National College of Ireland

BSHCYB4

Cybersecurity

2021/2022

James Kawala

X19330941

X19330941@student.ncirl.ie

Encrypto

Technical Report

Contents

Executive Summary

In this modern era companies and individuals are becoming more aware about device security and are heavily investing more resources to create or utilise cybersecurity applications that contain a number of useful forensic tools. These tools can include file analysis, network

analysis, encryption services and registry editor. Being able to track sensitive data, checking data integrity and encrypting sensitive file is an effective way to check, manage and secure any computer system.

My general interest in cybersecurity pushed me in creating an application that provide a user with comprehensive security and knowledge for their devices. I decided on focusing on creating a windows application that focuses on giving users a better understanding and security to their system by providing a range of powerful tools for analysing and protecting against potential threats. This application is called Encrypto, and it will provide the user with a host of modern-day forensic tools.

In this report, I will summarise my project, the background behind the idea of my project, the technologies I used, the system requirements and the applications overall functionalities. I will also highlight the system architecture, execution, GUI, testing of the application and if there is any further development needed for the project.

At the end of this report the user will have a better understanding of how to utilise Encrypto and how to better secure and encrypt their data to avoid security attacks or any risks. Customers using Encrypto will be able to make use of its fully fleshed components which allows them to quickly extract sensitive information pertinent to their organisation or individual needs. They will be able to narrow down the vast quantity of information such as file properties, network, and hash information. Additionally, users will have a better understanding of what security measures they need to implement on their devices to better secure themselves from potential threats.

1.0 **Introduction**

1.1. *Background*

The reason I choose to undertake this project was because of my personal interest in encryption software and also my interest in digital security. During my four years in college, I was lucky enough to experience using a variety of different digital forensic applications such as Recuva, FTK imager and Wireshark. Dealing with some of these tools gave me a general idea of what I was going to implement for my own final year project. After a couple weeks of research, I saw a gap in the market for an application that could provide a user with information and tools for analysing sensitive data and visualising a systems security. I decided on visualising, developing, and releasing a product that could inform the user of the files and software contained in their device and to provide the user with a number of modern-day forensic tools to better secure their device data.

1.2. *Aims*

The aim of the project is to create an application that would be able to provide users with a set of tools for analysing and securing their computer systems. The name of the application is called Encrypto. This application will have a number of forensic kit tools that will analyse system information and user data.

These forensic tools are designed to be used in a careful and methodical manner to ensure that data is collected and analysed in a way that preserves its integrity and accuracy either for personal use or law enforcement. There are four specific aims and functionalities in my application:

- **File analysis:** The tool will be able to collect data from a variety of sources and also give the user a number of options on where and how to collect the file data. The tool should also be able to export detailed reports and other documentation that can be used as evidence in court or to keep track of user information. The capabilities of the application can help users identify potentially malicious files or suspicious activity on their system.

- **Registry analysis:** This tool can provide insights into how programs and system components are interacting with the Windows Registry, which can help identify changes made by malware or other malicious activity. The tool will be able to analyse any data in a way that preserves its integrity and allows users to identify relevant information.

- **Network analysis:** This tool can perform an IP address lookup. IP address lookup can be used to troubleshoot network issues by identifying the IP address of a device or server and determining if it is accessible or if there are any connectivity issues. This tool can help identify potential network-based threats, such as malicious traffic or unauthorized access attempts.

- **MD5 analysis:** This tool can be used to calculate the MD5 (Message Digest 5) of any given file and also convert one or multiple files into MD5 data. The MD5 calculator and changer feature can help users verify the integrity of files and detect changes made to them, which is useful for maintaining the security and integrity of important data.

- **Encryption Process:** The aim of this encryption tool is to provide users with a way to analyse encrypted data and identify potential vulnerabilities or weaknesses in the encryption scheme. The encryption and decryption capabilities can help users protect sensitive information from unauthorized access.

1.3. *Technology*

The technologies I utilised during the development of my include:

- **Microsoft Visual studio:** The text editor I utilised for my application is Microsoft Visual Studio 2017. This text editor is a popular integrated development environment (IDE) used for developing a wide range of software applications. Visual Studio provides a powerful set of tools and features for coding, debugging, testing, and deploying applications. Visual Studio also includes a wide range of extensions and plugins that helped the development experience and improved my productivity. I used the programming language C# and Visual Basic .NET to create my application.
- **Trello:** This is a web-based application that is used for organisation, list making and project management. I used this application to manage my project objectives and to create a good project timeline.
- **Microsoft Teams:** This is a collaboration and communication platform that allows individuals and teams to chat, make audio and video calls, share files, and collaborate on projects. I used this software to communicate with my project supervisor.

- **Git and GitHub:** GIT and GitHub is a control system and internet hosting service tools. I used these platforms to create a repository, version control and test my application. I also used GitHub to gather useful API's and features that would complement my windows application.
- **Draw. Io:** This is a web-based diagramming tool used for creating diagrams such as flowcharts, process diagrams, network diagrams, UML diagrams, and more. I used this software to create design layouts for my application. I also used it to create UML diagrams for applications documentation.
- **.NET Framework (API):** When I was developing my applications using C#, I used the .NET Framework, which includes a diverse collection of APIs (Application Programming Interfaces) for various purposes, including networking, security, database access, and more. I found the documentation and download the .NET Framework from the Microsoft website.
- **NI MAX:** National Instruments Measurement and Automation Explorer, is a software application that is commonly used in the field of measurement and automation. It allows me to configure and troubleshoot devices that are connected to my computer, such as data acquisition devices, digital I/O devices, and others.

The application that I developed is running on windows OS and is compatible with 340 file types of files (MS document, PDF, Cip, and RAR etc).

1.4. *Market*

The features utilised in this application, would be categorised as a digital forensic software tool. The market for a digital forensic tool can be quite varied, as there are many different industries and sectors that can benefit from this type of software. Some potential markets for my application might include:

- **Law enforcement:** One of the primary markets for my application is law enforcement agencies (local, state, and federal law enforcement) . These agencies often need to conduct digital investigations as part of criminal investigations and need tools that can help them acquire and analyse digital evidence for legal or corporate cases.
- **Corporate investigations:** Many corporations have their own internal investigation teams that need digital forensic tools to help them investigate potential fraud, intellectual property theft, or other incidents that may require the analysis of digital evidence.
- **Legal firms:** Law firms may also need forensic tools to assist with electronical discovery and other legal proceedings that require the analysis of digital evidence.
- **Government agencies:** Government agencies at the local, state, and federal levels may also have a need for digital forensic tools to investigate cybercrime or other digital incidents.
- **Cybersecurity firms:** Cybersecurity firms may use forensic tools as part of their incident response processes to investigate and respond to cyberattacks.

The main competitor of my application in the forensic tool market, would be software like Passware, Prodiscover Forensic or Recuva etc.

The main difference between these products and Encrypto includes:

- Inclusion of diverse types of forensic features. For Example, Passware only does file analysis and recovery while my application Encrypto has network analysis and file analysis.
- The user interface of my application ls more intuitive/ user-friendly than competing tools, making it easier for investigators to use and navigate.
- My application is more efficient than other tools on the market, allowing investigators to analyse and process digital evidence more quickly.
- My application is more affordable or accessible than other digital forensic tools on the market, particularly for smaller organizations or individual investigators.

2.0 System

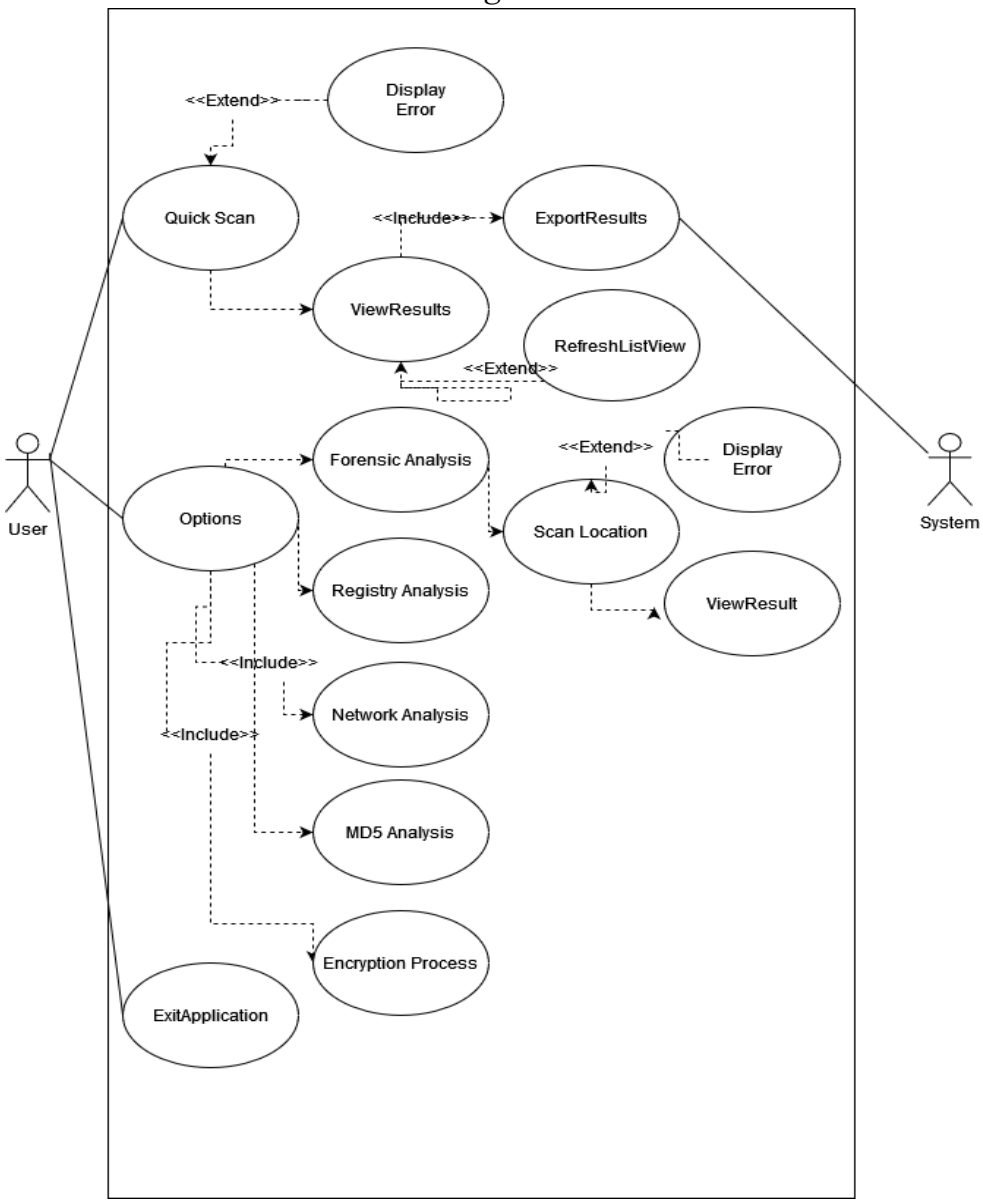
2.1. *Requirements*

The purpose of this section is to set out the requirements for the development of my application called Encrypto. The application's system shall consist of two main architectural components – data analysis/collection, where the data from the user's computer will be processed and analysed; and data exporting, where the user can collect and view the data that has been collected and split.

2.1.1. *Functional Requirements: System Analysis*

Feature	Requirement Description
Scan Options	The system should be able to allow the user to choose what type of scanning and analyses tool they want to use
Scan location	The system should allow the user to pick where they want the scanning and analysis to take place. E.g., System folder, computer drive, all local drives or web applications.
Quick Scan	The system should allow the user to quickly run a general analysis on their system.
ExitApplication	The user should be able to close the application at any moment.
ViewResults	The system should create a detailed report of the analysed data that the user can view
DisplayError	The system should be able to show the user an error message and what is the cause of that error.
ExportResults	The user should be able to manipulate data and export it as a txt or csv file.

