of the username or password. The text would light up red to show you are within the correct textbox in order to type. This would make it clear for the user and allow them to not get confused.

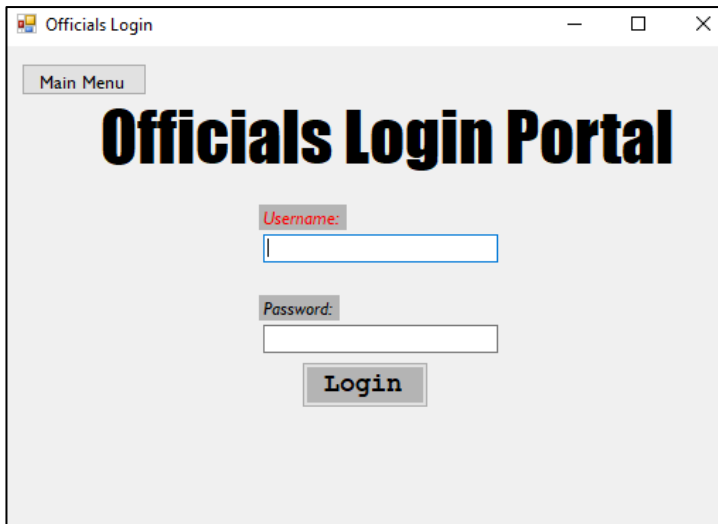
This is the login portal for citizens once clicked the button at the home screen. It will allow users (citizens) who have previously registered to login if they have an account which would access my database where all the login users are stored. However, if the user has not registered, there is a button to click to register which would redirect them to a different form where they would complete a registration in order to login to the citizen's home page.



User not registered. They have the option to Click to Register button on the bottom right. They would have many options required in order to fill out the registration form in order for access to the citizen's homepage. In order to create an account, it will ask for their first name, surname, age, gender, mobile no, postcode and password. This is in order for the officials to have contact information, to know where and how to contact users if needed. Once all the information is filled in and user has clicked the register button it will create an additional row in the database that will save the users login.

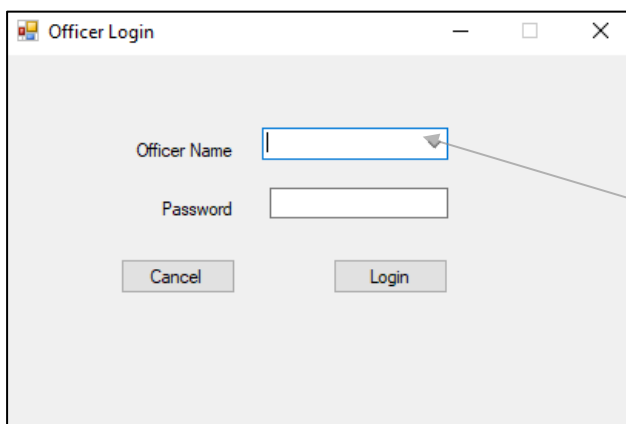
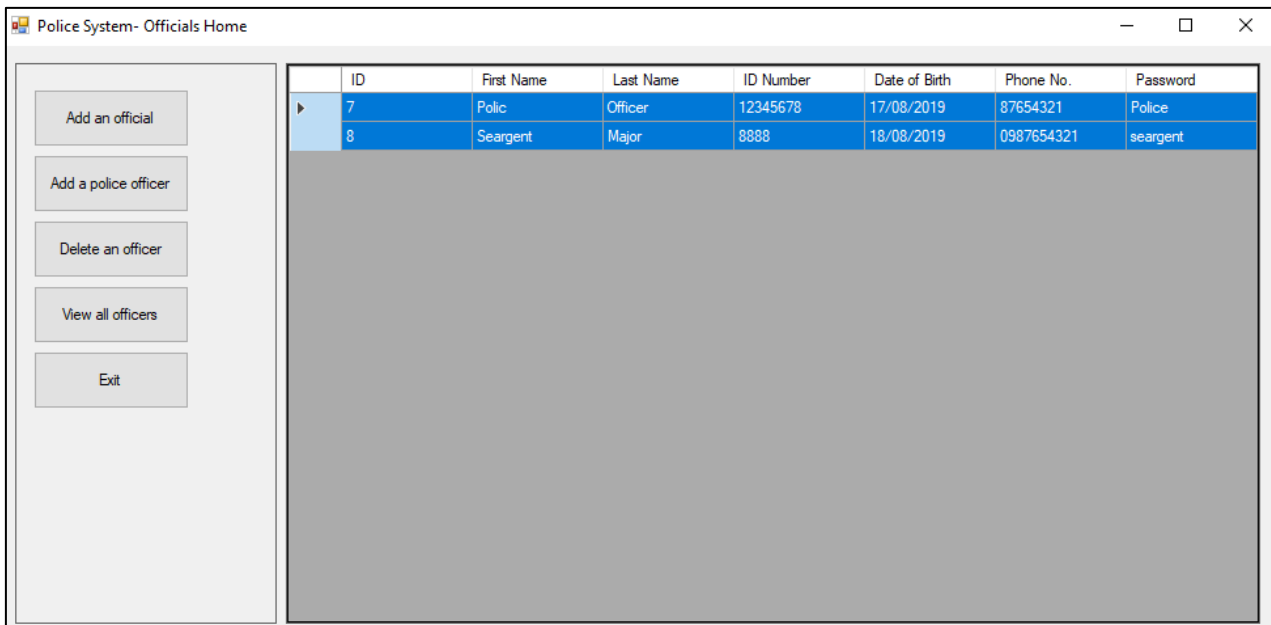
Moreover, I have also added an error checking properties within the application. As shown by the images if all of the required info are not entered it or the passwords do not match, it will pop up with a message box stating as shown.





Prototype, this is the official's login portal, it has same the design as citizen's login portal however, there is no option to register as this would most likely me one or two people, from the main office. A head official from the specific police station.

Once logged it, the official can add more police officers or even officials. And also view all the officers there currently are by using a data grid view.



This is the Police Officers portal, as this prototype shows, it is significantly different from the two login portals earlier. This one instead of an enter username. There is a drop down menu which links to the officer login database and displays all the officers, which the officer would select their name and enter their password in order to log themselves in.

## Database Design for Civilians Login Data

ID	FirstName	Surname	Age	Gender	MobileNum	Postcode	UPassword	Click to Add
10	Civilian	Civilian	20	Male	987654321	NY 123	civilian	
11	Ah	wag1	12	Male	07983875083	SW19 2LE	test123	
(New)			0					

This is the database design for the civilian's login data, where it is stored. The primary key is the ID and this would also later connect to other tables that holds reports from each user. The First Name is the username that the civilian will use in order to login as well as their password. There will be error checking to make sure both the information entered is correct and there is an existing user by checking this database, otherwise will state no user found, and would ask them to register.

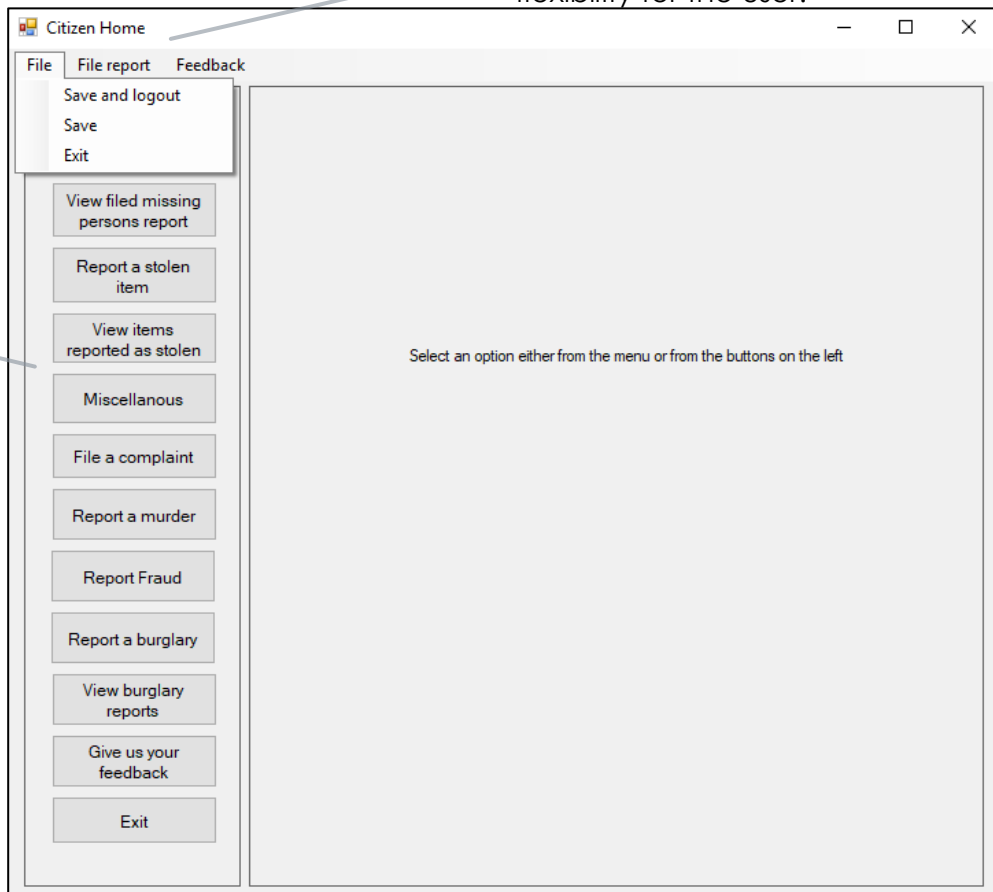
Field Name	Data Type
ID	AutoNumber
FirstName	Long Text
Surname	Long Text
Age	Number
Gender	Long Text
MobileNumber	Long Text
Postcode	Long Text
UPassword	Long Text

This shows the Fieldname as well as the Data Type for each.

## Civilian's Home Design

Menu bar, allowing the choice between and more flexibility for the user.

Different categories of crimes which the user can pick to report and also view each one they make.

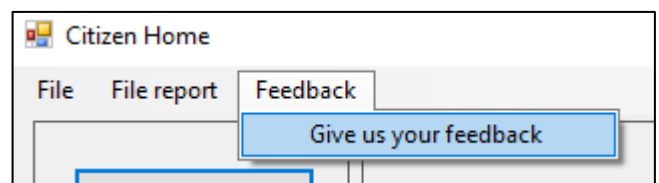
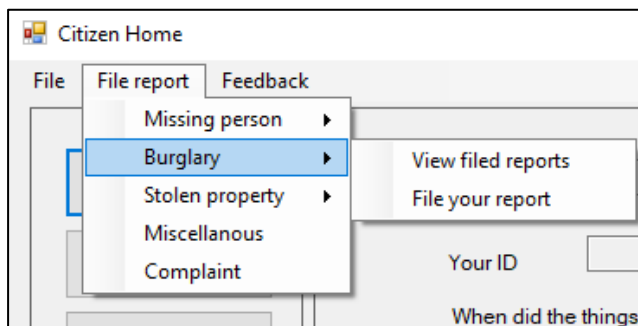


Once logged into the civilian's home page, it has a menu strip which consists of File, File report and Feedback as shown. Within the File drop down option it'll enable users to save and exit, after they have filed reports etc.

There is also a File report menu option, which lists all the crimes that the user can report. And also a miscellaneous category if there isn't a crime listed that they wish to report.

Having a menu strip just allows more ease of access and flexibility to either choose the button on the home page or within the menu bar.

Last of all there is a feedback on the menu bar which allows users to give the officers feedback and also suggestions for improvements or just general feedback regarding the department.



Mandatory required fields need to be all entered in order for the user to submit their report.

## Citizen Reports

Has a date time pick which allows the user to easily pick where the person was last seen.

The ability for the user to clear everything if needed to start over. Convenient for them, instead of having to remove everything.

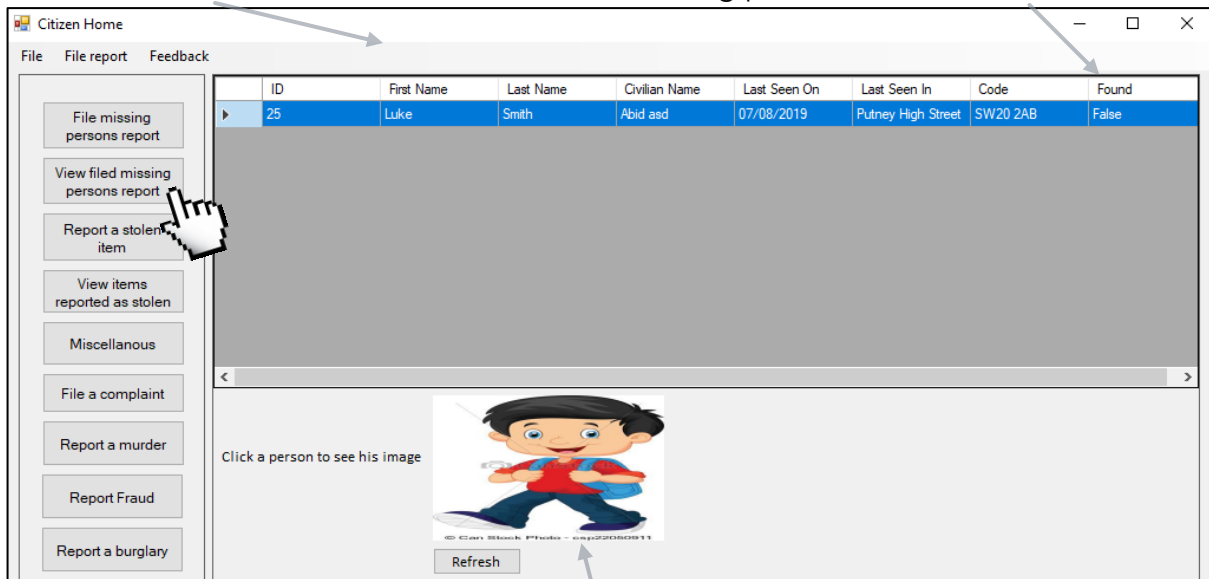
Has an option for the citizen to browse an image on their device of a photo of the missing person. Allows the police to identify.



The user citizen will have many choices to pick which crimes they'd like to report. As shown in the example for the File missing person's report. It allows to list of the key details required such as the first name last name and last seen as well as any phone number they may have had and their address. After all the necessary fields have been entered they can also attach an image by click browse which would allow the police to clearly know what the person looks like, if it is a friend or relative.

Thereafter, once the user clicks 'Submit report', it will check whether the inputted data is provided and is valid. It will display a message if the data is not valid. Once checked and is the data provided is valid, it will store all the information into a database with a unique ID which the police officers can access in their login homepage. As for the image, it would create a byte array to store the selected image which the officers can view.

Datagridview showing all the information the citizen has provided in the report.  
As well as the found column to see if the missing person has been found.



Allows the user to view the image they have provided in the report.

As well as the user's ability to file reports, they will also be able to view filed reports they have made as well. As you can see here, I have used a data grid view layout which lets the users the option to view the missing person, in this case, and see all the info. By clicking on a selected missing person, they would be able to view the image of the person as shown.

Another feature that is implemented into this, is the 'Found' column, this allows the user to see if the missing person report they have made, have been found by the police, etc. This can be edited in the police officer's homepage which will be shown later on.

This is another report option, which is slightly different from the file missing people's option. This is a different layout for reporting a stole item. It shows the users ID as well as their name, which is greyed out meaning it is not editable. Just for view. Gives an option to select a date which the stolen possession was lost as well as any description and information that the citizen could provide to the officers which will allow them to track

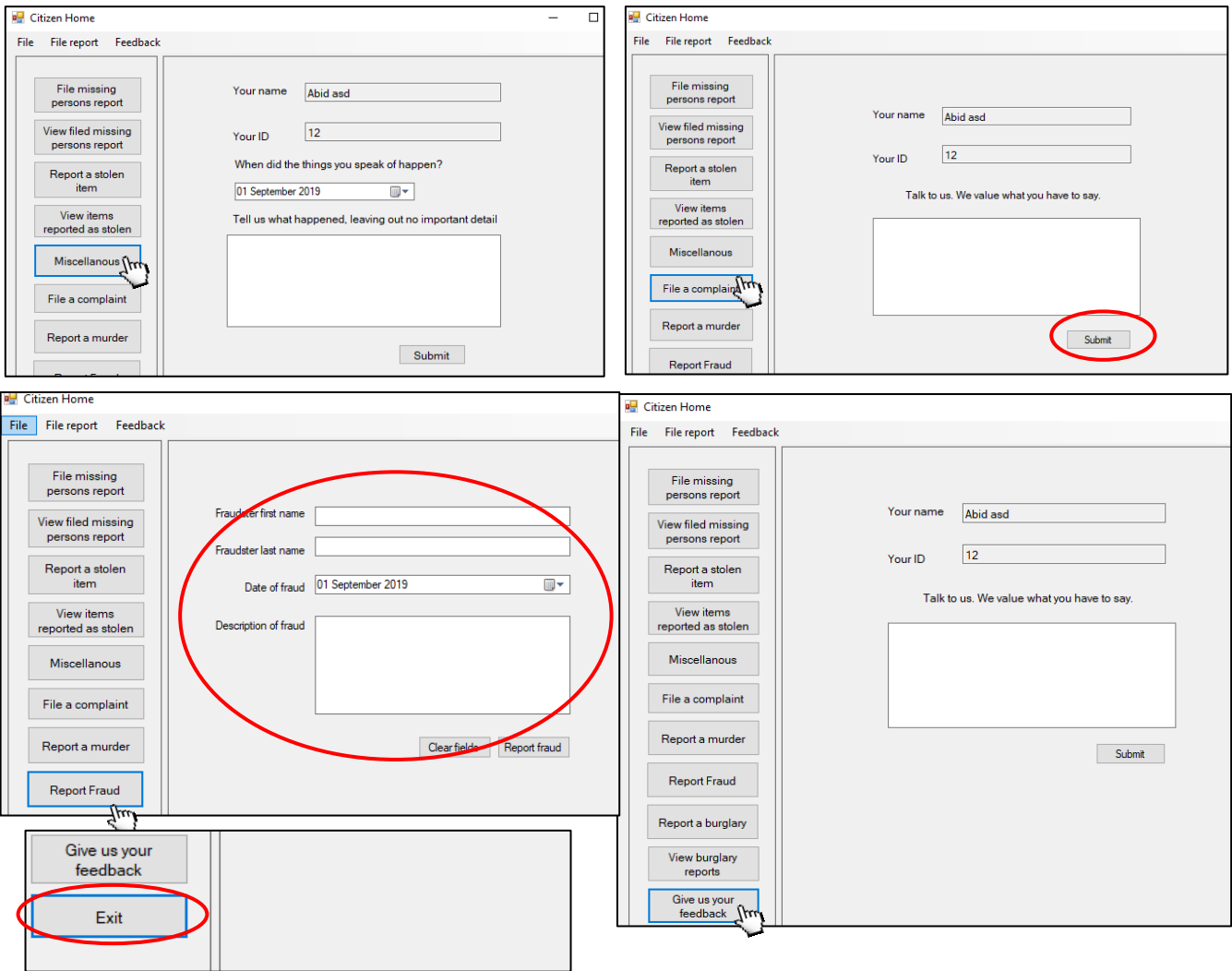
All stolen items:

	ID	Civilian Name	Date	Item found?
▶	7	Abid asd	21/07/2019	<input type="checkbox"/>

Click an item to see its description

My car has been stolen on 21st of July.  
It is a VW Golf.

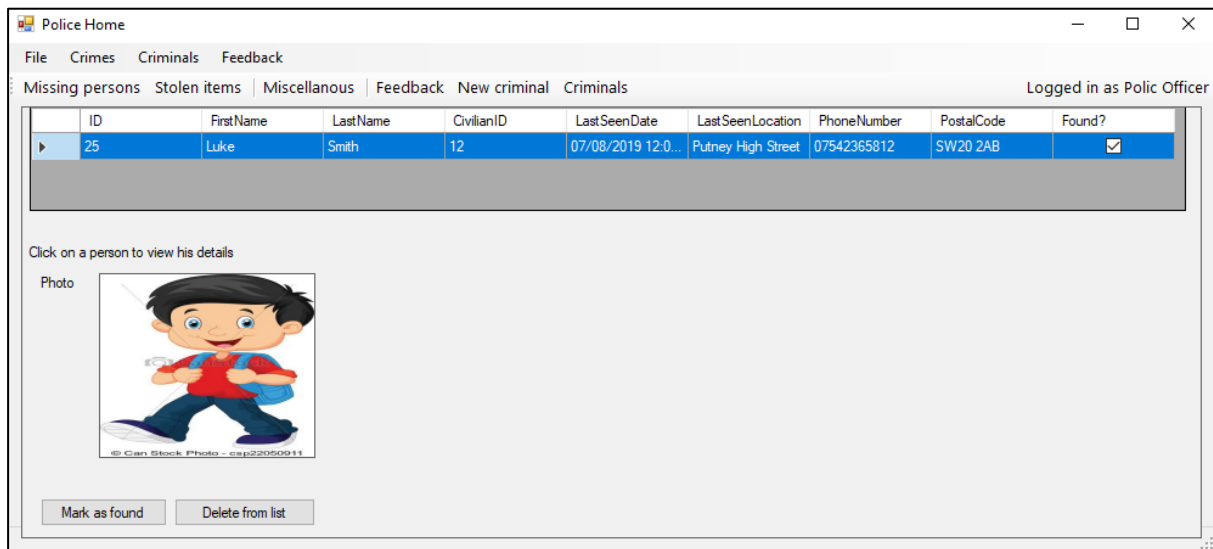
Simply layout as the missing persons reports. But the report for the stolen item once clicked, shows the description that was provided when reported. And also has the same feature if the item was found or not. As shown it is currently unchecked which means the item has not been found. If it is viewed as a check mark within the box, the item has been found and the citizen would be notified. Once again using the data grid view.



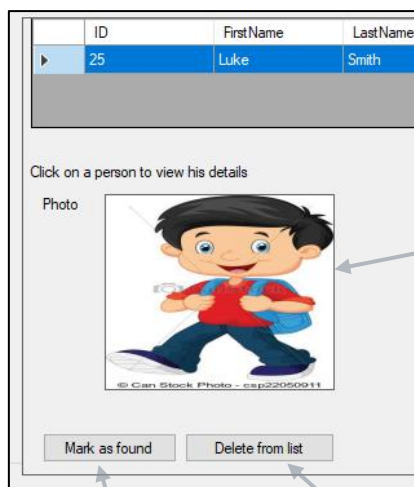
These are some other examples which shows how reports are laid out. As well as this, there is a feedback and complaint system which the users can submit which there after the officers can view if there are any issues etc. or any feedback that the civilian might want them to know or any suggestions.

Moreover, for convenience, there is also an exit button at the bottom which would allow the user to close their homepage, this allows the user to have the ease to choose between to options, the close button as well. However, in future designs as these are prototypes, I am looking to add a flat modern design look with the bar on top, which would be shown shortly.

## Officers Homepage



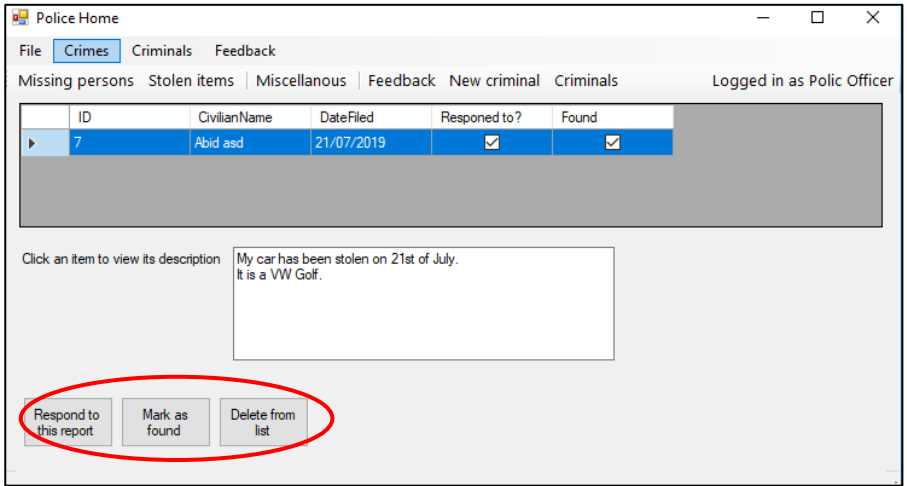
Once logged into the police officers home as shown from the login portal. This is where the officers would have access to view all reports submitted by the citizens as well as having the option to respond to them which would be explained.