2.1.1.1. *Use Case Diagram*



2.1.1.2. *Requirement 1: User scans system*

Use Case Name:	UserScanSystem
Description:	This use case allows the user to scan the system for any sensitive data.
Actors: Primary and Secondary:	Primary actor is the user who utilises the application. Secondary actor is the application (System).
Triggers:	The system starts and the first main menu page is displayed.
Pre-conditions:	No precondition in this iteration
Post Conditions:	System scans the files and starts generating a detailed report.
Normal Flow:	1. User enters the main menu page.

	2. The user clicks on the options they want to use to scan the system. 3. User clicks on the scan or start button. 4. The system verifies the user's request. Alternate 1: The user can't access the system and an error login page popups on the screen. 5. The system generates a report on the analysed data
Exceptional Flow:	The system catches an error and logs the user back into the scan option section
Includes:	There are no includes use-case for this application in v1.0.
Extends:	ScanOption ScanLocation
Special Conditions:	
Assumptions:	The user has data on there system hard drive

2.1.1.3. *Description & Priority*

Since Encrypto is labelled as a forensic tool, the user's main goal when utilising this application is to scan and generate a report of the data analysed. If the user can't scan the system, a report of any hidden data or insensitive information can't be achieved.

2.1.1.4. *Use Case*

The following diagram outlines the actors and use cases involved in the use access requirement.

Scope

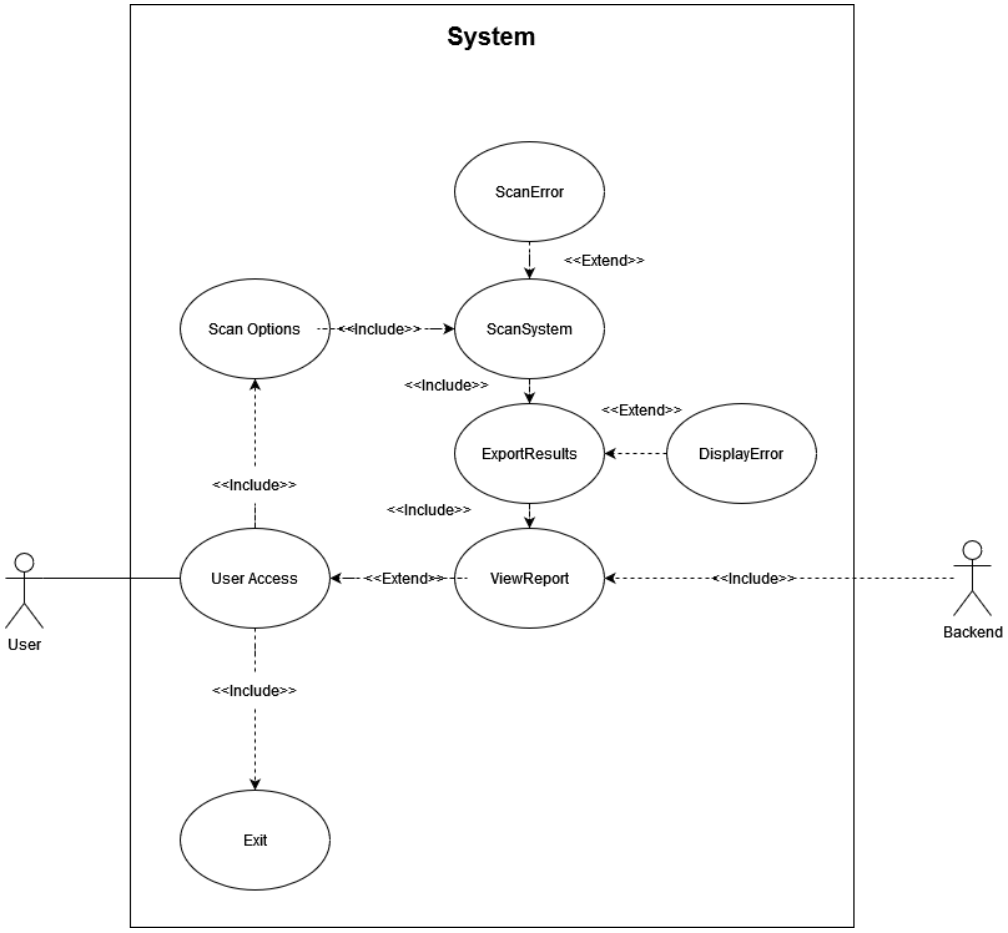
The scope of this use case is to control for non-administrative Encrypto users accessing the software system and the changes they can do relating to the system access.

Description

This use case describes the how the user scans the application, view report, changes setting, and exits the application.

Use Case Diagram

Diagram should highlight actors and uses cases involved in the User access requirement.



Flow Description

Precondition

The system is in initialisation mode.

Activation

This use case starts when the user scans the system.

Main flow

1. The system identifies the user's files and organises it into different categories.
2. If the user has no data on the system (See A1)
3. If the user has information but is denied access, (See E1)
4. The system generates a report.
5. The user can access the report.
6. The user exits out of the application.

Alternate flow

A1: No data

1. The system responds by telling the user there is no data in the computer.
2. The user needs to insert data to be scanned.
3. The use case continues at position 3 of the main flow.

Exceptional flow

E1: No access to the computer system

4. The system responds by asking the user for permission.
5. The user gives permission to the application to scan the system.
6. The use case continues at position 4 of the main flow.

Termination

The user clicks the exit button and cannot enter the system.

Post condition

The system goes into a wait state.

2.1.1.5. *Requirement 2: MD5 converter system*

Use Case Name:	MD5ConverterSystem
Description:	This use case allows the user to calculate and convert data to a hash file in the system
Actors: Primary and Secondary:	Primary actor is the user who utilises the application. Secondary actor is the application (System).
Triggers:	The system starts and the first main menu page is displayed.
Pre-conditions:	MD5 analysis radio button is selected in the Scan option page.
Post Conditions:	System scans the files and starts converting the files into a MD5 hash data.
Normal Flow:	1. User enters the main menu page. 2. The user clicks on the options they want to use to scan the system. 3. User clicks on the MD5 analysis button. 4. The system verifies the user's request. Alternate 1: The user can't access the system and an error login page popups on the screen. 5. The system calculates the MD5 hash data. Allows the user to input a message and generate its MD5 hash. 6. The system allows the user to input a message and convert it to its MD5 hash. 7. The system allows the user to input two MD5 hashes and compare them to check if they match. 8. The system allows the user to copy the MD5 hash to the clipboard. 9. The system allows the user to save the MD5 hash to a file. 10. The user can exit the system.

Exceptional Flow:	The system catches an error and logs the user back into the scan option section
Includes:	There are no includes use-case for this application in v1.0.
Extends:	ScanError ViewReport
Special Conditions:	
Assumptions:	The user has data on their system hard drive

2.1.1.6. *Description & Priority*

Since Encrypto is labelled as a forensic tool, the user's main goal when utilising this application is to scan, calculate and convert a file into a MD5 hash file. If the user can't access a file on the system, a MD5 can't be outputted.

2.1.1.7. *Use Case*

The following diagram outlines the actors and use cases involved in the MD5 converter requirement.

Scope

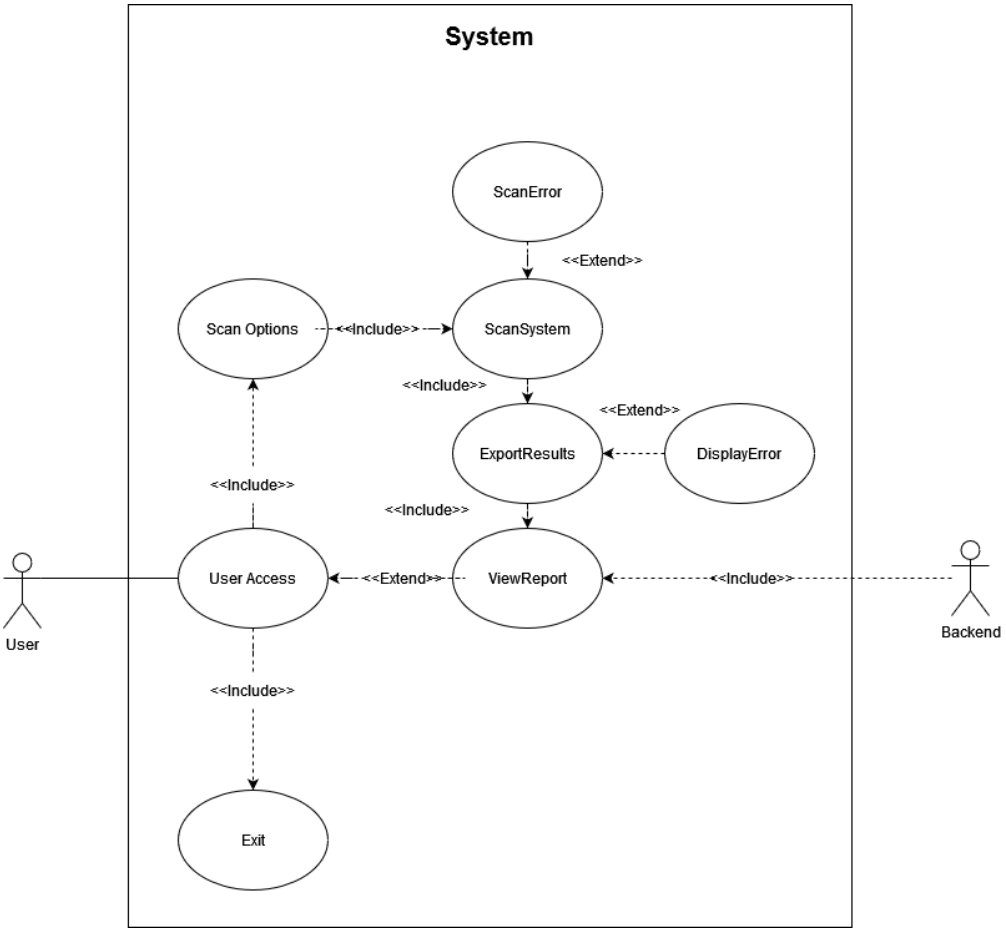
The scope of this use case is to control for non-administrative Encrypto users accessing the software system and the changes they can do relating to the system access.

Description

This use case allows the user to input a message and generate its MD5 hash.

Use Case Diagram

Diagram should highlight actors and uses cases involved in the MD5 converter requirement.



Flow Description

Precondition

The system starts and the first main menu page is displayed.

Activation

This use case starts when the user scans the system.

Main flow

1. The user selects the "Calculate MD5 hash" option.
2. The application presents the user with a text input field to enter the message.
3. The user enters the message in the input field and clicks on the "Calculate" button.
4. The application calculates the MD5 hash of the input message and displays it to the user.

Alternate flow

A1: No data

1. If the user inputs an empty message, the application displays an error message prompting the user to input a non-empty message.
2. If the user cancels the file save dialog, the application does not save the MD5 hash to a file.

Exceptional flow

E1: No access to the computer system

7. The system responds by asking the user for permission.
8. The user gives permission to the application to scan the system.
9. The use case continues at position 4 of the main flow.

Termination

The user clicks the exit button and cannot enter the system.

Post condition

The system goes into a wait state.

2.1.1.8. *Requirement 3: Registry editor system*

Use Case Name:	RegistryEditorSystem
Description:	This use case allows the user to manage there registry editor in the system.
Actors: Primary and Secondary:	Primary actor is the user who utilises the application. Secondary actor is the application (System).
Triggers:	The system starts and the first main menu page is displayed.
Pre-conditions:	Registry analysis radio button is selected in the Scan option page.
Post Conditions:	System opens up a registry tree view for the device.
Normal Flow:	1. The user launches the registry editor system. 2. The user navigates to the desired registry key they want to view. 3. The user selects the "View Registry Key" use case from the editor's menu or toolbar. 4. The system retrieves the contents of the selected registry key and displays them to the user in the editor's interface. 5. The user can then view the registry key's values and make any necessary changes or modifications. 6. The user can then either exit the registry editor system or proceed to perform other actions on the registry key, such as editing or deleting it.
Exceptional Flow:	1. The system attempts to retrieve the contents of the selected registry key but encounters an error due to insufficient permissions or a corrupted registry. 2. The system displays an error message to the user, informing them that the operation could not be completed.

	3. The user may choose to retry the operation, check their permissions, or attempt to repair the corrupted registry.
Includes:	There are no includes use-case for this application in v1.0.
Extends:	ScanError ViewReport
Special Conditions:	
Assumptions:	The user has data on their system hard drive

2.1.1.9. *Description & Priority*

Since Encrypto is labelled as a forensic tool, the user's main goal when utilising this application is to manage a systems registry. If the user can't access a file on the system, the registry won't be accessible.

2.1.1.10. *Use Case*

The following diagram outlines the actors and use cases involved in the registry editor requirement.

Scope

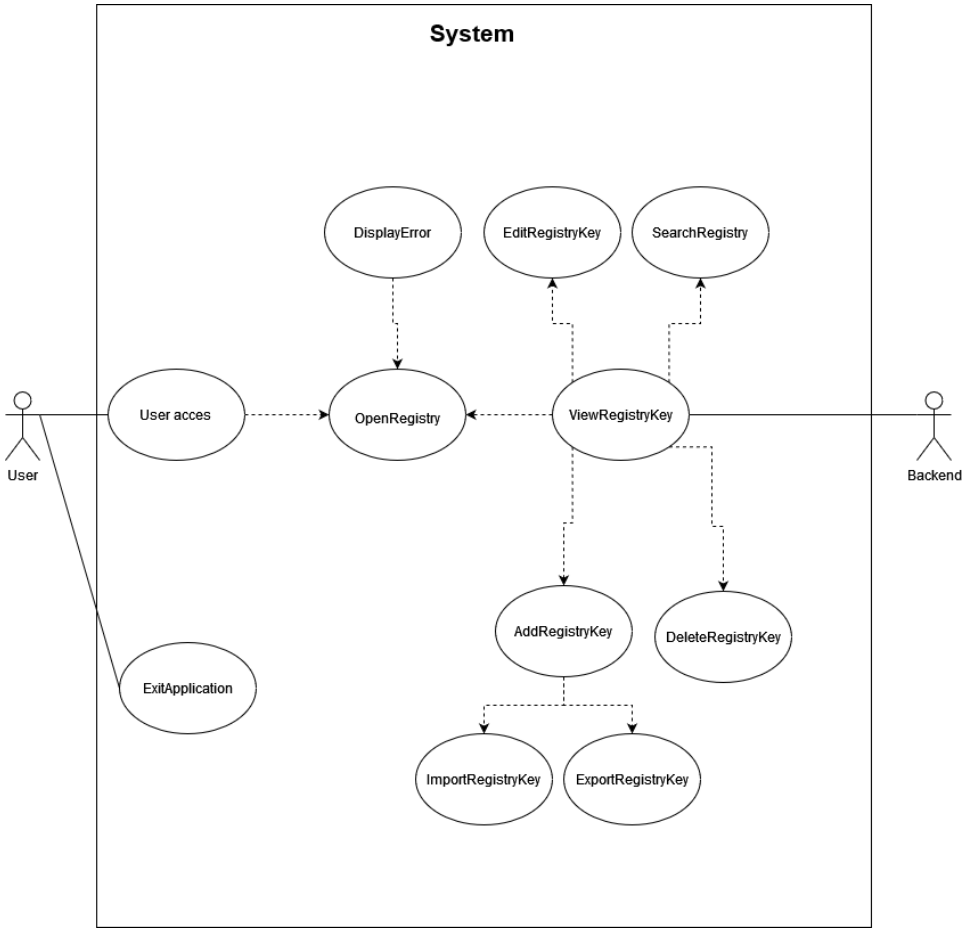
The scope of this use case is to control for non-administrative Encrypto users accessing the software system and the changes they can do relating to the system access.

Description

The user wants to view the contents of a specific registry key in the registry editor system.

Use Case Diagram

Diagram should highlight actors and uses cases involved in the registry editor requirement.



Flow Description

Precondition

The system starts and the first main menu page is displayed.

Activation

The user has launched the registry editor system.

Main flow

1. The user selects the "Registry analysis" option.
2. The user launches the registry editor system.
3. The user navigates to the desired registry key they want to view.
4. The user selects the "View Registry Key" use case from the editor's menu or toolbar.
5. The system retrieves the contents of the selected registry key and displays them to the user in the editor's interface.
6. The user views the registry key's values and makes any necessary changes or modifications.
7. The user can then either exit the registry editor system or proceed to perform other actions on the registry key, such as editing or deleting it.

Alternate flow

A1: No data

1. The system attempts to retrieve the contents of the selected registry key but encounters an error due to insufficient permissions or a corrupted registry.
2. The system displays an error message to the user, informing them that the operation could not be completed.

Exceptional flow

E1: No access to the computer system

1. The system attempts to retrieve the contents of the selected registry key but encounters an error due to insufficient permissions or a corrupted registry.
2. The system displays an error message to the user, informing them that the operation could not be completed.
3. The user may choose to retry the operation, check their permissions, or attempt to repair the corrupted registry.
4. The system responds by asking the user for permission.
5. The user gives permission to the application to edit the system registry.
6. The use case continues at position 4 of the main flow.

Termination

The user clicks the exit button and cannot enter the system.

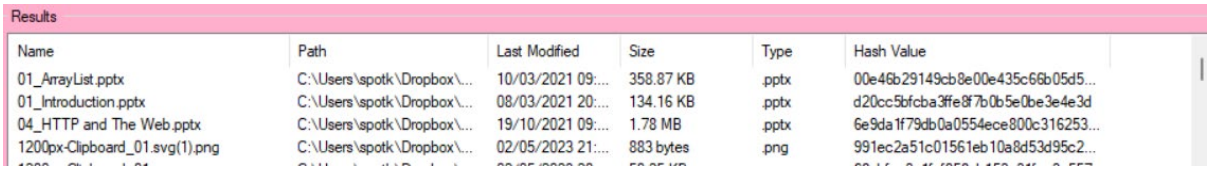
Post condition

The system goes into a wait state.

2.1.2. Data Requirements

Based on my project's goals and description, the following information is the data requirements for the Encrypto application:

1. ***File Data:*** The application will need to analyse, collect, and store file information from various sources in a device. This data may include file name, size, type of file, path location, modification, and hash information etc.


Results

Name	Path	Last Modified	Size	Type	Hash Value
01_ArrayList.pptx	C:\Users\spotk\Dropbox\...	10/03/2021 09:...	358.87 KB	.pptx	00e46b29149cb8e00e435c66b05d5...
01_Introduction.pptx	C:\Users\spotk\Dropbox\...	08/03/2021 20:...	134.16 KB	.pptx	d20cc5bfcba3ffe8f7b0b5e0be3e4e3d
04_HTTP and The Web.pptx	C:\Users\spotk\Dropbox\...	19/10/2021 09:...	1.78 MB	.pptx	6e9da1f79db0a0554ece800c316253...
1200px-Clipboard_01.svg(1).png	C:\Users\spotk\Dropbox\...	02/05/2023 21:...	883 bytes	.png	991ec2a51c01561eb10a8d53d95c2...

Figure 1: File Data

2. ***Registry data:*** My application will need to analyse, collect, and store registry data that is related to system components and device interactions. Some of the sensitive information that might be collected include, registry keys, values, and changes made by malware or malicious activity.

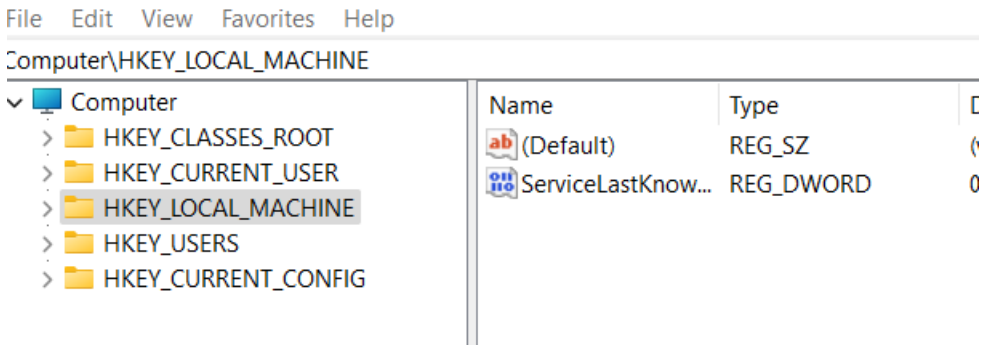


Figure 2: Registry Data

3. ***Network data:*** The application has to collect and store network data related to IP addresses, traffic, and access attempts made from the user's device. This data may include IP addresses, network traffic logs, and connectivity issues.

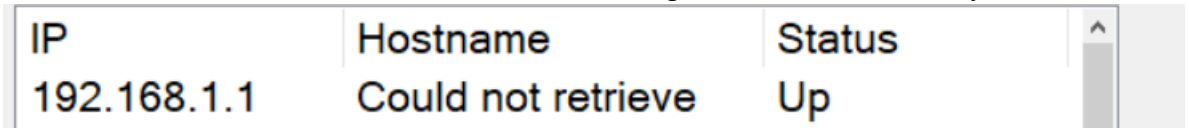


Figure 3: Network Data

4. ***Hash data:*** My application will need to collect analyse, collect, calculate and store MD5 data for files. MD5 (Message Digest Method 5) is a hash algorithm used for content verification and as a digital signature. It is an important feature in cryptography and is useful for investigators or users in checking a files integrity. My application will be utilising the MD5 hash values of individual files or multiple files.

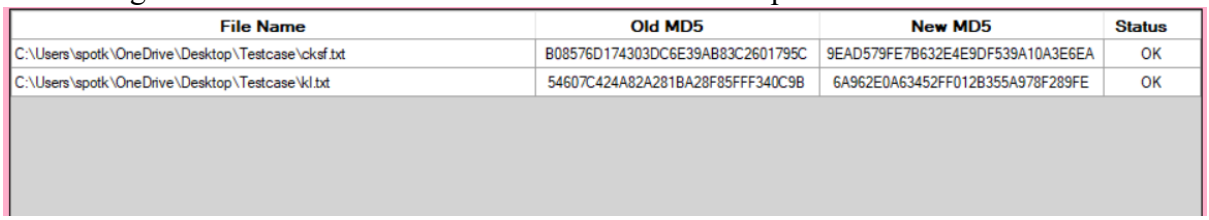


Figure 4: Hash data

5. ***System and user information:*** The application will be collecting and storing user information for reporting and documentation purposes. The information that will be exported as a report include file properties, metadata, usernames, IP addresses, registry results, and any other relevant data a device might hold.

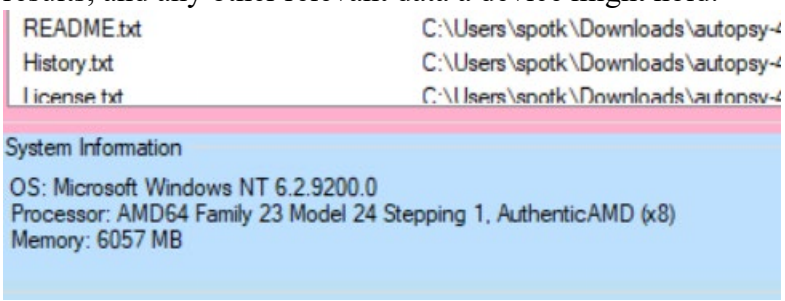


Figure 5:System Information

6. ***Encryption data:*** My application will be collecting and storing data into files that will either be encrypted or decrypted. The data involved in this process may include encryption keys, algorithms, and encryption schemes.