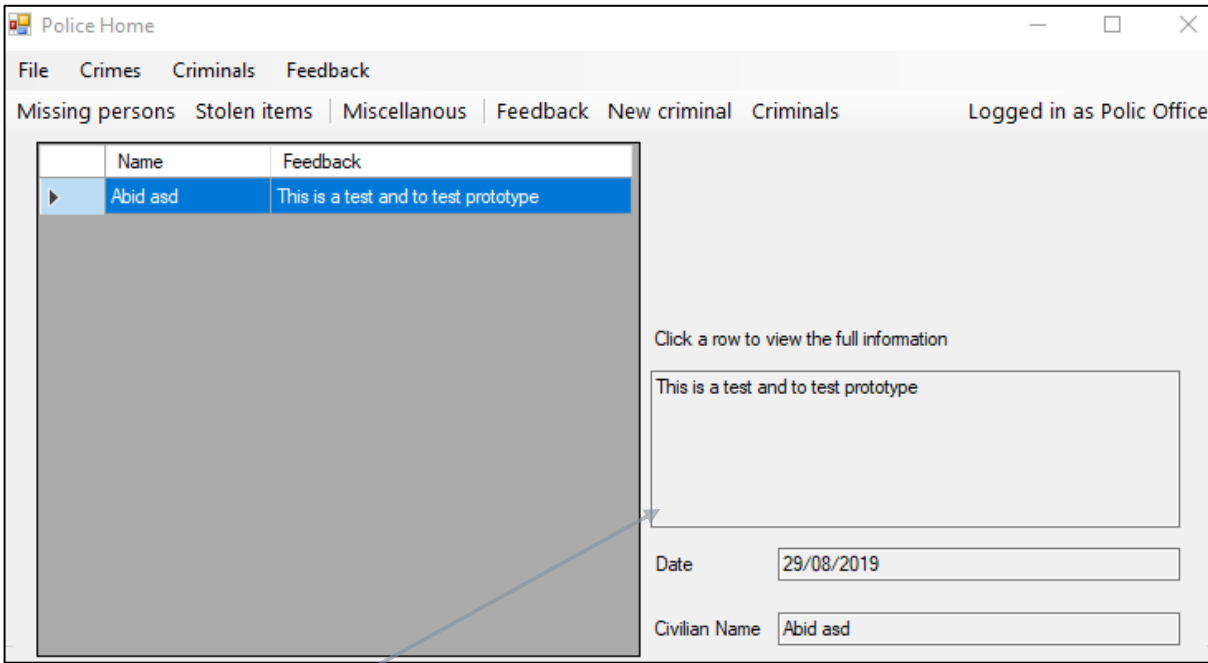
As shown in the above image, this is an example of a missing person which the user has reported.

Once clicked onto the selected mission person it will show the photo which would have also been submitted by the user as well.

The officers have a button which they can select to mark the missing person as found, this will be updated in the database and therefore citizens can view the missing person reported as being found. After a while, once all has been confirmed and additional paper work have been completed. The officers may click the delete from list button if they would like to remove the record from the database.

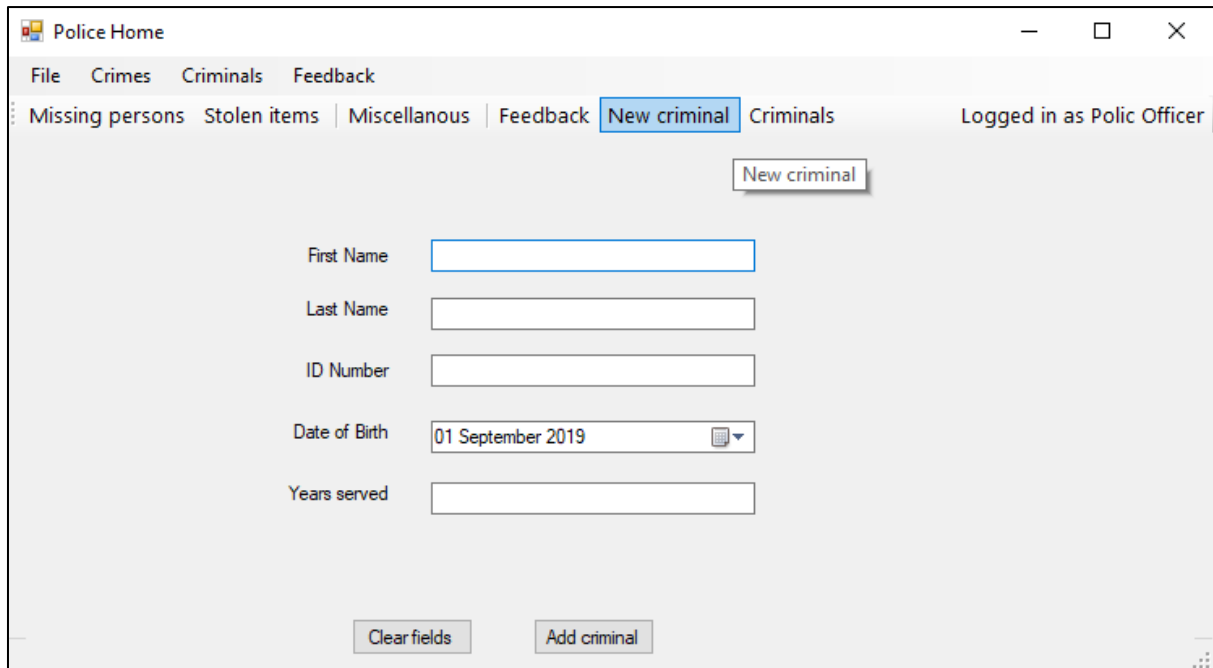


Another example of stolen items report, there is an additional respond to this report button which would allow the officers to keep track of if they have viewed and currently searching, this would allow them to be more organised as they would know they are investigating this specific report instead of marking as found. So two options, which allow the officers to keep in contact and then marking as found for the citizen.



This is the prototype design view of the Miscellaneous reports. As shown, the officers can see a data grid view of the feedback and the civilians name. However, if it contains larger information which is not shown. They will have the option to select the record, and it'll show the full information in the box provided, also includes the date the report was made in as well as the civilian's name that submitted it.

There are many more crimes and reports which the officers can view, but those are just some prototype examples of how the police can actually see and attend to, in order to find a solution and keep organised whilst being maintained.

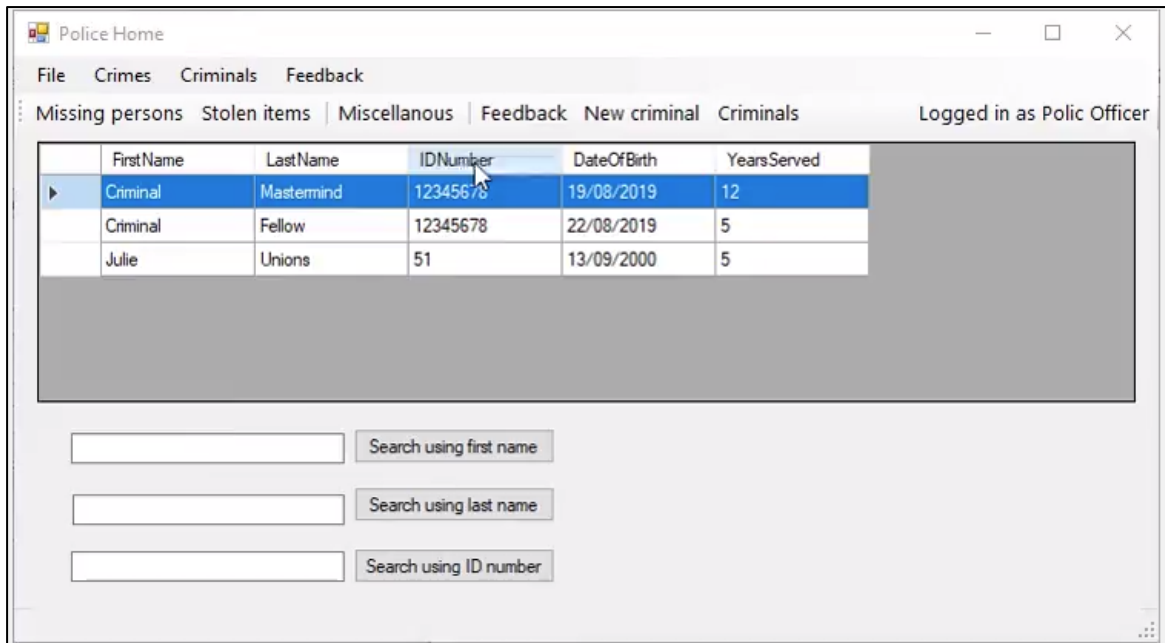


As well as the officers having the permission to view and respond to reports by the civilians.

They also have an option to add criminals themselves, this will benefit the department, for example if the criminal is on the watch list, they can add to keep track of all criminals as well as if they have arrested the criminal from a report that a citizen made they can add the criminal into the database here too.

As shown in the screenshot above, the option for the officers to add a new criminal to the database. By typing in their details. First name, last time, an ID number of choice they would like the criminal to be identified by, date of birth and how many years they have served so far.

As always, shown by other screenshots beforehand, there is a button which the officers can click in order to clear all fields and start from fresh instead of having to remove all the data in the fields. This will be much more convenient for them.



<https://streamable.com/og7qj>

This is the Criminals section design, where the officers can view all the criminals by their first and last name as well as their ID, date of birth and the years served. It is designed by a datagridview for the officers to view. In addition to this, there is a search function that the officers can search using first name, last name or by ID number.

This is favourable, as if there is many criminals listed, it will take a lot of time scrolling through all the criminals in the data grid view. Instead, by searching with one of the options it will be easier and much quicker to view if needed. Showing in the link above, is an example of how it works.

Through the integrated modules, AM Radiants software allows the police department to help and handle all the paper work online, giving scalable benefits to typical branches. Maintenance of record is also taken care of, making it manageable for the departments to work on a real-time basis.



## Designs prototype v2



These are some prototype version 2 of some designs which I have created in order for it to be more attractive.

However, by conducting some questionnaires, asking the public and some police stations. It was thought to have been too bright and too dynamic for professional and specialised use.

**Final V3 Designs coming soon..**

```

Public Class Form1
    Private Sub btnOfficial_Click(sender As Object, e As EventArgs)
        Dim official As New OfficialsLogin
        official.ShowDialog()
        ' Me.Visible = False
    End Sub

    Private Sub btnCivilian_Click(sender As Object, e As EventArgs)
        Dim civilian As New CiviliansLogin
        civilian.ShowDialog()
        ' Me.Visible = False
    End Sub

    Private Sub btnCivilian_Click_1(sender As Object, e As EventArgs) Handles
btnCivilian.Click
        Dim civilian As New CiviliansLogin
        Me.Visible = False
        civilian.ShowDialog()

    End Sub

    Private Sub btnOfficial_Click_1(sender As Object, e As EventArgs) Handles
btnOfficial.Click
        Dim official As New OfficialsLogin
        Me.Visible = False
        official.ShowDialog()
    End Sub

    Private Sub btnPoliceOfficer_Click(sender As Object, e As EventArgs) Handles
btnPoliceOfficer.Click
        Dim login As New OOfficerLogin
        login.ShowDialog()
    End Sub
End Class

```

```

Module Module1
    Public connectionString As String = "Provider=Microsoft.Jet.OLEDB.4.0; Data
Source=Databases.mdb" 'Connection string to the database
    Public backForm As Form ' will contain the home that we will go back to, if the
current form is closed
    Public civilianID As Integer = -1 'holds the id of the currently loggen in civilian
    Public officerID As Integer = -1 'holds the id of the currently logged in officer
End Module

Imports System.Runtime.InteropServices

Public Class FormMain
    Public Const WM_NCLBUTTONDOWN As Integer = &HA1
    Public Const HT_CAPTION As Integer = &H2

    <DllImport("user32.dll", SetLastError:=True, CharSet:=CharSet.Auto)>
    Private Shared Function SendMessage(ByVal hWnd As IntPtr,
        ByVal Msg As UInteger,
        ByVal wParam As IntPtr,
        ByVal lParam As IntPtr) As IntPtr
    End Function

    <DllImportAttribute("user32.dll")>
    Public Shared Function ReleaseCapture() As Boolean
    End Function

    Private Sub imgMin_Click(sender As Object, e As EventArgs) Handles imgMin.Click
        Me.WindowState = FormWindowState.Minimized
    End Sub

    Private Sub imgMax_Click(sender As Object, e As EventArgs) Handles imgMax.Click
        Me.WindowState = FormWindowState.Maximized
        Me.imgMax.Visible = False
        Me.imgRes.Visible = True
    End Sub

    Private Sub imgRes_Click(sender As Object, e As EventArgs) Handles imgRes.Click
        Me.WindowState = FormWindowState.Normal
        Me.imgRes.Visible = False
        Me.imgMax.Visible = True
    End Sub

    Private Sub imgClo_Click(sender As Object, e As EventArgs) Handles imgClo.Click
        Application.Exit()
    End Sub

    Private Sub btnCivilian_MouseMove(sender As Object, e As MouseEventArgs) Handles
btnCivilian.MouseMove
        btnCivilian.Image = ProjectV1.My.Resources.imgCitizen2
    End Sub

    Private Sub btnCivilian_MouseLeave(sender As Object, e As EventArgs) Handles
btnCivilian.MouseLeave
        btnCivilian.Image = ProjectV1.My.Resources.imgCitizen1
    End Sub

    Private Sub btnPoliceOfficer_MouseMove(sender As Object, e As MouseEventArgs)
Handles btnPoliceOfficer.MouseMove
        btnPoliceOfficer.Image = ProjectV1.My.Resources.imgPoliceOfficer2
    End Sub

```

```

Private Sub btnPoliceOfficer_MouseLeave(sender As Object, e As EventArgs) Handles
btnPoliceOfficer.MouseLeave
    btnPoliceOfficer.Image = ProjectV1.My.Resources.imgPoliceOfficer1
End Sub

Private Sub btnOfficial_MouseMove(sender As Object, e As MouseEventArgs) Handles
btnOfficial.MouseMove
    btnOfficial.Image = ProjectV1.My.Resources.imgOfficial2
End Sub

Private Sub btnOfficial_MouseLeave(sender As Object, e As EventArgs) Handles
btnOfficial.MouseLeave
    btnOfficial.Image = ProjectV1.My.Resources.imgOfficial1
End Sub

Private Sub pnlTitle_MouseMove(sender As Object, e As MouseEventArgs) Handles
pnlTitle.MouseMove
    Try
        If e.Button = MouseButton.Left Then
            ReleaseCapture()
            SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
        End If
    Catch ex As Exception

    End Try

End Sub

Private Sub btnCivilian_Click(sender As Object, e As EventArgs) Handles
btnCivilian.Click
    Dim civilian As New CiviliansLogin
    Me.Visible = False
    civilian.ShowDialog()
End Sub

Private Sub btnPoliceOfficer_Click(sender As Object, e As EventArgs) Handles
btnPoliceOfficer.Click
    Dim login As New OOfficerLogin
    Me.Visible = False
    login.ShowDialog()
End Sub

Private Sub btnOfficial_Click(sender As Object, e As EventArgs) Handles
btnOfficial.Click
    Dim official As New OfficialsLogin
    Me.Visible = False
    official.ShowDialog()
End Sub

Private Sub Label15_MouseMove(sender As Object, e As MouseEventArgs) Handles
Label15.MouseMove
    Try
        If e.Button = MouseButton.Left Then
            ReleaseCapture()
            SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
        End If
    Catch ex As Exception

    End Try
End Sub
End Class

```

```

Imports System.Data.OleDb
Imports System.Data
Imports System.Runtime.InteropServices

Public Class CiviliansLogin
    Public Const WM_NCLBUTTONDOWN As Integer = &HA1
    Public Const HT_CAPTION As Integer = &H2

    <DllImport("user32.dll", SetLastError:=True, CharSet:=CharSet.Auto)>
    Private Shared Function SendMessage(ByVal hWnd As IntPtr,
        ByVal Msg As UInteger,
        ByVal wParam As IntPtr,
        ByVal lParam As IntPtr) As IntPtr
    End Function

    <DllImportAttribute("user32.dll")>
    Public Shared Function ReleaseCapture() As Boolean
    End Function

    Private Sub txtCiviliansUsername_enter(sender As Object, e As EventArgs) Handles
txtCiviliansUsername.Enter
        labelCiviliansUsername.ForeColor = Color.Red
    End Sub
    Private Sub txtCiviliansUsername_Leave(sender As Object, e As EventArgs) Handles
txtCiviliansUsername.Leave
        labelCiviliansUsername.ForeColor = Color.Black
    End Sub

    Private Sub txtCiviliansPassword_enter(sender As Object, e As EventArgs) Handles
txtCiviliansPassword.Enter
        LabelCiviliansPassword.ForeColor = Color.Red
    End Sub
    Private Sub txtCiviliansPassword_Leave(sender As Object, e As EventArgs) Handles
txtCiviliansPassword.Leave
        LabelCiviliansPassword.ForeColor = Color.Black
    End Sub

    Private Sub lblCiviliansRegister_MouseEnter(sender As Object, e As EventArgs)
Handles lblCiviliansRegister.MouseEnter
        lblCiviliansRegister.ForeColor = Color.White
    End Sub
    Private Sub lblCiviliansRegister_MouseLeave(sender As Object, e As EventArgs)
Handles lblCiviliansRegister.MouseLeave
        lblCiviliansRegister.ForeColor = Color.Black
    End Sub

    Private Sub lblCiviliansRegister_MouseClick(sender As Object, e As MouseEventArgs)
Handles lblCiviliansRegister.MouseClick
        frmCiviliansRegister.Show()
        Me.Visible = False
        'Me.Hide()
    End Sub

    Private Sub btnCiviliansMainMenu_Click(sender As Object, e As EventArgs)
        Me.Visible = False 'Hide this form
        FormMain.Show() 'Show the first form of the application. (The one that enables
choosing of the login option: whether
        'police officer, administrator, or citizen)
    End Sub
End Class

```

```

End Sub

'Private Sub btnCiviliansMainMenu_MouseEnter(sender As Object, e As EventArgs)
'    btnCiviliansMainMenu.ForeColor = Color.Red
'End Sub

'Private Sub btnCiviliansMainMenu_MouseLeave(sender As Object, e As EventArgs)
'    btnCiviliansMainMenu.ForeColor = Color.Black
'End Sub

Private Sub btnCiviliansLogin_Click(sender As Object, e As EventArgs) Handles
btnCiviliansLogin.Click
    If txtCiviliansPassword.Text = "" Or txtCiviliansUsername.Text = "" Then
        MessageBox.Show("Username And Password", "Please Enter") ',
MsgBoxStyle.YesNo)=MsgBoxResult.Yes)
        Return
    End If

    Dim uname As String = ""
    Dim pword As String
    Dim username As String = ""
    Dim pass As String = ""

    uname = txtCiviliansUsername.Text
    pword = txtCiviliansPassword.Text

    Dim query As String = "SELECT CiviliansLoginData.ID, " &
        "CiviliansLoginData.FirstName, CiviliansLoginData.UPassword FROM
CiviliansLoginData" &
        " where CiviliansLoginData.FirstName = '" & uname & "' and
CiviliansLoginData.UPassword " &
        " = '" & pword & "';"

    Try
        Dim connection = New OleDbConnection(Module1.connectionString) 'Create a
connection to the database
        connection.Open() 'Open the connection
        Dim cmd As New OleDbCommand(query, connection) 'Command to execute against
the connection
        Dim reader As OleDbDataReader = cmd.ExecuteReader() 'Read the data in the
database
        'Only the roww that matche the user name and password that the user has
provided are selected
        If reader.HasRows Then
            reader.Read()
            Integer.TryParse(reader.GetValue(0), Module1.civilianID)
            Me.Visible = False
            txtCiviliansUsername.Clear()
            txtCiviliansPassword.Clear()
            Dim frm As New FormCivilianHome
            Module1.backForm = Me
            connection.Close() 'Close the connection to free up resources
            Me.Close()
            frm.ShowDialog()
        Else
            'No data matches the one provided
            MessageBox.Show("Login does not exist!", "Login Failed")
        End If
    Catch ex As Exception
        MessageBox.Show("An error ocured: " & vbCrLf &

```

```

        ex.Message & vbCrLf & ex.StackTrace &
        vbCrLf & query, "Login Failed")
    End Try
End Sub

Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
    Me.Visible = False 'Hide this form
    FormMain.Show() 'Show the first form of the application. (The one that enables
choosing of the login option: whether
'police officer, administrator, or citizen)
End Sub

Private Sub pnlTitle_MouseMove(sender As Object, e As MouseEventArgs) Handles
pnlTitle.MouseMove
    Try
        If e.Button = MouseButton.Left Then
            ReleaseCapture()
            SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
        End If
    Catch ex As Exception

    End Try
End Sub

Private Sub Label1_MouseMove(sender As Object, e As MouseEventArgs) Handles
Label1.MouseMove
    Try
        If e.Button = MouseButton.Left Then
            ReleaseCapture()
            SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
        End If
    Catch ex As Exception

    End Try
End Sub
End Class

```

```

Imports System.Data.OleDb
Imports System.Data

```

```

Public Class OfficialsLogin

```

```

    Private Sub txtCiviliansUsername_enter(sender As Object, e As EventArgs) Handles
txtOfficialsUsername.Enter
        labelOfficialsUsername.ForeColor = Color.Red
    End Sub
    Private Sub txtCiviliansUsername_Leave(sender As Object, e As EventArgs) Handles
txtOfficialsUsername.Leave
        labelOfficialsUsername.ForeColor = Color.Black
    End Sub

    Private Sub txtOfficialsPassword_enter(sender As Object, e As EventArgs) Handles
txtOfficialsPassword.Enter
        LabelOfficialsPassword.ForeColor = Color.Red
    End Sub
    Private Sub txtCiviliansPassword_Leave(sender As Object, e As EventArgs) Handles
txtOfficialsPassword.Leave
        LabelOfficialsPassword.ForeColor = Color.Black

```



```

End Sub

Private Sub btnOfficialsLogin_Click(sender As Object, e As EventArgs) Handles
btnOfficialsLogin.Click
    'If some input has been provided, return and display an error message
    If txtOfficialsPassword.Text = "" Or txtOfficialsUsername.Text = "" Then
        info!")
        MessageBox.Show("Please provide your user name and password", "Missing
        Return
    End If

    Dim uname As String = ""
    Dim pword As String
    Dim username As String = ""
    Dim pass As String = ""
    If txtOfficialsUsername.Text = "" Or txtOfficialsPassword.Text = "" Then
        'MessageBox.Show("Username And Password", "Please Enter")
    Else
        'Check if the provided user name and password are already in the database.
        'If not, display an error message and return. Otherwise, the login is
        valid, therefore proceed as normal
        uname = txtOfficialsUsername.Text
        pword = txtOfficialsPassword.Text
        Dim query As String = "Select password From OfficialsLoginData where
name= '" & uname & "';"
        Dim dbsource As String = Module1.connectionString
        Dim connection = New OleDbConnection(dbsource)
        Dim cmd As New OleDbCommand(query, connection)
        connection.Open()
        Try
            pass = cmd.ExecuteScalar().ToString
        Catch ex As Exception
            MessageBox.Show("Login does not exist!", "Login Failed")
        End Try
        If (pword = pass) Then
            txtOfficialsPassword.Clear()
            txtOfficialsUsername.Clear()
            Dim frm As New OfficialHome
            frm.ShowDialog()
            Me.Visible = False
        Else
            MessageBox.Show("Incorrect user name or password!", "Login Failed")
            txtOfficialsUsername.Clear()
            txtOfficialsPassword.Clear()
        End If
    End If
End Sub

Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
    Me.Visible = False
    FormMain.Show()
End Sub
End Class