Establishing and following these data requirements is critical to minimizing data complexity over the applications development. The application may also need to adhere to data privacy and security regulations and ensure the proper handling and protection of sensitive data.

2.1.3. User Requirements

The user requirements for my project include:

1. **Ease of use**: The application should be intuitive and easy to use, with clear instructions and a user-friendly interface.
2. **Flexibility:** The application should be able to accommodate a variety of user needs and scenarios, such as different file types, network configurations, and encryption schemes.
3. **Reliability:** The application should be reliable and accurate, providing consistent and trustworthy results to users.
4. **Security:** The application should ensure the security and privacy of user data, protecting sensitive information from unauthorized access.
5. **Documentation and reporting:** The application should provide detailed documentation and reporting capabilities, allowing users to generate reports and documentation that can be used as evidence in court or for other purposes.
6. **Compatibility:** The application should be compatible with different operating systems and platforms, making it accessible to a wide range of users.
7. **Timeliness:** The application should provide results in a timely manner, allowing users to quickly identify and address potential threats or vulnerabilities.
8. **Support:** The application should have a support system in place to address any issues or questions that users may have, aiding and guidance as needed.

2.1.4. Environmental Requirements

The environmental requirements for Encrypto application include:

- **Operating System:** The application may require a specific version of the windows operating system, such as Windows 10, Windows 8.1, or Windows 7. The user should check the compatibility of the application with different versions of Windows. The best solution to avoid system errors is to have the latest version of Windows.
- **Processor:** The application may require a specific processor architecture, such as 32-bit or 64-bit. The user should ensure that the processor architecture of the target machine is compatible with the application.
- **RAM:** The application may require a specific amount of RAM to function efficiently. The user should ensure that the target machine has enough RAM to run the application.
- **Hard Disk Space:** The application may require a specific amount of free disk space to install and run. The user's device should have a minimum amount of 19MB disk space to accommodate the application running efficiently.
- **Dependencies:** The application may depend on other software components, such as .NET Framework, Visual C++ Redistributable, or other libraries. The user should ensure that these dependencies are installed on the target machine.
- **Security Permissions:** The application may require specific security permissions, such as access to certain files or folders. The user should ensure that they have administrative privileges when running the application.

2.1.5. Usability Requirements

The usability requirements for the application include:

- **Intuitive and easy to use:** The application has a user-friendly interface and is intuitive and easy to use, with clear instructions and straightforward navigation.

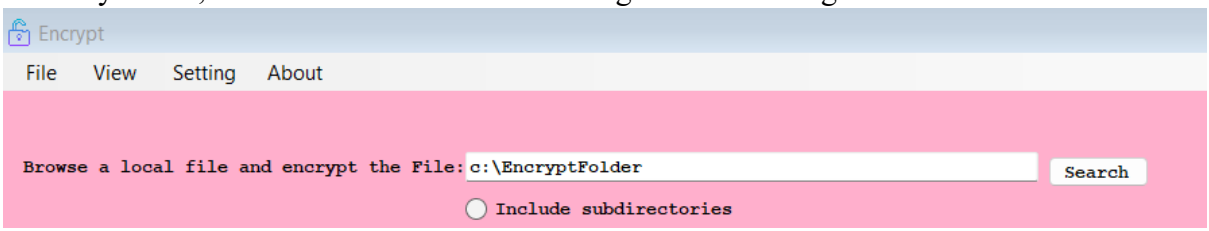


Figure 6: Easy to use UI and clear instructions.

- **Consistency:** The application is consistent in terms of the user interface, user experience, and functionality, providing users with a predictable and familiar experience.

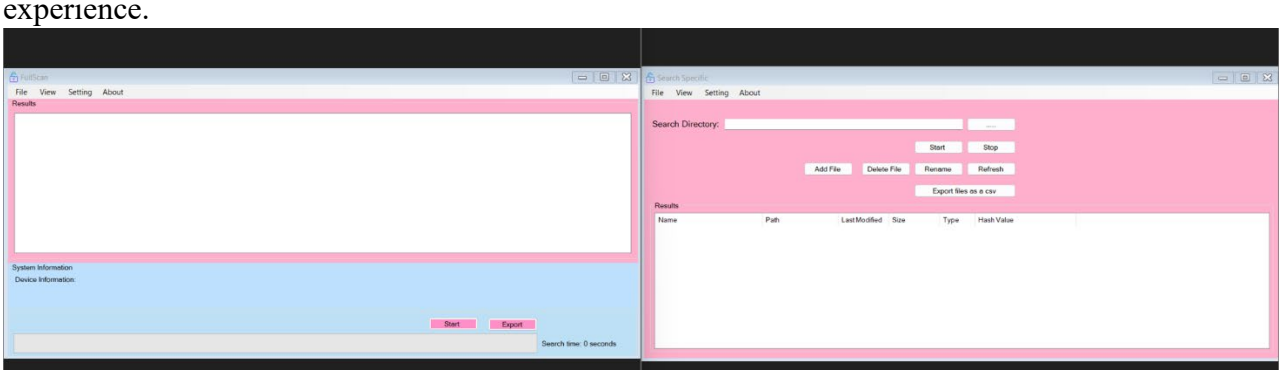


Figure 7:GUI consistency

- **Customizability:** The application allows the user to customize certain settings or options to suit their preferences or needs.

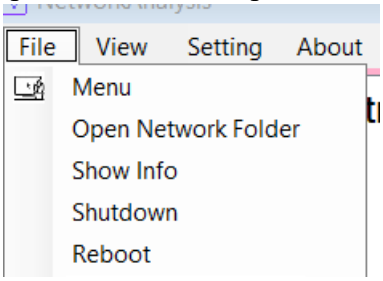


Figure 8: Menu highlights the customizability of the application.

- **Responsiveness:** The application is responsive and provides the user with real-time feedback, such as progress bars or notifications, to ensure a seamless and efficient experience.

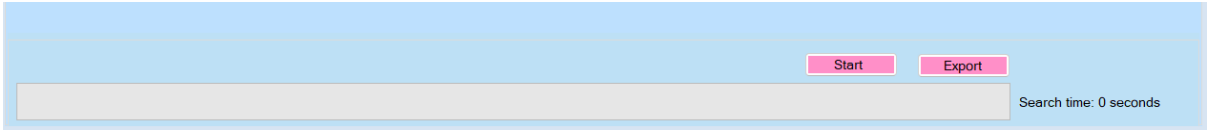


Figure 9: progress bar is an example of responsiveness.

- **Accessibility**: The application is accessible to users with disabilities, such as providing keyboard shortcuts or screen reader support.
- **Error handling:** The application should have robust error handling and provide users with clear and concise error messages or feedback, helping them identify and resolve issues quickly.

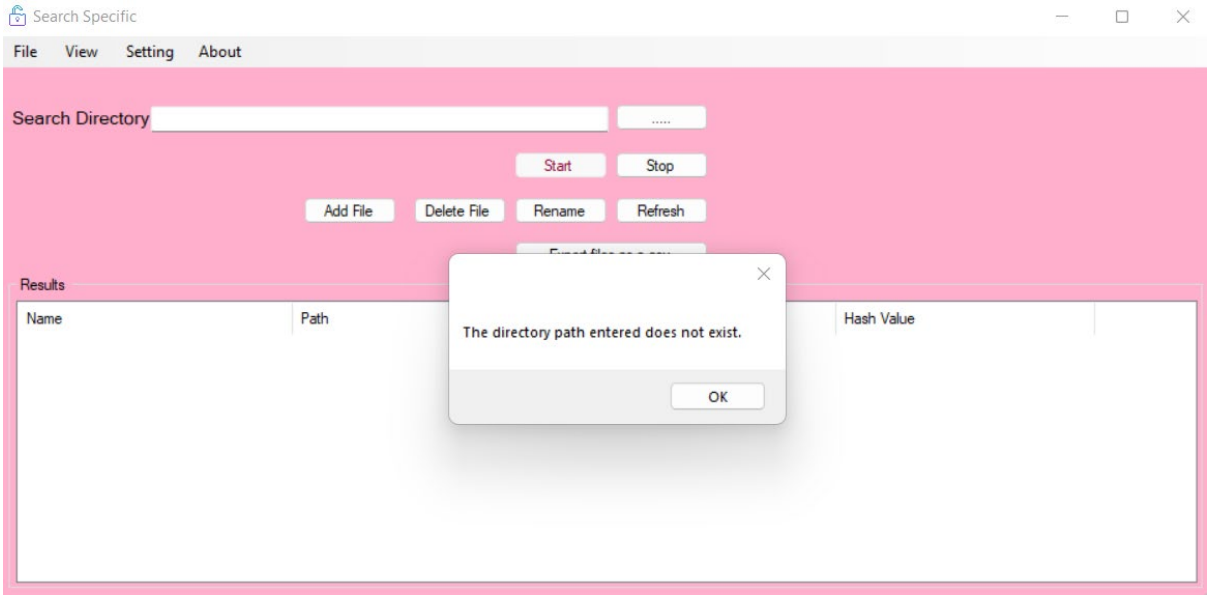


Figure 10:Dialogue box appear to show different error messages.

- **User feedback:** The application allows the user to provide feedback, such as ratings or reviews, to help improve the application and address any issues or concerns.

2.1.6. Non-Functional Requirements

The following non-functional requirements are expected of Encrypto:

- **Response time:** Encrypto will need to have a fast response time as it is a system that depends on the speed with which the data is being analysed for the end user. It is essential that the data is analysed within 10 minutes so that the report can be generated faster for the end user.
- **Availability:** It is essential that the data generated report need to be accessible to the user at all times and has a high availability from the end user's perspective. Encrypto must be able to analyse the information but also have it available to the users at any point that they might require it.
- **Reusability:** The reuse of information, features, algorithms, and other aspects will be required in order to minimise system development and to quickly advance and produce new iterations of the software.
- **Maintainability:** Encrypto will need to be easily maintainable from the perspective of the administrator of the system, allowing them to troubleshoot any problems/issues.
- **Reliability:** Encrypto should be reliable and operate consistently, without errors or crashes, and provide users with a high level of uptime and availability.
- **Security:** Encrypto should have robust security measures in place, such as encryption, authentication, and authorization, to protect user data and prevent unauthorized access.

2.2. Design & Architecture

The design of my application called Encrypto, is based on a modular or component-based approach, where each functionality or tool is implemented as a separate component. This design scheme can help with maintainability, scalability, and flexibility, allowing for easy modification and updates to individual components without affecting the overall system.