```

```
Imports System.Data.OleDb
```

```
Public Class OOfficerLogin
```

```
Private Sub OOfficerLogin_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
'Retrieve the first names of the already existing officers
```

```
Dim query As String = "Select FirstName from Officer"
```

```
Try
```

```
Dim conn As New OleDbConnection(Module1.connectionString)
```

```
Dim reader As OleDbDataReader
```

```
Dim cmd As New OleDbCommand(query, conn)
```

```
conn.Open()
```

```
reader = cmd.ExecuteReader()
```

```
If reader.HasRows Then
```

```
    cmbName.Items.Clear()
```

```
    While reader.Read()
```

```
        cmbName.Items.Add(reader.GetString(0))
```

```
    End While
```

```
End If
```

```
Catch ex As Exception
```

```
    MessageBox.Show("An error occured: " & vbCrLf & ex.Message,  
                    "Error!", MessageBoxButtons.OK, MessageBoxIcon.Error)
```

```
End Try
```

```
End Sub
```

```
Private Sub bt_Click(sender As Object, e As EventArgs)
```

```
Me.Close()
```

```
FormMain.Show()
```

```
End Sub
```

```
Private Sub btnLogin_Click(sender As Object, e As EventArgs) Handles btnLogin.Click
```

```
'Validate login. If login exists, login as normal. Otherwise, display an error message
```

```
Dim password = txtPassword.Text
```

```
Dim query = "Select OfficerPassword, ID from Officer"
```

```
Try
```

```
Dim conn As New OleDbConnection(Module1.connectionString)
```

```
Dim cmd As New OleDbCommand(query, conn)
```

```
Dim reader As OleDbDataReader
```

```
conn.Open()
```

```
reader = cmd.ExecuteReader()
```

```
Dim valid As Boolean = False
```

```
Dim id As Integer = -1
```

```
If reader.HasRows Then
```

```
    While reader.Read()
```

```
        Dim pass = reader.GetString(0)
```

```
        If pass = password Then
```

```
            valid = True
```

```
            id = CInt(reader.GetValue(1))
```

```
            Exit While
```

```
        End If
```

```
    End While
```

```
End If
```

```
If valid Then
```

```
    Module1.officerID = id
```

```
Me.Hide()
```

```
Dim home As New FormPoliceHome
```

```
home.ShowDialog()
```

```
Else
```

```

        MessageBox.Show("The password you provided is not valid! Please try
again",
        "Error!", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End If
Catch ex As Exception
    MessageBox.Show("An error occured: " & vbCrLf & ex.Message,
        "Error!", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
End Sub

Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
    Me.Close() 'Hide this form
    FormMain.Show() 'Show the first form of the application. (The one that enables
choosing of the login option: whether
'police officer, administrator, or citizen)
End Sub
End Class

Imports System.Runtime.InteropServices

Public Class FormCivilianHome
    Public Const WM_NCLBUTTONDOWN As Integer = &HA1
    Public Const HT_CAPTION As Integer = &H2

    <DllImport("user32.dll", SetLastError:=True, CharSet:=CharSet.Auto)>
    Private Shared Function SendMessage(ByVal hwnd As IntPtr,
        ByVal msg As UInteger,
        ByVal wParam As IntPtr,
        ByVal lParam As IntPtr) As IntPtr
    End Function

    <DllImportAttribute("user32.dll")>
    Public Shared Function ReleaseCapture() As Boolean
    End Function

    Private Sub CivilianHome_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles Me.FormClosed
        Module1.backForm.Show() 'Show the form that was opened before this one
    End Sub

    Private Sub CivilianHome_Closed(sender As Object, e As EventArgs) Handles
Me.Closed

    End Sub

    Private Sub mnuFileReportMissingPerson_Click(sender As Object, e As EventArgs)

    End Sub

    ''' <summary>
    ''' Attempts to open a form (a usercontrol) inside the main panel
    ''' </summary>
    ''' <param name="form">The form to be opened inside the main center panel</param>
    Private Sub openFormInPanel(ByRef form As UserControl)
        pnlMain.Controls.Clear() 'Remove any control that might already be on the main
panel
        form.Dock = DockStyle.Fill 'Make the user control fill the main panel
        pnlMain.Controls.Add(form) 'Add the user control to the main panel
    End Sub

```

```

Private Sub mnuFileReportMissingPersonFileReport_Click(sender As Object, e As
EventArgs) Handles mnuFileReportMissingPersonFileReport.Click
    Dim cntrl As New FormFileMissingPerson
    openFormInPanel(cntrl)
End Sub

Private Sub mnuFileReportMissingPersonViewReports_Click(sender As Object, e As
EventArgs) Handles mnuFileReportMissingPersonViewReports.Click
    Dim cntrl As New FormViewMissingPersons
    openFormInPanel(cntrl)
End Sub

Private Sub mnuFileReportBurglaryFileReport_Click(sender As Object, e As
EventArgs) Handles mnuFileReportBurglaryFileReport.Click
    Dim cntrl As New frmFileBurglaryReport
    openFormInPanel(cntrl)
End Sub

Private Sub mnuFileReportBurglaryViewReports_Click(sender As Object, e As
EventArgs) Handles mnuFileReportBurglaryViewReports.Click
    Dim cntrl As New frmViewFiledBurglaryReports
    openFormInPanel(cntrl)
End Sub

Private Sub FileYourReportToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles FileYourReportToolStripMenuItem.Click
    Dim cntrl As New frmFileStolenItemReport
    openFormInPanel(cntrl)
End Sub

Private Sub btnFileMissingPersonsReport_Click(sender As Object, e As EventArgs)
Handles btnFileMissingPersonsReport.Click
    openFormInPanel(New FormFileMissingPerson)
End Sub

Private Sub btnViewFiledMissingPersonsReports_Click(sender As Object, e As
EventArgs) Handles btnViewFiledMissingPersonsReports.Click
    openFormInPanel(New FormViewMissingPersons)
End Sub

Private Sub btnReportStolenItem_Click(sender As Object, e As EventArgs) Handles
btnReportStolenItem.Click
    openFormInPanel(New frmFileStolenItemReport)
End Sub

Private Sub btnReportDeath_Click(sender As Object, e As EventArgs) Handles
btnReportDeath.Click
    openFormInPanel(New frmMiscellaneous)
End Sub

Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
    Me.Close()
    FormMain.Show()
End Sub

Private Sub btnViewStolenItems_Click(sender As Object, e As EventArgs) Handles
btnViewStolenItems.Click
    openFormInPanel(New frmViewStolenItems)
End Sub

```

```

Private Sub ViewReportsToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles ViewReportsToolStripMenuItem.Click
    openFormInPanel(New frmViewStolenItems)
End Sub

Private Sub mnuFileReportDeath_Click(sender As Object, e As EventArgs) Handles
mnuFileReportDeath.Click
    openFormInPanel(New frmMiscellaneous)
End Sub

Private Sub btnFileComplaint_Click(sender As Object, e As EventArgs) Handles
btnFileComplaint.Click
    openFormInPanel(New frmFeedback)
End Sub

Private Sub btnFeedback_Click(sender As Object, e As EventArgs) Handles
btnFeedback.Click
    openFormInPanel(New frmFeedback)
End Sub

Private Sub mnuFeedbackGiveUsFeedback_Click(sender As Object, e As EventArgs)
Handles mnuFeedbackGiveUsFeedback.Click
    openFormInPanel(New frmFeedback)
End Sub

Private Sub mnuFileReportComplaint_Click(sender As Object, e As EventArgs) Handles
mnuFileReportComplaint.Click
    openFormInPanel(New frmFeedback)
End Sub

Private Sub mnuFileExit_Click(sender As Object, e As EventArgs) Handles
mnuFileExit.Click
    Me.Close()
    FormMain.Show()
End Sub

Private Sub mnuFileSaveAndLogout_Click(sender As Object, e As EventArgs) Handles
mnuFileSaveAndLogout.Click
    Me.Close()
    FormMain.Show()
End Sub

Private Sub btnReportMurder_Click(sender As Object, e As EventArgs) Handles
btnReportMurder.Click
    openFormInPanel(New frmReportMurder)
End Sub

Private Sub btnReportFraud_Click(sender As Object, e As EventArgs) Handles
btnReportFraud.Click
    openFormInPanel(New frmReportFraud)
End Sub

Private Sub btnReportBurglary_Click(sender As Object, e As EventArgs) Handles
btnReportBurglary.Click
    openFormInPanel(New frmFileBurglaryReport)
End Sub

Private Sub btnBurglaryReports_Click(sender As Object, e As EventArgs) Handles
btnBurglaryReports.Click
    openFormInPanel(New frmViewFiledBurglaryReports)
End Sub

```

```

Private Sub imgMin_Click(sender As Object, e As EventArgs) Handles imgMin.Click
    Me.WindowState = FormWindowState.Minimized
End Sub

Private Sub imgMax_Click(sender As Object, e As EventArgs) Handles imgMax.Click
    Me.WindowState = FormWindowState.Maximized
    Me.imgMax.Visible = False
    Me.imgRes.Visible = True
End Sub

Private Sub imgRes_Click(sender As Object, e As EventArgs) Handles imgRes.Click
    Me.WindowState = FormWindowState.Normal
    Me.imgRes.Visible = False
    Me.imgMax.Visible = True
End Sub

Private Sub imgClo_Click(sender As Object, e As EventArgs) Handles imgClo.Click
    Me.Close()
    FormMain.Show()
End Sub

Private Sub pnlTitle_MouseMove(sender As Object, e As MouseEventArgs) Handles
pnlTitle.MouseMove
    Try
        If e.Button = MouseButton.Left Then
            ReleaseCapture()
            SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
        End If
    Catch ex As Exception

    End Try
End Sub

Private Sub Label2_MouseMove(sender As Object, e As MouseEventArgs) Handles
Label2.MouseMove
    Try
        If e.Button = MouseButton.Left Then
            ReleaseCapture()
            SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
        End If
    Catch ex As Exception

    End Try
End Sub

End Class

Imports System.Data.OleDb

Public Class frmFileBurglaryReport
    Private Sub btnClearAllFields_Click(sender As Object, e As EventArgs) Handles
btnClearAllFields.Click
        clearFields() 'A method that clears input fields
    End Sub

    ''' <summary>
    ''' Clear all the input fields
    ''' </summary>
    Private Sub clearFields()
        txtAddress.Clear()
        txtDescription.Clear()

```

```

End Sub

Private Sub BtnFileReport_Click(sender As Object, e As EventArgs) Handles
btnFileReport.Click
    'Check for the validity of input data before entering it to the database
    'If the data is not valid, show an error message and return
    If txtAddress.Text <> String.Empty Or txtDescription.Text <> String.Empty _
    Or txtDate.Text <> String.Empty Then
        Dim query As String
        query = "Insert into Burglary (CivilianID, DateOfBurglary,
AddressOfBurglary, " &
                "DescriptionOfBurglary, RespondedTo) Values (" & Module1.civilianID &
", '" &
                txtDate.Value.ToShortDateString & "', '" & txtAddress.Text & "', '" &
                txtDescription.Text & "', 'False')"
        Try
            Dim conn As New OleDbConnection(Module1.connectionString)
            Dim cmd As New OleDbCommand(query, conn)
            conn.Open()
            cmd.ExecuteNonQuery()
            conn.Close()
            MessageBox.Show("Your report has been filed successfully.", "Success",
                MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            MessageBox.Show("An error occured while filing your report!", "Fail",
                MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    Else
        MessageBox.Show("Please fill out all fields", "Missing information",
            MessageBoxButtons.OK, MessageBoxIcon.Error)
    End If
End Sub

Private Sub frmFileBurglaryReport_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
    'Load the already existing data from the database
    txtDate.MaxDate = Date.Today 'Set the maximum date of the datetimefield to
today's date
    Dim query As String = "Select FirstName, Surname from CiviliansLoginData where
ID = " & Module1.civilianID
    Try
        Dim conn As New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        Dim reader As OleDbDataReader
        conn.Open()
        reader = cmd.ExecuteReader
        If reader.HasRows Then
            reader.Read()
            txtID.Text = CStr(Module1.civilianID)
            txtName.Text = reader.GetString(0) & " " & reader.GetString(1)
        End If
        conn.Close()
    Catch ex As Exception
        MessageBox.Show("An error occured while loading data", "Error!",
            MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub
End Class

Imports System.Data.OleDb

```

```

Public Class FormFileMissingPerson
    Private Sub btnBrowse_Click(sender As Object, e As EventArgs) Handles
btnBrowse.Click
        OpenFileDialog1.Filter = "Picture Files (*.*)|*.bmp;*.gif;*.jpg" 'Allow opening
of
        'image files only (bitmap, gif and jpeg images)
        If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
            pbPicture.Image = Image.FromFile(OpenFileDialog1.FileName) 'Display the
selected image in the picture box
        End If
    End Sub

    Private Sub btnClearEverything_Click(sender As Object, e As EventArgs) Handles
btnClearEverything.Click
        txtFirstName.Clear()
        txtLastName.Clear()
        txtLastSeenLocation.Clear()
        txtPhoneNumber.Clear()
        txtPostalCode.Clear()
        pbPicture.Image = Nothing
    End Sub

    Private Sub btnSubmitReport_Click(sender As Object, e As EventArgs) Handles
btnSubmitReport.Click
        Dim fName, lName, loc, phone, postal As String
        Dim lastSeenDate As Date
        fName = txtFirstName.Text
        lName = txtLastName.Text
        loc = txtLastSeenLocation.Text
        phone = txtPhoneNumber.Text
        postal = txtPostalCode.Text
        lastSeenDate = dtpLastSeenDate.Value
        'Check whether the input data is provided and is valid. Display a message if
the data is not valid.
        If fName = String.Empty Or lName = String.Empty Or loc = String.Empty _
Or phone = String.Empty Or postal = String.Empty Or pbPicture.Image Is
Nothing Then
            MessageBox.Show("Please fill in all fields", "All fields are required")
            Return
        End If
        Dim bytImage() As Byte
        Try
            'Create a byte array to store the selected image
            Dim ms As New System.IO.MemoryStream
            Dim bmpImage As New Bitmap(pbPicture.Image)
            bmpImage.Save(ms, Imaging.ImageFormat.Jpeg)
            bytImage = ms.ToArray()
            ms.Close()
        Catch ex As Exception
            MsgBox(ex.Message)
            Return
        End Try
        Dim query As String = "Insert into MissingPerson(FirstName, LastName,
CivilianID, LastSeenDate, LastSeenLocation, PhoneNumber, PostalCode, PersonFound,
PersonImage) Values(?, ?, ?, ?, ?, ?, ?, ?, ?)"
        Dim connection As New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand With {
            .Connection = connection,
            .CommandType = CommandType.Text,
            .CommandText = query
        }

```



```

'Add paramaters to the command object created and execute the query
cmd.Parameters.Add(New OleDbParameter("FirstName", fName))
    cmd.Parameters.Add(New OleDbParameter("LastName", lName))
cmd.Parameters.Add(New OleDbParameter("CivilianID", Module1.civilianID))
cmd.Parameters.Add(New OleDbParameter("LastSeenDate",
lastSeenDate.ToShortDateString))
cmd.Parameters.Add(New OleDbParameter("LastSeenLocation", loc))
cmd.Parameters.Add(New OleDbParameter("PhoneNumber", phone))
cmd.Parameters.Add(New OleDbParameter("PostalCode", postal))
Dim found As String = "False"
cmd.Parameters.Add(New OleDbParameter("PersonFound", found))
cmd.Parameters.Add(New OleDbParameter("PersonImage", bytImage))
Try
    connection.Open()
    cmd.ExecuteNonQuery()
    connection.Close()
    MessageBox.Show("Missing person report filed successfully", "Success")
Catch ex As Exception
    MessageBox.Show("Action failed" & vbCrLf & ex.Message &
        vbCrLf & query, "Failed!")
    MessageBox.Show(cmd.CommandText)
End Try
End Sub
End Class

```

```
Imports System.Data.OleDb
```

```
Public Class FormViewMissingPersons
```

```

''' <summary>
''' This method is executed when the form loads.
''' It retrieves missing persons from the database.
''' </summary>
''' <param name="sender">The object that created this event</param>
''' <param name="e">The information about this event</param>

```

```
Private Sub FormViewMissingPersons_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```

    Dim query As String
    query = "Select MissingPerson.ID, MissingPerson.FirstName,
MissingPerson.LastName, " &
        "MissingPerson.PostalCode, MissingPerson.LastSeenDate, " &
        "CiviliansLoginData.ID, CiviliansLoginData.FirstName,
CiviliansLoginData.Surname," &
        "MissingPerson.LastSeenLocation, MissingPerson.PersonImage,
MissingPerson.PersonFound " &
        "From CiviliansLoginData INNER Join MissingPerson On CiviliansLoginData.ID
= " &
        "MissingPerson.CivilianID;"
    Dim connection = New OleDbConnection(Module1.connectionString) 'Create a
connection to the database
    Dim command As New OleDbCommand(query, connection) 'The command to execute
against the above connection
    Try
        connection.Open() 'Open the connection to the database
        Dim reader As OleDbDataReader = command.ExecuteReader()
        If reader.HasRows Then 'If there are rows in the table
            While reader.Read
                Dim id As Integer = CInt(reader.GetValue(0))
                Dim firstName As String = CStr(reader.GetValue(1))
                Dim lastName As String = CStr(reader.GetValue(2))
                Dim code As String = CStr(reader.GetValue(3))
                Dim civilianName As String = CStr(reader.GetValue(6)) &

```

```

        " " & CStr(reader.GetValue(7))
        Dim lastSeenLocation As String = CStr(reader.GetValue(8))
        Dim found As String = CStr(reader.GetValue(10))
        Dim lastSeenOn As String = CStr(reader.GetValue(4))
        Dim row() As Object = {id, firstName, lastName, civilianName,
lastSeenOn,
        lastSeenLocation, code, found}
        dgvMissingPersons.Rows.Add(row)
    End While
Else
    MessageBox.Show("There are no new missing persons yet", "No missing
persons",
        MessageBoxButtons.OK, MessageBoxIcon.Information)
End If
Catch ex As Exception
    MessageBox.Show(ex.Message & vbCrLf & ex.StackTrace)
End Try
End Sub

Private Sub dgvMissingPersons_CellClick(sender As Object, e As
DataGridViewCellEventArgs) Handles dgvMissingPersons.CellClick
    'Show more information about the missing person: where he or she was lost, his
description etc
    For Each item In dgvMissingPersons.SelectedRows
        Dim row As Integer = dgvMissingPersons.Rows.IndexOf(item)
        Dim id As Integer =
CInt(dgvMissingPersons.Rows.Item(row).Cells.Item(0).Value)
        Dim query As String = "Select PersonImage from MissingPerson where ID = "
&
        id
        Try
            Dim conn As New OleDbConnection(Module1.connectionString)
            Dim cmd As New OleDbCommand(query, conn)
            conn.Open()
            Dim reader As OleDbDataReader = cmd.ExecuteReader
            If reader.HasRows Then
                reader.Read()
                Dim pictureData As Byte() = CType(reader.GetValue(0), Byte())
                Dim picture As Image
                Dim stream As New IO.MemoryStream(pictureData)
                picture = Image.FromStream(stream)
                pbPicture.Image = picture
            End If
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        End Try

    Next
End Sub

End Class

```

Imports System.Runtime.InteropServices

```

Public Class OfficialHome
    Public Const WM_NCLBUTTONDOWN As Integer = &HA1
    Public Const HT_CAPTION As Integer = &H2

    <DllImport("user32.dll", SetLastError:=True, CharSet:=CharSet.Auto)>
    Private Shared Function SendMessage(ByVal hWnd As IntPtr,
        ByVal Msg As UInteger,
        ByVal wParam As IntPtr,
        ByVal lParam As IntPtr) As IntPtr
    End Function

    <DllImportAttribute("user32.dll")>
    Public Shared Function ReleaseCapture() As Boolean
    End Function

    Private Sub btnExit_Click(sender As Object, e As EventArgs) Handles btnExit.Click
        Close() 'Close the form and perform proper termination housekeeping
        FormMain.Show()
    End Sub

    Private Sub btnAddOfficial_Click(sender As Object, e As EventArgs) Handles
btnAddOfficial.Click

        pnlMain.Controls.Clear() 'Remoe any existing controls
        Dim frm As New frmAddOfficial 'Declare the form (user control) to be added to
the panel
        frm.Anchor = AnchorStyles.Right + AnchorStyles.Left +
            AnchorStyles.Top + AnchorStyles.Bottom 'Anchor the form properly so that
it resizes with the main panel
            'when the main panel is resized
        frm.Dock = DockStyle.Fill
        pnlMain.Controls.Add(frm) 'Add the form

    End Sub

    Private Sub btnAddPoliceOfficer_Click(sender As Object, e As EventArgs) Handles
btnAddPoliceOfficer.Click
        pnlMain.Controls.Clear()
        Dim frm As New frmAddOfficer
        frm.Anchor = AnchorStyles.Right + AnchorStyles.Left +
            AnchorStyles.Top + AnchorStyles.Bottom
        frm.Dock = DockStyle.Fill
        pnlMain.Controls.Add(frm)
    End Sub

    Private Sub btnDeleteOfficer_Click(sender As Object, e As EventArgs) Handles
btnDeleteOfficer.Click
        pnlMain.Controls.Clear()
        Dim frm As New frmDeleteOfficer
        frm.Anchor = AnchorStyles.Right + AnchorStyles.Left +
            AnchorStyles.Top + AnchorStyles.Bottom
        frm.Dock = DockStyle.Fill
        pnlMain.Controls.Add(frm)
    End Sub

    Private Sub btnViewAllOfficers_Click(sender As Object, e As EventArgs) Handles
btnViewAllOfficers.Click
        pnlMain.Controls.Clear()
        Dim frm As New frmAllOfficers
        frm.Anchor = AnchorStyles.Right + AnchorStyles.Left +
            AnchorStyles.Top + AnchorStyles.Bottom

```

```

        frm.Dock = DockStyle.Fill
        pnlMain.Controls.Add(frm)
    End Sub

    Private Sub imgMin_Click(sender As Object, e As EventArgs) Handles imgMin.Click
        Me.WindowState = FormWindowState.Minimized
    End Sub

    Private Sub imgMax_Click(sender As Object, e As EventArgs) Handles imgMax.Click
        Me.WindowState = FormWindowState.Maximized
        Me.imgMax.Visible = False
        Me.imgRes.Visible = True
    End Sub

    Private Sub imgRes_Click(sender As Object, e As EventArgs) Handles imgRes.Click
        Me.WindowState = FormWindowState.Normal
        Me.imgRes.Visible = False
        Me.imgMax.Visible = True
    End Sub

    Private Sub imgClo_Click(sender As Object, e As EventArgs) Handles imgClo.Click
        Me.Close()
        FormMain.Show()
    End Sub

    Private Sub pnlTitleTREST_MouseMove(sender As Object, e As MouseEventArgs)
        Try
            If e.Button = MouseButton.Left Then
                ReleaseCapture()
                SendMessage(Handle, WM_NCLBUTTONDOWN, HT_CAPTION, 0)
            End If
        Catch ex As Exception

        End Try
    End Sub

    Private Sub PictureBox2_Click(sender As Object, e As EventArgs) Handles
PictureBox2.Click
        If pnlButtons.Width >= 246 Then
            pnlButtons.Width = 86
        Else
            pnlButtons.Width = 246
        End If
    End Sub
End Class

```

```
Imports System.Data.OleDb
```

```
Public Class frmAddOfficer
```

```
Private Sub btnClearAllFields_Click(sender As Object, e As EventArgs)  
    clearFields()  
End Sub
```

```
''' <summary>  
''' Clears all the necessary input fields  
''' </summary>
```

```
Private Sub clearFields()  
    txtOfficerIDNumber.Clear()  
    txtOfficerPassword.Clear()  
    txtOfficerPhoneNumber.Clear()  
    txtOfficerConfirmPassword.Clear()  
    txtOfficerFirstName.Clear()  
    txtOfficerLastName.Clear()  
End Sub
```

```
Private Sub btnCl_Click(sender As Object, e As EventArgs)
```

```
    Dim id, p1, p2, phone, fname, lname As String
```

```
    id = txtOfficerIDNumber.Text
```

```
    p1 = txtOfficerPassword.Text
```

```
    p2 = txtOfficerConfirmPassword.Text
```

```
    phone = txtOfficerPhoneNumber.Text
```

```
    fname = txtOfficerFirstName.Text
```

```
    lname = txtOfficerLastName.Text
```

```
    'Check for data validity.
```

```
    'If data is not valid, display a message and return.
```

```
    'If it is valid, add the data to the database.
```

```
    If id = String.Empty Or p1 = String.Empty Or phone = String.Empty _  
        Or fname = String.Empty Or lname = String.Empty Then  
        MessageBox.Show("Please fill out all fields", "Missing fields")  
        Return  
    End If
```

```
    If p1 <> p2 Then  
        MessageBox.Show("The two passwords do not match.", "Passwords!")  
        Return  
    End If
```

```
    If txtOfficerDateOfBirth.Text = String.Empty Then  
        MessageBox.Show("Please fill out all fields", "Missing fields")  
        Return  
    End If
```

```
    Dim query As String  
    query = "Insert into Officer (FirstName, LastName, IDNumber,  
DateOfBirth, " &
```

```
        "PhoneNumber, OfficerPassword) Values('" & fname & "', '" & lname  
& "', '" &  
        id & "', '" & txtOfficerDateOfBirth.Value.ToShortDateString & "',  
        '" &  
        phone & "', '" & p1 & "')
```

```
    Try
```

```
    Dim conn As New OleDbConnection(Module1.connectionString)
```

```
    Dim cmd As New OleDbCommand(query, conn)
```

```
    conn.Open()
```

```
    cmd.ExecuteNonQuery()
```

```
    conn.Close()
```

```
    MessageBox.Show("Your officer has been added successfully", "Success!")
```

```
Catch ex As Exception
```

```
    MessageBox.Show("An error occured: " & vbCrLf &  
        ex.Message, "Error!")
```

```
End Try
```

```
End Sub
```