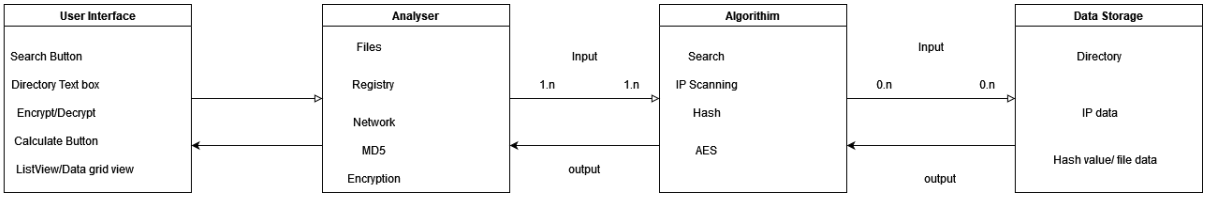


Figure 11:architectural Diagram

System architecture - The system architecture of Encrypto typically involves a layered architecture, with each layer responsible for a specific aspect of the application's functionality. Here is a high-level overview of the system architecture for my application:

- **Presentation layer -** This layer is responsible for presenting the user interface to the user and receiving input from the user. It communicates with the application layer to perform searches and display results.
- **Application layer -** This layer is responsible for processing search requests from the presentation layer and communicating with the domain layer to retrieve data results. It also handles any business logic or validation required for the analysis process.
- **Domain layer -** This layer is responsible for interacting with the file system to retrieve files and directories that match the analysis criteria. It may also implement search algorithms and filters to optimize the search process.
- **Data access layer -** This layer is responsible for interacting with any data storage used by the application, such as a database or file system. It provides an interface for the domain layer to retrieve and store data.

In summary, this type of architecture allows for separation of concerns and modularity, making it easier to maintain and update the application over time. It also allows for flexibility in how the application is deployed, with each layer able to be hosted on different hardware or software platforms.

Components: My application uses a number of components, that work hand in hand together to provide the user with the best data results and experience. The components used in my application include:

- **UI component -** A UI (User Interface) component is a self-contained element of a graphical user interface (GUI) that displays certain information. This component includes things like buttons, text boxes, menus, sliders, tabs, icons, and list views etc. UI components play an important role in creating a consistent and intuitive user experience, as they allow users to interact with an interface in a familiar and predictable way. As a developer it helped me build and maintain complex UIs, because I can simply reuse existing components rather than having to reinvent the wheel for every new feature.
- **File analysis component -** This is a tool that is designed to examine files and gather information about their content, structure, and behaviour. The primary purpose of a file analysis component is to help users understand the nature of the files they are working with, and to identify any potential issues or risks associated with those files.
- **Registry analysis component -** A tool that is used to analyse and interpret the data stored in the Windows Registry. This component can help users identify potential issues, errors, or security vulnerabilities in the registry by examining the values, keys, and data stored within it. It can also be used in identifying changes made to the registry

by malware or other unauthorized programs, which can help in detecting and removing malicious software.

- **Network analysis -** A tool used to analyse and monitor computer network traffic. It can help identify and troubleshoot network problems, optimize network performance, and detect security threats.
- **MD5 analysis component -** This is a tool that is designed to analyse and compute the MD5 hash value of a file or set of files. This tool can be used to detect file tampering, identify malicious files, and ensure the authenticity of downloaded files. It can also be used to compare the MD5 hash value of two files to determine whether they are identical or not.
- **Encryption/decryption component -** This component is designed to secure data by transforming it into an unreadable format using an encryption algorithm. This component is responsible for encoding the information in a way that can only be accessed by authorized users with the proper decryption key. This component is critical for ensuring that sensitive information remains confidential and protected from unauthorized access.

Algorithms: The main algorithms used in my application vary depending on the specific functionality being implemented. This is list of algorithms used in my application:

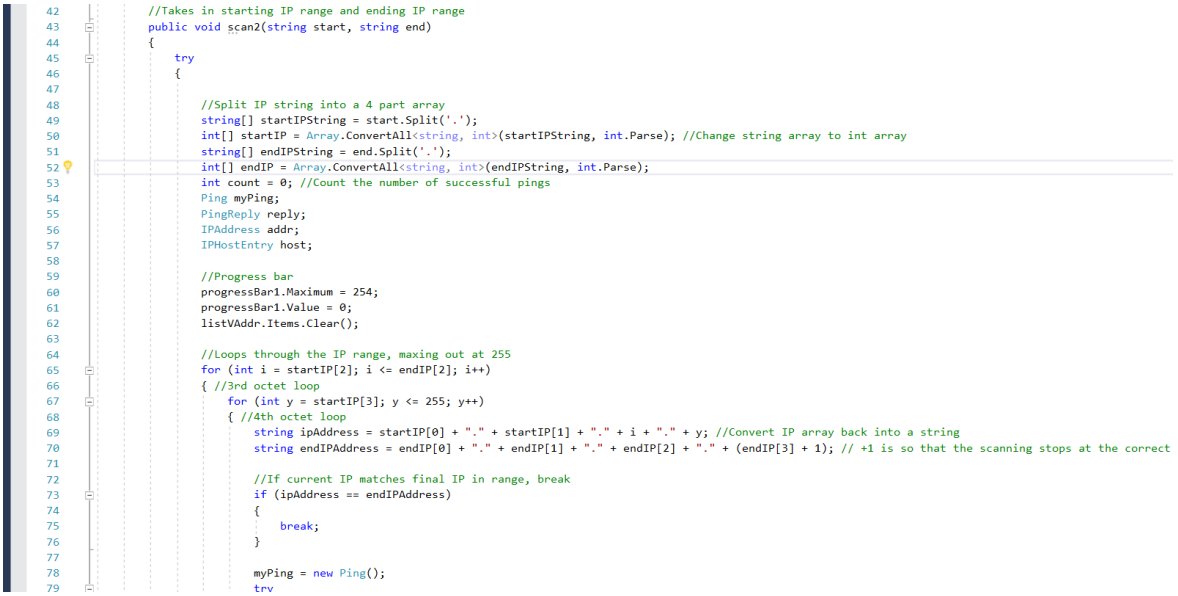
- **Depth-first search algorithm:** This algorithm was used for file analysis. An example of this algorithm being used, is in my "quick scan" file option. I used a recursive file search method to search and explore a directory and all of it subdirectories, and their subdirectories and so on. This method of searching a file is part of the depth-first search algorithm.

```
132
133         private IEnumerable<string> RecursiveFileSearch(string path, string pattern, ICollection<string> filePathCollector = null)
134         {
135             try
136             {
137                 filePathCollector = filePathCollector ?? new LinkedList<string>();
138
139                 var matchingFilePaths = Directory.GetFiles(path, pattern)
140                                                  .Where(file => file.EndsWith(".txt") || file.EndsWith(".doc"));
141
142                 foreach (var matchingFile in matchingFilePaths)
143                 {
144                     filePathCollector.Add(matchingFile);
145                 }
146
147                 var subDirectories = Directory.EnumerateDirectories(path);
148
149                 foreach (var subDirectory in subDirectories)
150                 {
151                     RecursiveFileSearch(subDirectory, pattern, filePathCollector);
152                 }
153
154                 return filePathCollector;
155             }
156             catch (Exception error)
157             {
158                 bool isIgnorableError = error is PathTooLongException || error is UnauthorizedAccessException;
159
160                 if (isIgnorableError)
161                 {
162                     return Enumerable.Empty<string>();
163                 }
164
165                 throw error;
166             }
```

Figure 12: Implementation of depth-first search in Encrypto

- **IP Scanning algorithm:** This algorithm is used in the Network analysis section. In my program the IP scanning algorithm iterates through a range of IP addresses and attempts to ping each one to determine if it is active or not. The algorithm also logs the IP address and additional system information about the device if the ping is successful. The

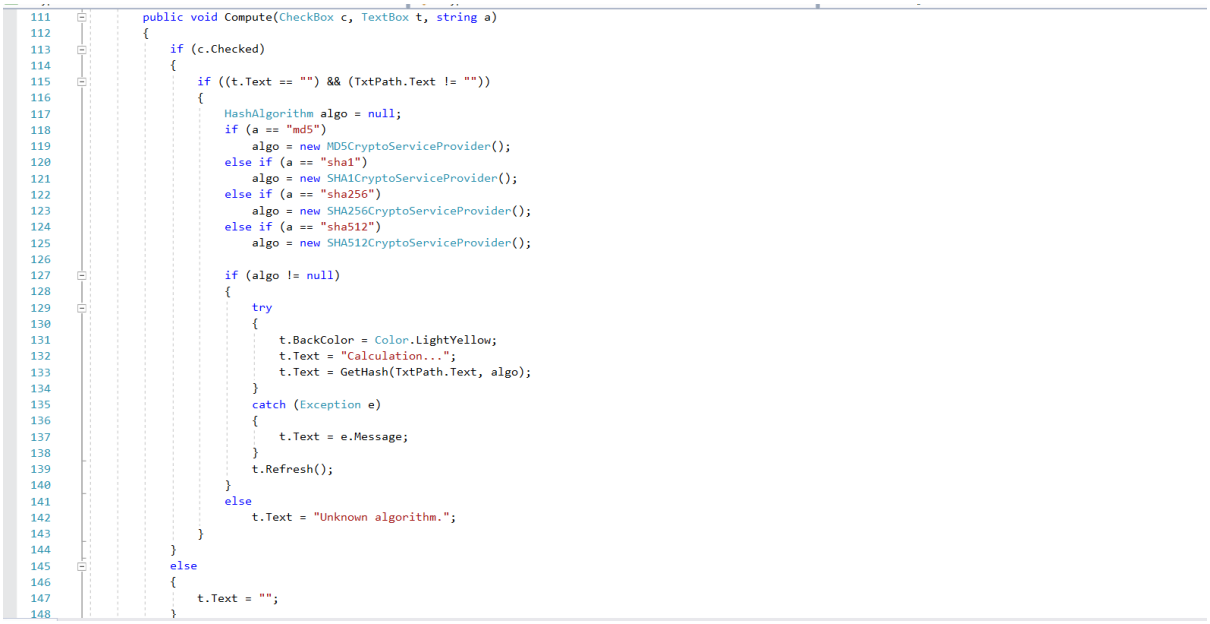
scanning process is multithreaded to improve performance, with one thread handling the scanning and another thread handling UI updates.



```
//Takes in starting IP range and ending IP range
public void scan2(string start, string end)
{
    try
    {

        //Split IP string into a 4 part array
        string[] startIPString = start.Split('.');
        int[] startIP = Array.ConvertAll<string, int>(startIPString, int.Parse); //Change string array to int array
        string[] endIPString = end.Split('.');
        int[] endIP = Array.ConvertAll<string, int>(endIPString, int.Parse);
        int count = 0; //Count the number of successful pings
        Ping myPing;
        PingReply reply;
        IPAddress addr;
        IPHostEntry host;

        //Progress bar
        progressBar1.Maximum = 254;
        progressBar1.Value = 0;
        listVAddr.Items.Clear();

        //Loops through the IP range, maxing out at 255
        for (int i = startIP[2]; i <= endIP[2]; i++)
        { //3rd octet loop
            for (int y = startIP[3]; y <= 255; y++)
            { //4th octet loop
                string ipAddress = startIP[0] + "." + startIP[1] + "." + i + "." + y; //Convert IP array back into a string
                string endIPAddress = endIP[0] + "." + endIP[1] + "." + endIP[2] + "." + (endIP[3] + 1); // +1 is so that the scanning stops at the correct

                //If current IP matches final IP in range, break
                if (ipAddress == endIPAddress)
                {
                    break;
                }

                myPing = new Ping();
                trv
```

Figure 13: IP Scanning algorithm

- **MD5 Hash algorithm:** In my hash calculator I'm using this algorithm to calculate hash values of files.



```
public void Compute(CheckBox c, TextBox t, string a)
{
    if (c.Checked)
    {
        if ((t.Text == "") && (TxtPath.Text != ""))
        {
            HashAlgorithm algo = null;
            if (a == "md5")
                algo = new MD5CryptoServiceProvider();
            else if (a == "sha1")
                algo = new SHA1CryptoServiceProvider();
            else if (a == "sha256")
                algo = new SHA256CryptoServiceProvider();
            else if (a == "sha512")
                algo = new SHA512CryptoServiceProvider();

            if (algo != null)
            {
                try
                {
                    t.BackColor = Color.LightYellow;
                    t.Text = "Calculation...";
                    t.Text = GetHash(TxtPath.Text, algo);
                }
                catch (Exception e)
                {
                    t.Text = e.Message;
                }
                t.Refresh();
            }
            else
                t.Text = "Unknown algorithm.";
        }
    }
    else
    {
        t.Text = "";
    }
```

Figure 14: An example of MD5 Hash algorithm

- **AES algorithm:** Advanced encryption standards (AES) algorithm is used in the encryption/decryption component of the application. AES is a symmetric encryption algorithm, meaning that the same key is used for both encryption and decryption. This makes it fast and efficient for encrypting large amounts of data. It also provides a high

level of security, with no known practical attacks against the algorithm when used with a sufficiently long key.

```
48          private void encryptButton_Click(object sender, EventArgs e)
49          {
50              if(string.IsNullOrWhiteSpace(textBox_EncrptFolder.Text) || string.IsNullOrWhiteSpace(outputFileTextBox.Text) || string.IsNullOrWhiteSpace(encryptionKeyT
51              {
52                  MessageBox.Show("Please enter all the required information.");
53                  return;
54              }
55
56              byte[] key = new byte[32]; // 256 bits
57              byte[] IV = new byte[16]; //128 bits
58
59              Aes aesAlg = Aes.Create();
60              aesAlg.KeySize = 256;
61
62              aesAlg.Key = key;
63              aesAlg.IV = IV;
64
65              FileStream inputFileStream = new FileStream(textBox_EncrptFolder.Text, FileMode.Open, FileAccess.Read);
66              FileStream outputFileStream = new FileStream(outputFileTextBox.Text, FileMode.Create, FileAccess.Write);
67
68              CryptoStream cryptoStream = new CryptoStream(outputFileStream, aesAlg.CreateEncryptor(), CryptoStreamMode.Write);
69
70              byte[] buffer = new byte[4096];
71              int bytesRead;
72              while ((bytesRead = inputFileStream.Read(buffer, 0, buffer.Length)) > 0)
73              {
74                  cryptoStream.Write(buffer, 0, bytesRead);
75              }
76
```

Figure 15: AES algorithm

2.3. Implementation

Specific File Search

```
1   using System;
2   using System.Collections.Generic;
3   using System.ComponentModel;
4   using System.Data;
5   using System.Drawing;
6   using System.Linq;
7   using System.Text;
8   using System.Threading.Tasks;
9   using System.Windows.Forms;
10  using System.IO;
11
12
13  namespace Encrypt.Ui
14  {
15      public partial class SearchSpecific : Form
16      {
17          private bool cancelSearch = false;
18
19          public SearchSpecific()
20          {
21              InitializeComponent();
22
23      }
24
25          private void menToolStripMenuItem_Click(object sender, EventArgs e)
26          {
27
28              //this will clear all the cookies created by the page.
29
30              this.Hide(); // Hide the current form.
31
32              Main f2 = new Main();
33
34              f2.Show();
35          }
36
37          private void SearchSpecific_Load(object sender, EventArgs e)
38          {
```

Figure 16: The start of the SearchSpecific component

The above code is used for searching files in a directory and exporting the results to a CSV file. The main algorithms, classes, and functions used in the code are:

- **FolderBrowserDialog:** This class is used to open a dialog box that allows the user to select a folder.

```
50          }
51
52          private void selectSearchFirButton_Click(object sender, EventArgs e)
53          {
54              FolderBrowserDialog dlg = new FolderBrowserDialog();
55              dlg.SelectedPath = searchDirTextBox.Text;
56              if (dlg.ShowDialog(this) == DialogResult.OK)
57              {
58                  searchDirTextBox.Text = dlg.SelectedPath;
59              }
60          }
61
```

Figure 17: folder browser dialog

The way this class works is by utilising the “searchButton”. If the “dlg” has been selected by the user, the “searchDirTextBox” will load up the directory path of the user’s device. Once the user has selected the path, they can press okay to load the path information into the text box of the interface.

- **File search algorithm:** This algorithm searches for a file in a specified directory and directory and displays some information about each file in a file in a list view.

```
72
73          //Get the parameters for the search thread:
74          if (Directory.Exists(searchDirTextBox.Text))
75          {
76              string[] files = Directory.GetFiles(searchDirTextBox.Text, "*.*", SearchOption.TopDirectoryOnly);
77
78              foreach(string file in files)
79              {
80                  if (cancelSearch)
81                  {
82                      break;
83                  }
84
85
86
87                  string fileName = Path.GetFileName(file); //file name
88                  string filePath = Path.GetDirectoryName(file); // Path of the file
89                  DateTime lastModified = File.GetLastWriteTime(file); //The last time the file was modified
90                  ulong fileSize = (ulong)new FileInfo(file).Length; //Get the file size
91                  string fileType = Path.GetExtension(file); //File type
92                  string hashValue = GetFileHash(file); // Get the file hash
93
94                  //Convert the size of the file to KB,MB and GB value
95                  string fileSizeINKBMBGB = GetFileSizeInKBMBGB(fileSize);
96
97                   ListViewItem item = new ListViewItem(new string[] { fileName, filePath, lastModified.ToString(), fileSizeINKBMBGB, fileType, hashValue });
98                  resultsList.Items.Add(item);
99              }
100         }
101         else
102         {
103             MessageBox.Show("The directory path entered does not exist. ");
104         }
```

Figure 18:File search algorithm

If the directory exists, it retrieves an array of file paths using the “Directory.GetFiles()” method, which returns all the files in the specified directory that match the specified search pattern ("." in this case).

For each file in the directory, the algorithm retrieves the following information: file name, file path, last modified time, file size, file type, and file hash. It stores this information in a “ListViewItem”, which is then added to the results list.

If the user clicks the stop button while the algorithm is searching for files, the “cancelSearch” variable is set to true, which allows the algorithm to break out of the loop that searches for files. After all files have been searched, the algorithm enables the search button and disables the stop button.

```
152
153     private void btnExport_Click(object sender, EventArgs e)
154     {
155         if (resultsList.Items.Count > 0)
156         {
157             SaveFileDialog saveFileDialog = new SaveFileDialog();
158             saveFileDialog.Filter = "CSV Files|*.csv";
159             saveFileDialog.Title = "Save search results to CSV file";
160             saveFileDialog.FileName = "search_results.csv";
161             if (saveFileDialog.ShowDialog() == DialogResult.OK)
162             {
163                 string filePath = saveFileDialog.FileName;
164                 // call a method to export the search results to the selected file
165                 ExportToCsv(filePath);
166             }
167         }
168         else
169         {
170             MessageBox.Show("No search results to export.");
171         }
172     }
173     private void ExportToCsv(string filePath)
174     {
175         // Open the selected file for writing
176         using (StreamWriter writer = new StreamWriter(filePath))
177         {
178             // Write the column headers to the file
179             writer.WriteLine("File Name,File Path,Last Modified,Size,Type,Hash Value");
180
181             // Write each search result to the file
182             foreach (ListViewItem item in resultsList.Items)
183             {
184                 writer.WriteLine(string.Join(",", item.SubItems.Cast<ListViewItem.ListViewSubItem>().Select(subitem => subitem.Text)));
185             }
186         }
187
188         MessageBox.Show("Search results exported to CSV file successfully.");
189     }
```

Figure 19: algorithm exports the results as csv files.

If the user clicks the export button, the algorithm opens a “SaveFileDialog” and allows the user to save the search results to a CSV file. If the user selects a file, the algorithm calls the “ExportToCsv ()” method, which writes the search results to the specified file.

Hash Converter

```
3   using System.ComponentModel;
4   using System.Data;
5   using System.Drawing;
6   using System.Linq;
7   using System.Text;
8   using System.Threading.Tasks;
9   using System.Windows.Forms;
10  using System.Diagnostics;
11  using System.IO;
12  using System.Security.Cryptography;
13  using System.Threading;
14
15  namespace Encrypt.Ui
16  {
17      public partial class HashChanger : Form
18      {
19          public int currentRowIndex = 0;
20          public bool running = false;
21
22          public HashChanger()
23          {
24              InitializeComponent();
25          }
26
27          private void HashChanger_Load(object sender, EventArgs e)
28          {
29
30          }
31
32          private void openFileToolStripMenuItem_Click(object sender, EventArgs e)
33          {
34              try
35              {
36                  Process.Start(dgvMD5.Rows[currentRowIndex].Cells[0].Value.ToString());
37              }
38              catch { }
39          }
```

Figure 20: The start of the hash converter

The above code is a C# function that is used for changing the MD5 hash of files. The code uses two cryptographic algorithms such as MD5 and RNGCryptoServiceProvider. The application consists of a single form named "HashChanger" with several controls such as DataGridView, Label, ProgressBar, Button, and MenuStrip. Here are the main algorithms/classes/functions used in the code:

- **OpenFileDialog** - The “OpenFileDialog” is a built-in class in C# that allows users to browse and select files from their computer. The class is used to display the open file dialog when the user clicks the "Add Files" button.

```
41          private void btnAddFiles_Click(object sender, EventArgs e)
42          {
43              OpenFileDialog selectFile = new OpenFileDialog();
44              selectFile.Multiselect = true;
45              bool flag = selectFile.ShowDialog() == DialogResult.OK;
46              if (flag)
47              {
48                  int totalFiles = selectFile.FileNames.Length;
49                  labelItem.Text = "0";
50                  labelTotalItem.Text = totalFiles.ToString();
51                  progressBarStatus.Value = 0;
52                  progressBarStatus.Maximum = totalFiles;
53                  Thread t = new Thread(() => checkMD5(selectFile.FileNames));
54                  t.IsBackground = true;
55                  t.Start();
56              }
57          }
```

Figure 21: Add files to get converted.

The OpenFileDialog is configured to allow the selection of multiple files by setting the Multiselect property to true.

- **MD5 -** The MD5 is a built-in class in C# that is used to compute the MD5 hash value of a file. The class uses two functions - checkMD5 and changeMD5. In checkMD5, the MD5 class is used to compute the MD5 hash value of each selected file and compare it with the stored hash value in DataGridView.

```
96          private void changeMD5(string[] fileNames)
97          {
98              Random random = new Random();
99              Thread.Sleep(1000);
100             this.Invoke((MethodInvoker)delegate ()
101             {
102                 this.btnStartMD5.Enabled = true;
103             });
104
105             for (int i = 0; i < fileNames.Length; i++)
106             {
107                 if (!running)
108                 {
109                     this.Invoke((MethodInvoker)delegate ()
110                     {
111                         this.btnStartMD5.Text = "Start Change MD5";
112                         running = false;
113                     });
114                     break;
115                 }
116                 int num = random.Next(2, 7);
117                 byte[] extraByte = new byte[num];
118                 for (int j = 0; j < num; j++)
119                 {
120                     extraByte[j] = (byte)0;
121                 }
122                 long fileSize = new FileInfo(fileNames[i]).Length;
123                 if (fileSize == 0L)
124                 {
125                     this.Invoke((MethodInvoker)delegate ()
126                     {
127                         this.dgvMD5.Rows[i].Cells[3].Value = "Empty";
128                     });
129                 }
130                 else
131                 {
132                     using (FileStream fileStream = new FileStream(fileNames[i], FileMode.Append))
133                     {
```

Figure 22: Change MD5 class

In changeMD5, the MD5 class is used to compute the new MD5 hash value of the file after modifying its contents.

- **Thread -** The Thread is a built-in class in C# that provides functionality for multi-threading. In the application, two threads are used. One in checkMD5 and another in changeMD5 in order to perform time-consuming operations in the background without blocking the main UI thread.

```
41      private void btnAddFiles_Click(object sender, EventArgs e)
42      {
43          OpenFileDialog selectFile = new OpenFileDialog();
44          selectFile.Multiselect = true;
45          bool flag = selectFile.ShowDialog() == DialogResult.OK;
46          if (flag)
47          {
48              int totalFiles = selectFile.FileNames.Length;
49              labelItem.Text = "0";
50              labelTotalItem.Text = totalFiles.ToString();
51              progressBarStatus.Value = 0;
52              progressBarStatus.Maximum = totalFiles;
53              Thread t = new Thread(() => checkMD5(selectFile.FileNames));
54              t.IsBackground = true;
55              t.Start();
56          }
57      }
```

Figure 23: Thread in the check MD5

The IsBackground property is set to true to indicate that the thread should terminate when the application exits.