End Class

Imports System.Data.OleDb

Public Class frmAddOfficial

```
    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles
btnClear.Click
        clearFields() 'Clear the input fields (textboxes and stuff)
    End Sub
    ''' <summary>
    ''' Clears all the necessary input fields
    ''' </summary>
    Private Sub clearFields()
        txtConfirmPassword.Clear()
        txtPassword.Clear()
        txtOfficialName.Clear()
    End Sub

    Private Sub btnAdd_Click(sender As Object, e As EventArgs) Handles btnAdd.Click
        Dim p1, p2, name As String
        p1 = txtPassword.Text
        p2 = txtConfirmPassword.Text
        name = txtOfficialName.Text
        'Check for data validity. If the data is valid, proceed; else, display an
        'error message and return without doing anything
        If p1 = String.Empty Or p2 = String.Empty Or name = String.Empty Then
            MessageBox.Show("Please fill all fields", "Missing information")
            Return
        End If
        If p1 <> p2 Then
            MessageBox.Show("The passwords you have provided do not match!",
                "Different passwords")
            Return
        End If
        Dim query As String
        query = "Insert into OfficialsLoginData Values('" &
            name & "', '" & p1 & "')"
        Try
            Dim conn As New OleDbConnection(Module1.connectionString)
            Dim cmd As New OleDbCommand(query, conn)
            conn.Open()
            cmd.ExecuteNonQuery()
            conn.Close()
            MessageBox.Show("The official has been added successfully",
                "Success")
        Catch ex As Exception
            MessageBox.Show("An error occured: " &
                vbCrLf & ex.Message, "Fail!",
                MessageBoxButtons.OK,
                MessageBoxIcon.Error)
        End Try
    End Sub
End Class
```

```
Imports System.Data.OleDb
```

```
''' <summary>  
''' This class enables an officer to view all officers that are in the system  
''' </summary>  
Public Class frmAllOfficers  
  
    ''' <summary>  
    ''' Load officers that exist in the database  
    ''' </summary>  
    ''' <param name="sender"></param>  
    ''' <param name="e"></param>  
    Private Sub frmAllOfficers_Load(sender As Object, e As EventArgs) Handles Me.Load  
        Dim query As String = "Select * from Officer "  
        Try  
            Dim conn As New OleDbConnection(Module1.connectionString)  
            Dim cmd As New OleDbCommand(query, conn)  
            conn.Open()  
            Dim reader As OleDbDataReader  
            reader = cmd.ExecuteReader  
            Dim id, fName, lName, idNumber, dob, phone, password  
            While reader.Read  
                id = reader.GetValue(0).ToString  
                fName = reader.GetString(1)  
                lName = reader.GetString(2)  
                idNumber = reader.GetString(3)  
                dob = reader.GetDateTime(4).ToShortDateString()  
                phone = reader.GetString(5)  
                password = reader.GetString(6)  
                Dim obj As Object() = {id, fName, lName, idNumber, dob, phone,  
password}  
                dgvOfficers.Rows.Add(obj)  
            End While  
        Catch ex As Exception  
            MessageBox.Show("An error occured", "Error", MessageBoxButtons.OK,  
MessageBoxIcon.Error)  
        End Try  
    End Sub  
End Class
```

```
Imports System.Data.OleDb
```

```
''' <summary>  
''' Load any feedback that might be existing in the database, and enable the user  
''' to submit his feedback  
''' </summary>  
Public Class frmFeedback  
    Private Sub frmFeedback_Load(sender As Object, e As EventArgs) Handles  
MyBase.Load  
        Dim query As String  
        query = "SELECT CiviliansLoginData.FirstName, CiviliansLoginData.Surname  
" &  
        "FROM CiviliansLoginData INNER JOIN StolenProperty ON  
CiviliansLoginData.ID " &  
        "= StolenProperty.CivilianID where CiviliansLoginData.ID = " &  
Module1.civilianID  
        query = "Select FirstName, Surname from CiviliansLoginData where ID = "  
&  
        civilianID  
        Try  
            Dim conn As New OleDbConnection(Module1.connectionString)
```

```

        Dim cmd As New OleDbCommand(query, conn)
        Dim reader As OleDbDataReader
        conn.Open()
        reader = cmd.ExecuteReader
        If reader.HasRows Then
            reader.Read()
            txtID.Text = Module1.civilianID
            txtName.Text = reader.GetString(0) & " " &
reader.GetString(1)
        End If
        conn.Close()
    Catch ex As Exception
        MessageBox.Show("An error occured while loading data from the
database",
                        "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End Try
End Sub

Private Sub btnSubmit_Click(sender As Object, e As EventArgs) Handles
btnSubmit.Click
    If txtOccurrence.Text = String.Empty Then
        txtOccurrence.Text = "This field is required!"
        Return
    End If
    Dim query As String
    query = "Insert into Feedback (CivilianID, FDescription, FDate) Values("
&
&
        Module1.civilianID & ", '" & txtOccurrence.Text.Replace("'", "")
        "', '" & Date.Today.ToShortDateString() & "')"
    Try
        Dim conn = New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        cmd.ExecuteNonQuery()
        conn.Close()
        MessageBox.Show("Thank you for your feedback.", "Thank you",
            MessageBoxButtons.OK, MessageBoxIcon.Information)
    Catch ex As Exception
        MessageBox.Show("We are sorry, but an error has occured. ",
            "Please try again later!", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End Try
End Sub
End Class

Imports System.Data.OleDb

Public Class Feedback
    Private Sub btnClear_Click(sender As Object, e As EventArgs) Handles
btnClear.Click
        txtFeedback.Clear()
    End Sub

    Private Sub LoadData()
        Dim query As String
        query = "SELECT Officer.FirstName, Officer.LastName, " &
            "OfficerFeedback.OFeedback FROM Officer INNER JOIN " &
            "OfficerFeedback ON Officer.ID = OfficerFeedback.OfficerID"
    Try

```



```

        Dim conn As New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        Dim reader = cmd.ExecuteReader()
        If reader.HasRows Then
            lstFeedback.Items.Clear()
            While reader.Read()
                Dim data As String
                data = ""
                data &= reader.GetString(2)
                lstFeedback.Items.Add(data)
                data = vbTab & reader.GetString(0) & " " &
reader.GetString(1)
                lstFeedback.Items.Add(data)
            End While
        End If
    Catch ex As Exception
        MessageBox.Show("An error occured: " & vbCrLf &
ex.Message, "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub

Private Sub btn_Click(sender As Object, e As EventArgs) Handles btn.Click
    If txtFeedback.Text = String.Empty Then
        Return
    End If
    Dim f = txtFeedback.Text
    f = f.Replace("'", "")
    Dim query As String = "Insert into OfficerFeedback (OfficerID,
OFeedback) " &
        " Values(" & Module1.officerID & ", '" & f & "')"
    'query = "Insert into OfficerFeedback (OfficerID, OFeedback) " &
'
' Values(" & Module1.officerID & ", '" & f & "')"
    Try
        Dim conn As New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        cmd.ExecuteNonQuery()
        conn.Close()
        MessageBox.Show("Feedback submitted successfully", "Success",
            MessageBoxButtons.OK, MessageBoxIcon.Information)
        LoadData()
    Catch ex As Exception
        MessageBox.Show("Feedback submission failed!", "Error",
            MessageBoxButtons.OK, MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Feedback_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    LoadData()
End Sub
End Class

Imports System.ComponentModel
Imports System.Data.OleDb

Public Class frmViewUserFeedback

    Private Sub frmViewUserFeedback_Load(sender As Object, e As EventArgs) Handles
MyBase.Load

```

```

        LoadData()
End Sub

''' <summary>
''' Attempts to load the feedback data that might exist in the database.
''' If no data exists, this method simply does nothing.
''' If an error occurs, this method will throw and catch an <c>Exception</c>
''' and inform the user of the exception information.
''' </summary>
Private Sub LoadData()
    Dim query As String
    query = "SELECT Feedback.ID, CiviliansLoginData.FirstName, " &
"CiviliansLoginData.Surname, Feedback.FDate, " &
"Feedback.FDescription FROM CiviliansLoginData " &
"INNER JOIN Feedback ON CiviliansLoginData.ID = Feedback.CivilianID"
    Try
        Dim conn As New OleDbConnection(connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        Dim reader = cmd.ExecuteReader()
        If reader.HasRows Then
            dgvData.Rows.Clear()
            While reader.Read()
                Dim id = CInt(reader.GetValue(0)) 'The id of the current row
                Dim fName = reader.GetString(1) 'The first name of the person who
submitted the current feedback
                Dim lName = reader.GetString(2) 'The last name of the person
                Dim dataF = reader.GetDateTime(3).ToShortDateString() 'The date
this feedback was submitted
                Dim desc = reader.GetString(4) 'The actual feedback
                Dim data() As Object = {id, fName, lName, dataF, desc}
                dgvData.Rows.Add(data) 'Add the row to the datagrid view
            End While
        End If
        Catch ex As Exception
            MessageBox.Show("An error has occurred: " & vbCrLf & ex.Message,
                "Error!", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub

''' <summary>
''' Retrieves the actual feedback of the selected row.
''' Throws and catches an exception if an error occurs.
''' The user is informed of the exception information
''' </summary>
''' <param name="sender"></param>
''' <param name="e"></param>
Handles dgvData.CellClick
Private Sub dgvData_CellClick(sender As Object, e As DataGridViewCellEventArgs)
    Dim id = CInt(dgvData.SelectedRows.Item(0).Cells.Item(0).Value)
    Dim query = "Select FDescription from Feedback where ID = " & id
    Try
        Dim conn As New OleDbConnection(connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        Dim reader = cmd.ExecuteReader()
        If reader.HasRows Then
            reader.Read()
            txtDescription.Text = reader.GetString(0)
        End If
    End Try
End Sub

```

```

        End If
    Catch ex As Exception
        MessageBox.Show("An error has occurred: " & vbCrLf & ex.Message,
            "Error!", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub
End Class

```

```
Imports System.Data.OleDb
```

```
Public Class frmMurders
```

```
    Private Sub frmMurders_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
        LoadData() 'Display in the datagrid view any data that might be in the database
    End Sub

```

```

    Private Sub LoadData()
        Dim query As String
        query = "SELECT Murder.ID, Murder.VictimFName, " &
            "Murder.VictimLName, Murder.Location, Murder.DateM, " &
            "Murder.DescriptionM, CiviliansLoginData.FirstName, " &
            "CiviliansLoginData.Surname " &
            "FROM CiviliansLoginData INNER JOIN Murder " &
            "ON CiviliansLoginData.ID = Murder.CivilianID"

        'Attempt to read data from the database, throwing an exception in case
        anything goes wrong
        Try
            Dim conn As New OleDbConnection(connectionString)
            Dim cmd As New OleDbCommand(query, conn)
            conn.Open()
            Dim reader = cmd.ExecuteReader()
            If reader.HasRows Then
                dgvData.Rows.Clear()
                While reader.Read()
                    Dim id = CInt(reader.GetValue(0))
                    Dim fNameV = reader.GetString(1) 'First name of the victim (the
                    person reported as being murdered)
                    Dim lNameV = reader.GetString(2) 'Last name of the victim
                    Dim loc = reader.GetString(3) 'The location that the victim was
                    murdered, or found dead
                    Dim dateM = reader.GetDateTime(4).ToShortDateString() 'The date of
                    the reported murder
                    Dim desc = reader.GetString(5) 'The description of the scene of
                    murder, or the body etc
                    Dim fName = reader.GetString(6) 'The first name of the person who
                    filed this report
                    Dim lName = reader.GetString(7) 'The last name of the person who
                    filed this report

                    'Create an array to contain all the above information
                    Dim data() As Object = {
                        id, fNameV, lNameV, fName, lName, dateM,
                        desc,
                        loc}
                    dgvData.Rows.Add(data) 'Add a row containing the above data to the
                    datagrid view
                End While
            End If
            conn.Close()
        End Try
    End Sub

```

```

        Catch ex As Exception
            MessageBox.Show("An error occured: " & vbCrLf & ex.Message,
                "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub

    Private Sub dgvData_CellClick(sender As Object, e As DataGridViewCellEventArgs)
        Handles dgvData.CellClick
            'Show the description of the murder scene, victim etc, of the row that has
            been selected in
            'the datagrid view
            Dim id = CInt(dgvData.SelectedRows.Item(0).Cells.Item(0).Value)
            Dim query = "Select DescriptionM from Murder where ID = " & id
            'Attempt to retrieve the data, throwing an exception if things are not
            'how we expect them to be
            Try
                Dim conn As New OleDbConnection(connectionString)
                Dim cmd As New OleDbCommand(query, conn)
                conn.Open()
                Dim reader = cmd.ExecuteReader() 'Retrieve the data returned by executing
                the
                'Select query above
                If reader.HasRows Then
                    reader.Read()
                    txtDescription.Text = reader.GetString(0)
                End If
                conn.Close()
            Catch ex As Exception
                MessageBox.Show("An error occured: " & vbCrLf & ex.Message,
                    "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End Sub
    End Class

```

```
Imports System.Data.OleDb
```

```

''' <summary>
''' Filing a miscellaneous report;
''' A miscellaneous report is a report that does not fall under
''' the categories such as murder, burglary, death etc
''' </summary>
Public Class frmMiscellaneous
    Private Sub frmMiscellaneous_Load(sender As Object, e As EventArgs) Handles
        MyBase.Load
            'Load the first name and last name and ID of the currently logged in citizen
            'Display an error message in case of any exception
            Dim query As String
            query = "SELECT CiviliansLoginData.FirstName, CiviliansLoginData.Surname " &
                "FROM CiviliansLoginData INNER JOIN StolenProperty ON CiviliansLoginData.ID "
            &
                "= StolenProperty.CivilianID where CiviliansLoginData.ID = " &
                Module1.civilianID
            Try
                Dim conn As New OleDbConnection(Module1.connectionString)
                Dim cmd As New OleDbCommand(query, conn)
                Dim reader As OleDbDataReader
                conn.Open()
                reader = cmd.ExecuteReader
                If reader.HasRows Then
                    reader.Read()
                    txtID.Text = Module1.civilianID
                End If
            Catch ex As Exception
                MessageBox.Show("An error occured: " & vbCrLf & ex.Message,
                    "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End Sub
    End Class

```

```

        txtName.Text = reader.GetString(0) & " " & reader.GetString(1)
    End If
    conn.Close()
Catch ex As Exception
    MessageBox.Show("An error occured while loading data from the database",
        "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

End Sub

Private Sub btnSubmit_Click(sender As Object, e As EventArgs) Handles
btnSubmit.Click
    'Submit the input information
    'Simply return (do nothing, if the citizen has not provided any information)
    If txtDescription.Text = String.Empty Then
        Return
    End If
    Dim query As String
    Dim text As String = txtDescription.Text
    text = text.Replace("'", "")
    query = "Insert into Miscellaneous (CivilianID, MDescription, MDate) " &
        "Values (" & Module1.civilianID & ", '" & txtDescription.Text & "', '" &
        txtDate.Value.ToShortDateString & "')"
    Try
        Dim conn As New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        cmd.ExecuteNonQuery()
        conn.Close()
        MessageBox.Show("Your miscellaneous report has been filed successfully",
            "Success", MessageBoxButtons.OK, MessageBoxIcon.Information)
    Catch ex As Exception
        MessageBox.Show("We could not handle your submission now. Please " &
            "try again later",
            "Failed!", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub
End Class

```

```
Imports System.Data.OleDb
```

```
Public Class frmDeleteOfficer
```

```

    Dim conn As OleDbConnection 'Connection object to the database
    Dim reader As OleDbDataReader 'Reader to contain data that is read from the
    database

```

```

Private Sub loadData()
    Dim query As String = "Select * from Officer"

    Try
        conn = New OleDbConnection(Module1.connectionString)
        Dim cmd As New OleDbCommand(query, conn)
        conn.Open()
        reader = cmd.ExecuteReader
    Catch ex As Exception
        MessageBox.Show("An error occured: " & vbCrLf &
            ex.Message, "Error!",
            MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub

```

```

        Private Sub frmDeleteOfficer_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
            loadData() 'Refresh the data in the database
        End Sub

        Private Sub btnNext_Click(sender As Object, e As EventArgs) Handles
btnNext.Click
            'Attempt to retrieve the next record from the database.
            If reader.Read() Then
                txtDateofBirth.Text = reader.GetDateTime(4).ToShortDateString()
                txtFirstName.Text = reader.GetString(1)
                txtLastName.Text = reader.GetString(2)
                txtID.Text = reader.GetValue(0).ToString
                txtIDNumber.Text = reader.GetString(3)
                txtPassword.Text = reader.GetString(6)
                txtPhoneNumber.Text = reader.GetString(5)
            Else
                loadData() 'Load the data again if we have reached the last record in the
database
            End If
        End Sub

        ''' <summary>
        ''' Clear all the input fields
        ''' </summary>
        Private Sub clearFields()
            txtDateofBirth.Clear()
            txtFirstName.Clear()
            txtID.Clear()
            txtIDNumber.Clear()
            txtLastName.Clear()
            txtPassword.Clear()
            txtPhoneNumber.Clear()
        End Sub

        Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
            'Delete the selected record

            If txtID.Text = String.Empty Then
            Else
                Dim query As String = "Delete from Officer where ID = " &
CInt(txtID.Text)
                Try
                    Dim connection As New OleDbConnection(Module1.connectionString)
                    Dim cmd As New OleDbCommand(query, connection)
                    connection.Open()
                    cmd.ExecuteNonQuery()
                    connection.Close()
                    MessageBox.Show("Officer deleted successfully!", "Success")
                    loadData()
                    clearFields()
                Catch ex As Exception
                    MessageBox.Show("An error occured", "Fail!")
                End Try
            End If
        End Sub
    End Class

```

```
Imports System.Data.OleDb
```

```
Public Class frmViewCriminals
```

```
    Private Sub btnFirstName_Click(sender As Object, e As EventArgs) Handles  
btnFirstName.Click  
        'Search for a criminal whose first name starts with (or contains) the letters  
in the  
        'first name text box  
        lblFirstName.Text = ""  
        If txtFirstName.Text = String.Empty Then  
            lblFirstName.Text = "Please provide the first name"  
            Return  
        End If  
        dgvData.Rows.Clear()  
        LoadFirstName()  
    End Sub
```

```
    Private Sub btnLastName_Click(sender As Object, e As EventArgs) Handles  
btnLastName.Click  
        'Search for a criminal whose first name starts with (or contains) the letters  
in the  
        'first name text box  
        lblLastName.Text = ""  
        If txtLastName.Text = String.Empty Then  
            lblLastName.Text = "Please provide the last name"  
            Return  
        End If  
        dgvData.Rows.Clear()  
        LoadLastName()  
    End Sub
```

```
    Private Sub btnIDNumber_Click(sender As Object, e As EventArgs) Handles  
btnIDNumber.Click  
        'Search for a criminal whose first name starts with (or contains) the letters  
in the  
        'first name text box  
        lblIDNumber.Text = ""  
        If txtIDNumber.Text = String.Empty Then  
            lblIDNumber.Text = "Please provide the ID Number"  
            Return  
        End If  
        dgvData.Rows.Clear()  
        LoadID()  
    End Sub
```

```
    Private Sub LoadData()  
        'Load all criminals, without considering any criteria. Simply load the  
criminals  
        Dim query As String  
        query = "SELECT Criminal.ID, Criminal.FirstName, Criminal.LastName, " &  
"Criminal.IDNumber, Criminal.DateOfBirth, Criminal.YearsServed FROM Criminal"  
        Try  
            Dim conn As New OleDbConnection(connectionString)  
            conn.Open()  
            Dim cmd As New OleDbCommand(query, conn)  
            Dim reader = cmd.ExecuteReader()  
            If reader.HasRows Then  
                dgvData.Rows.Clear()  
                While reader.Read()  
                    Dim id = CInt(reader.GetValue(0)) 'The ID of this record  
                    Dim fName = reader.GetString(1) 'The first name of the criminal
```

```

        Dim lName = reader.GetString(2) 'The last name of the criminal
        Dim idNumber = reader.GetString(3) 'The ID number of the criminal
        Dim dob = reader.GetDateTime(4).ToShortDateString() 'The date of
        birth of the criminal
        Dim yearsServed = CInt(reader.GetValue(5)) 'The number of years
        this criminal has served before
        Dim data() As Object = {id, fName, lName, idNumber, dob,
yearsServed}
        dgvData.Rows.Add(data)
    End While
End If
Catch ex As Exception
    MessageBox.Show("An error occurred while loading data: " &
        vbCrLf & ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
End Try
End Sub

Private Sub frmViewCriminals_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
    LoadData() 'Load the criminal data that might be existing in the database
End Sub

Private Sub LoadFirstName()
    'Search for criminals whose first names contain the character(s) in the first
name textbox
    Dim query As String
    Dim fName = txtFirstName.Text.Replace("'", "")
    query = "SELECT Criminal.ID, Criminal.FirstName, Criminal.LastName, " &
        "Criminal.IDNumber, Criminal.DateOfBirth, Criminal.YearsServed FROM Criminal"
    &
        " where Criminal.FirstName Like '%" & fName & "%'"
    Try
        Dim conn = New OleDbConnection(connectionString)
        conn.Open()
        Dim cmd As New OleDbCommand(query, conn)
        Dim result = cmd.ExecuteNonQuery()
        Dim reader = cmd.ExecuteReader()
        If reader.HasRows Then
            dgvData.Rows.Clear()
            While reader.Read()
                Dim id = CInt(reader.GetValue(0))
                fName = reader.GetString(1)
                Dim lName = reader.GetString(2)
                Dim idNumber = reader.GetString(3)
                Dim dob = reader.GetDateTime(4).ToShortDateString()
                Dim yearsServed = CInt(reader.GetValue(5))
                Dim data() As Object = {id, fName, lName, idNumber, dob,
yearsServed}
                dgvData.Rows.Add(data)
            End While
        Else
            End If
        conn.Close()
    Catch ex As Exception
        MessageBox.Show("An error occurred while loading data: " &
            vbCrLf & ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End Try
End Sub

Private Sub LoadLastName()

```



```

        'Search for criminals whose last names contain the character(s) in the last
name textbox'Search for
        Dim query As String
        Dim lName = txtLastName.Text.Replace("'", "")
        query = "SELECT Criminal.ID, Criminal.FirstName, Criminal.LastName, " &
"Criminal.IDNumber, Criminal.DateOfBirth, Criminal.YearsServed FROM Criminal"
&
        " where Criminal.LastName Like '%" & lName & "%'"
        Try
            Dim conn As New OleDbConnection(Module1.connectionString)
            Dim cmd As New OleDbCommand(query, conn)
            conn.Open()
            Dim reader = cmd.ExecuteReader()
            If reader.HasRows Then
                dgvData.Rows.Clear()
                While reader.Read()
                    Dim id = CInt(reader.GetValue(0))
                    Dim fName = reader.GetString(1)
                    lName = reader.GetString(2)
                    Dim idNumber = reader.GetString(3)
                    Dim dob = reader.GetDateTime(4).ToShortDateString()
                    Dim yearsServed = CInt(reader.GetValue(5))
                    Dim data() As Object = {id, fName, lName, idNumber, dob,
yearsServed}
                    dgvData.Rows.Add(data)
                End While
            End If
            Catch ex As Exception
                MessageBox.Show("An error occured while loading data: " &
vbCrLf & ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
            End Try
        End Sub

        Private Sub LoadID()
            'Search for criminals whose ID numbers contain the character(s) in the ID
number textbox
            Dim query As String
            Dim fName = txtIDNumber.Text.Replace("'", "")

            query = "SELECT Criminal.ID, Criminal.FirstName, Criminal.LastName, " &
"Criminal.IDNumber, Criminal.DateOfBirth, Criminal.YearsServed FROM Criminal "
&
            "where Criminal.IDNumber Like '%" & fName & "%'"
            Try
                Dim conn As New OleDbConnection(Module1.connectionString)
                Dim cmd As New OleDbCommand(query, conn)
                conn.Open()
                Dim reader = cmd.ExecuteReader()
                If reader.HasRows Then
                    dgvData.Rows.Clear()
                    While reader.Read()
                        Dim id = CInt(reader.GetValue(0))
                        fName = reader.GetString(1)
                        Dim lName = reader.GetString(2)
                        Dim idNumber = reader.GetString(3)
                        Dim dob = reader.GetDateTime(4).ToShortDateString()
                        Dim yearsServed = CInt(reader.GetValue(5))
                        Dim data() As Object = {id, fName, lName, idNumber, dob,
yearsServed}
                        dgvData.Rows.Add(data)
                    End While
                End If
            End Try
        End Sub
    End Sub

```

```

        End If
    Catch ex As Exception
        MessageBox.Show("An error occured while loading data: " &
            vbCrLf & ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End Try
End Sub

Private Sub txtFirstName_TextChanged(sender As Object, e As EventArgs) Handles
txtFirstName.TextChanged
    'Search for criminals whose first names contain the character(s) in the first
name textbox
    dgvData.Rows.Clear()
    LoadFirstName()
End Sub

Private Sub txtIDNumber_TextChanged(sender As Object, e As EventArgs) Handles
txtIDNumber.TextChanged
    'Search for criminals whose IDs contain the character(s) in the ID textbox
    dgvData.Rows.Clear()
    LoadID()
End Sub

Private Sub txtLastName_TextChanged(sender As Object, e As EventArgs) Handles
txtLastName.TextChanged
    'Search for criminals whose last names contain the character(s) in the last
name textbox
    dgvData.Rows.Clear()
    LoadLastName()
End Sub
End Class

```

## Testing

## **Evaluation**