- **DataGridView -** DataGridView is a built-in control in C# that displays data in a tabular format. The control is used to display the list of selected files with their corresponding path, original MD5 hash value, new MD5 hash value, and status. The user can select one or more rows in the DataGridView and remove them by clicking the "Remove Selected" button.

```
59      private void btnRemoveSelected_Click(object sender, EventArgs e)
60      {
61          foreach (DataGridViewRow row in dgvMD5.SelectedRows)
62          {
63              dgvMD5.Rows.RemoveAt(row.Index);
64          }
65          dgvMD5.ClearSelection();
66          labelItem.Text = "0";
67          labelTotalItem.Text = dgvMD5.RowCount.ToString();
68      }
69
```

Figure 24: Remove in the DataGridView

DataGridView is also used to update the status and new MD5 hash value of each file after modification.

- **FileStream -** The FileStream is a built-in class in C# that provides functionality for reading and writing to a file. The class is used in the changeMD5 function to append extra bytes to the end of the file before computing its new MD5 hash value. The FileMode is set to Append to indicate that the bytes should be written to the end of the file, and the extraByte array is filled with zeros using a loop.

```
132                 using (FileStream fileStream = new FileStream(fileNames[i], FileMode.Append))
133                 {
134                     fileStream.Write(extraByte, 0, extraByte.Length);
135                 }
136                 int bufferSize = fileSize > 1048576L ? 1048576 : 4096;
137                 string md5hash = "";
138                 using (MD5 md = MD5.Create())
139                 {
140                     using (FileStream fileStream2 = new FileStream(fileNames[i], FileMode.Open, FileAccess.Read, FileShare.Read, bufferSize))
141                     {
142                         md5hash = BitConverter.ToString(md.ComputeHash(fileStream2)).Replace("-", "");
143                     }
144                 }
145                 this.Invoke((MethodInvoker)delegate ()
146                 {
147                     bool flag2 = this.dgvMD5.Rows[i].Cells[2].Value.ToString() != "";
148                     if (flag2)
149                     {
150                         this.dgvMD5.Rows[i].Cells[1].Value = this.dgvMD5.Rows[i].Cells[2].Value;
151                     }
152                     this.labelItem.Text = (i + 1).ToString();
153                     this.progressBarStatus.Value = i + 1;
154                     this.dgvMD5.Rows[i].Cells[2].Value = md5hash;
155                     this.dgvMD5.Rows[i].Cells[3].Value = "OK";
156                 });
157             }
158         }
159         this.Invoke((MethodInvoker)delegate ()
160         {
161             this.btnStartMD5.Text = "Start Change MD5";
162             running = false;
163         });
```

Figure 25:Using Filestream in the changeMD5 function.

From the changeMD5 function, a new FileStream is created for the current file with FileMode set to Append. The extraByte array is then written to the end of the file using the Write method of the FileStream. This ensures that the file is modified before computing its new MD5 hash value.

2.4. Graphical User Interface (GUI)

The following are screenshots of the GUI of the project:

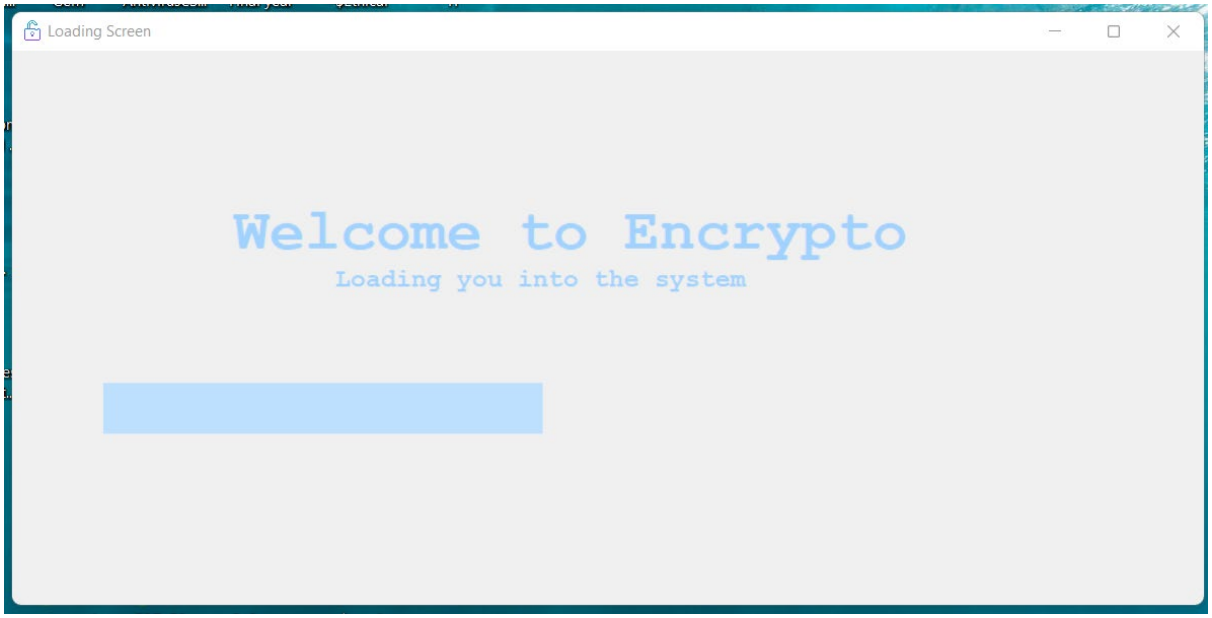


Figure 26: Loading screen of the application

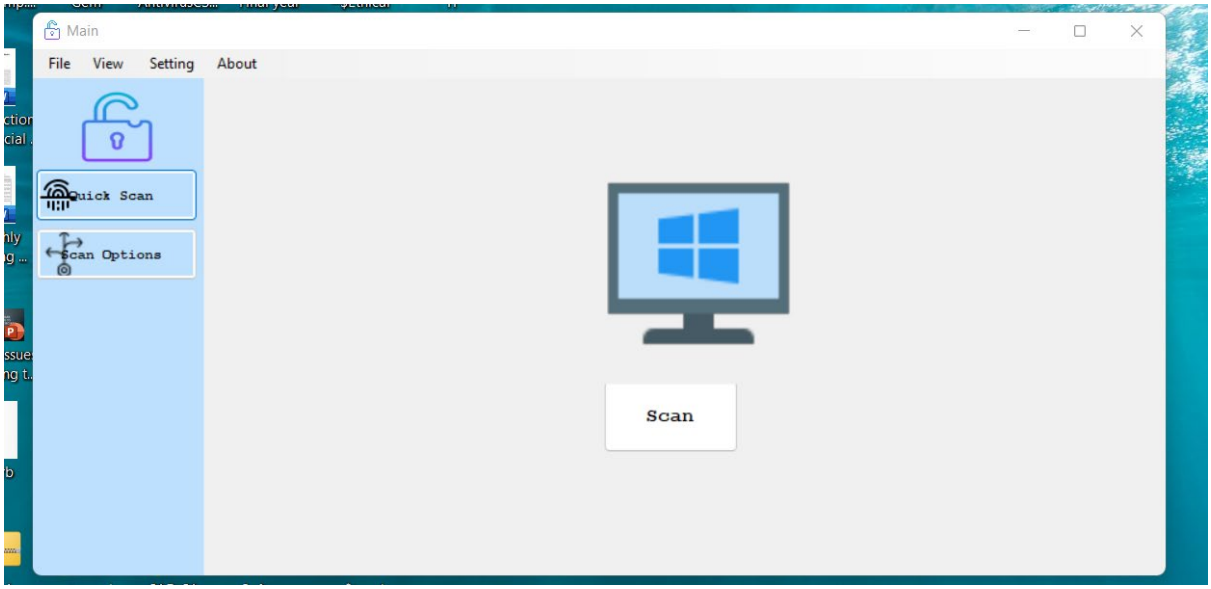


Figure 27: The Main form of the application

The user is given two options (Quick Scan and Scan Option). If the user clicks Quick Scan button it will take them to the GUI page below.

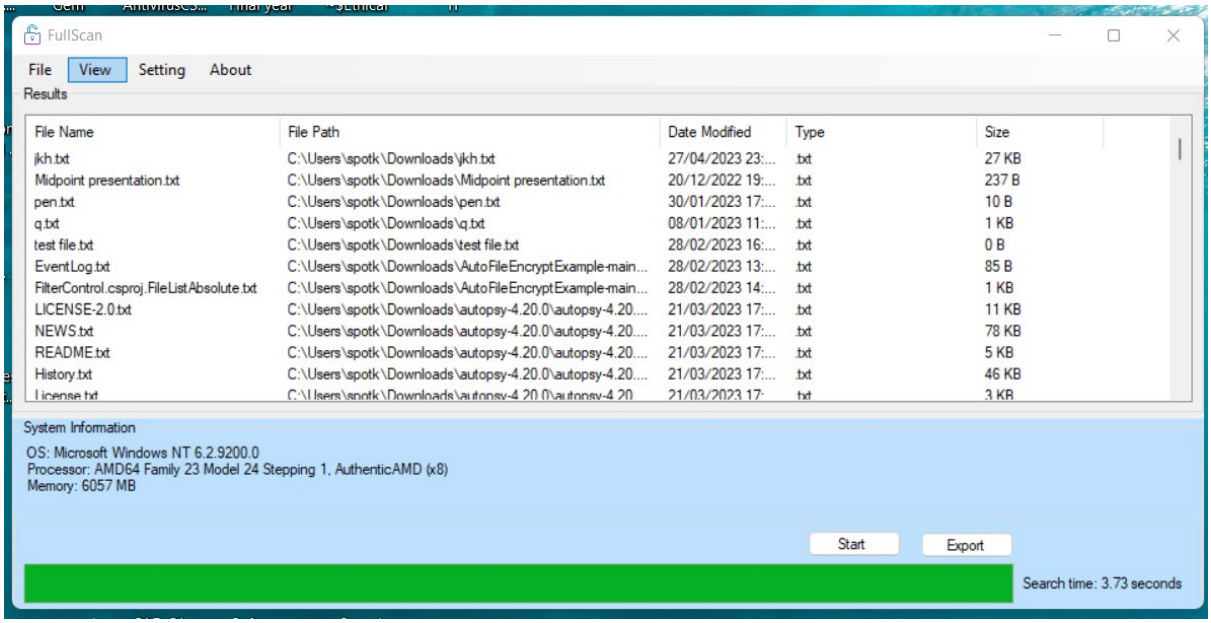


Figure 28: Quick Scan of the user's device

If the user clicks the second option (Scan Option). If the user clicks the Option button, it will take them to the GUI page below.

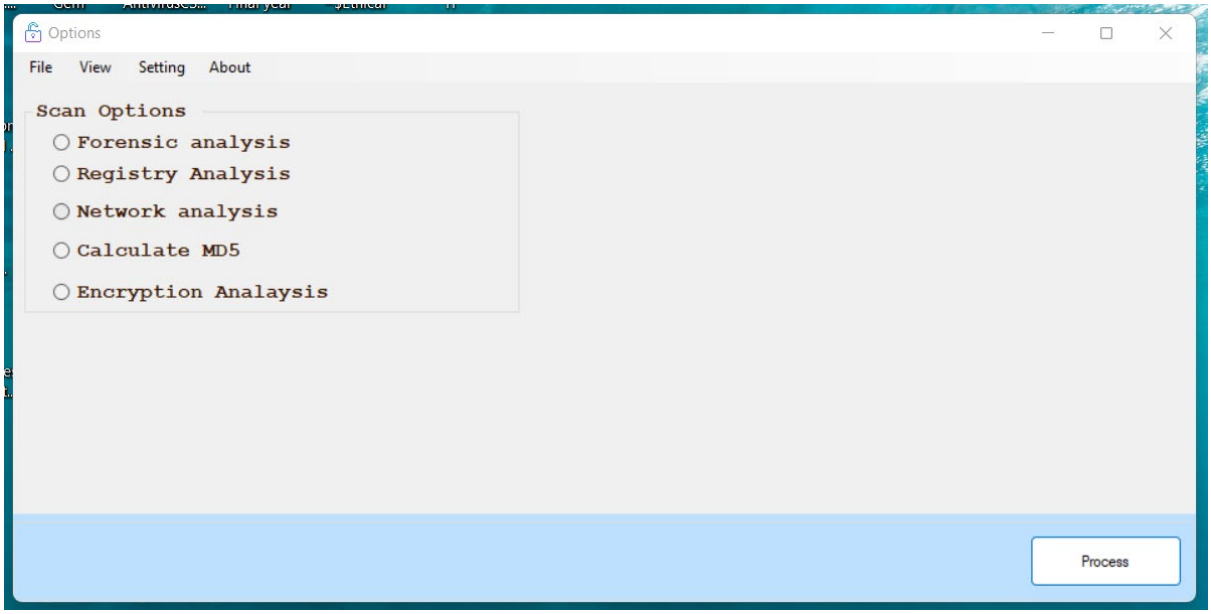


Figure 29: The option form

If the user picks a radio button, they will either be taken to another form or given a second option of radio buttons.

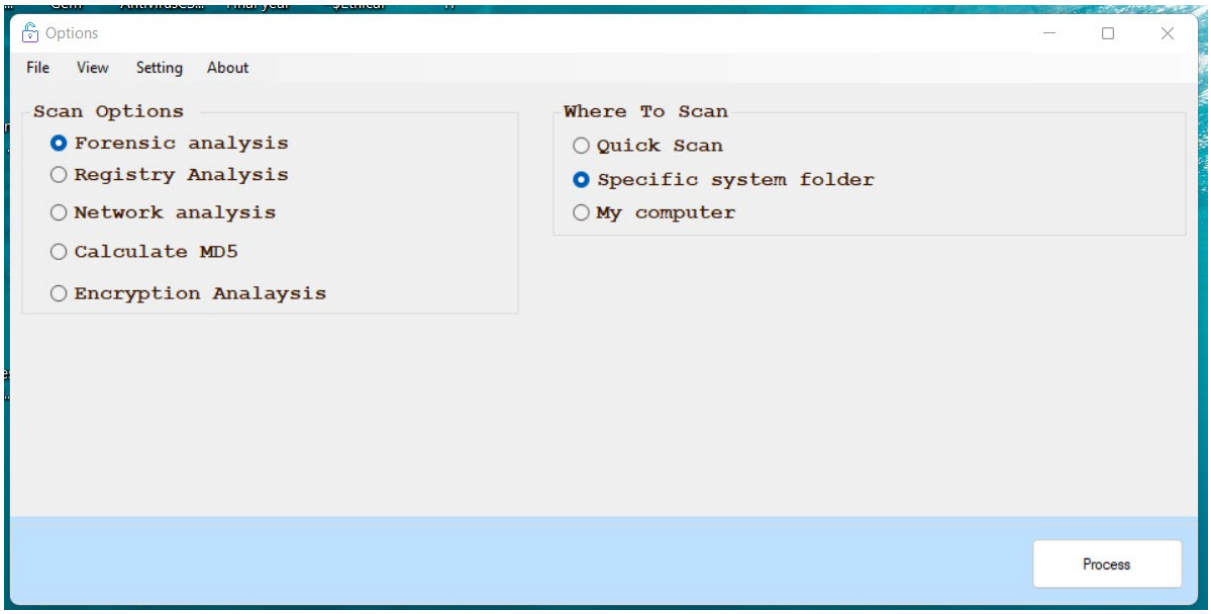


Figure 30: option of where to scan.

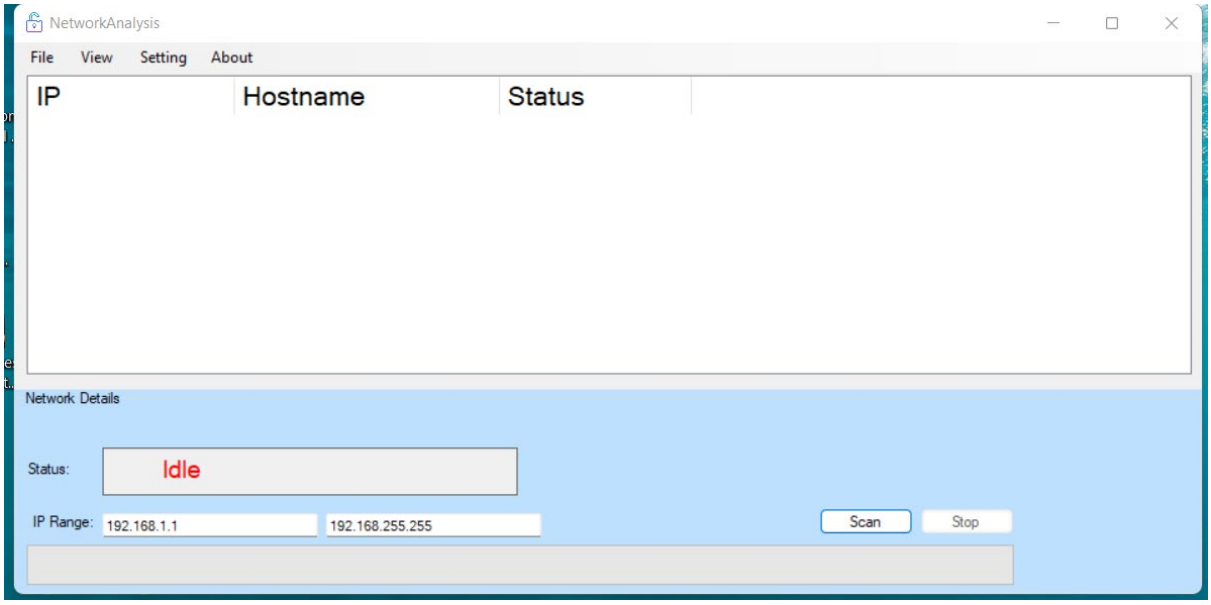


Figure 31: Network Analysis

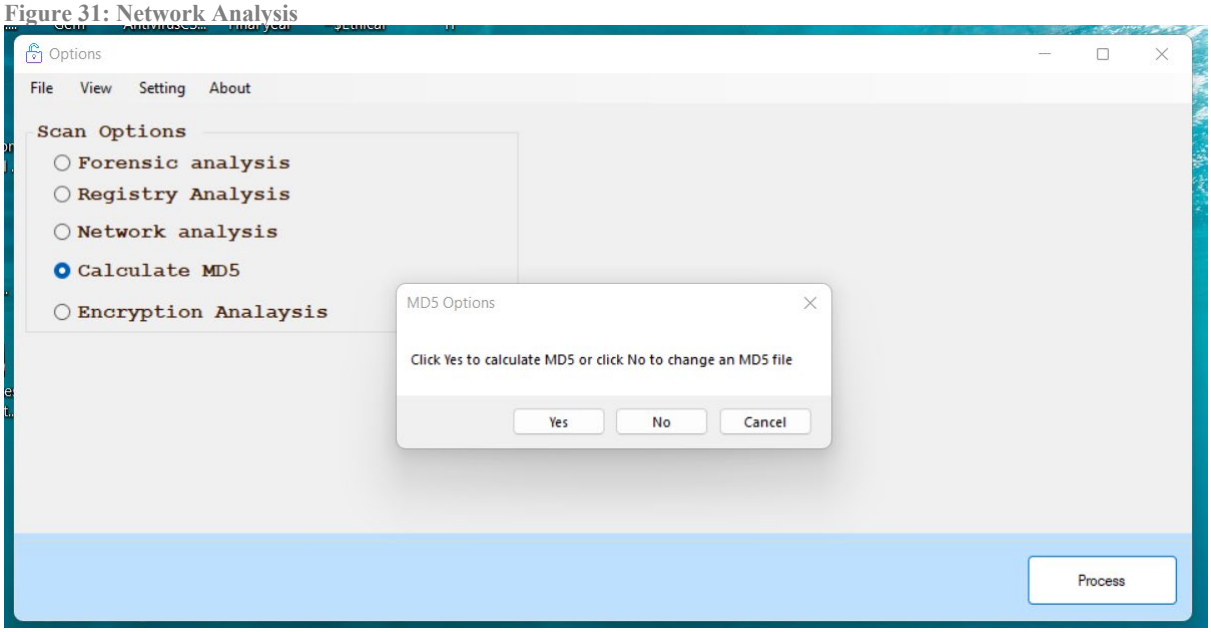


Figure 32: Calculate MD5 Options

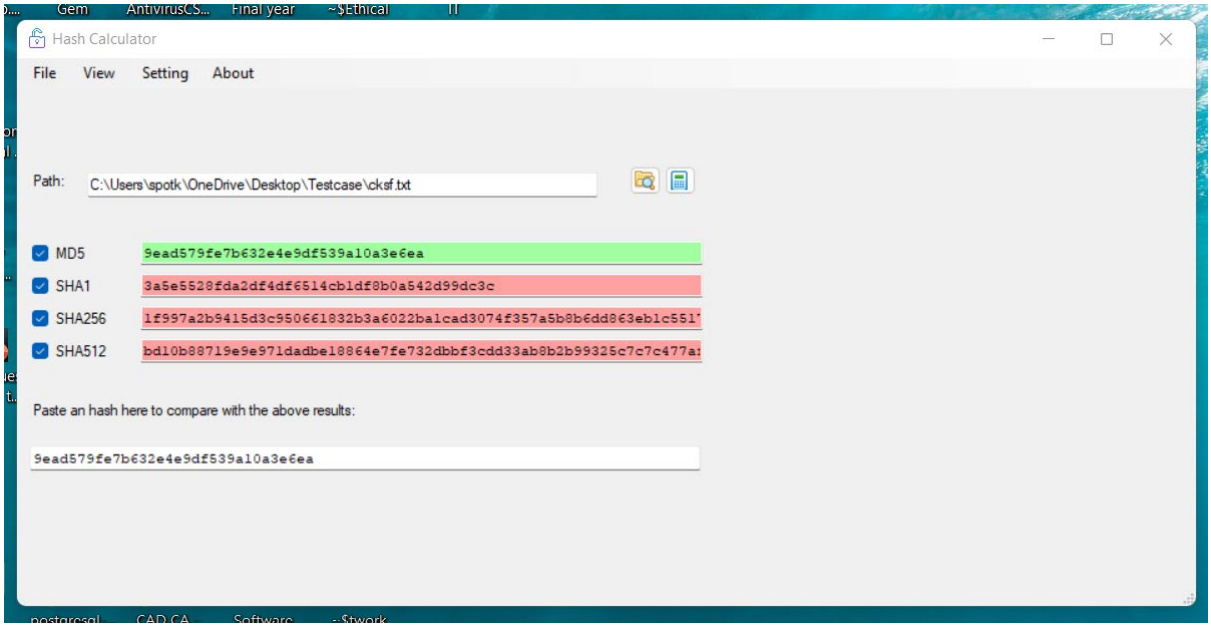


Figure 33: Hash Calculator

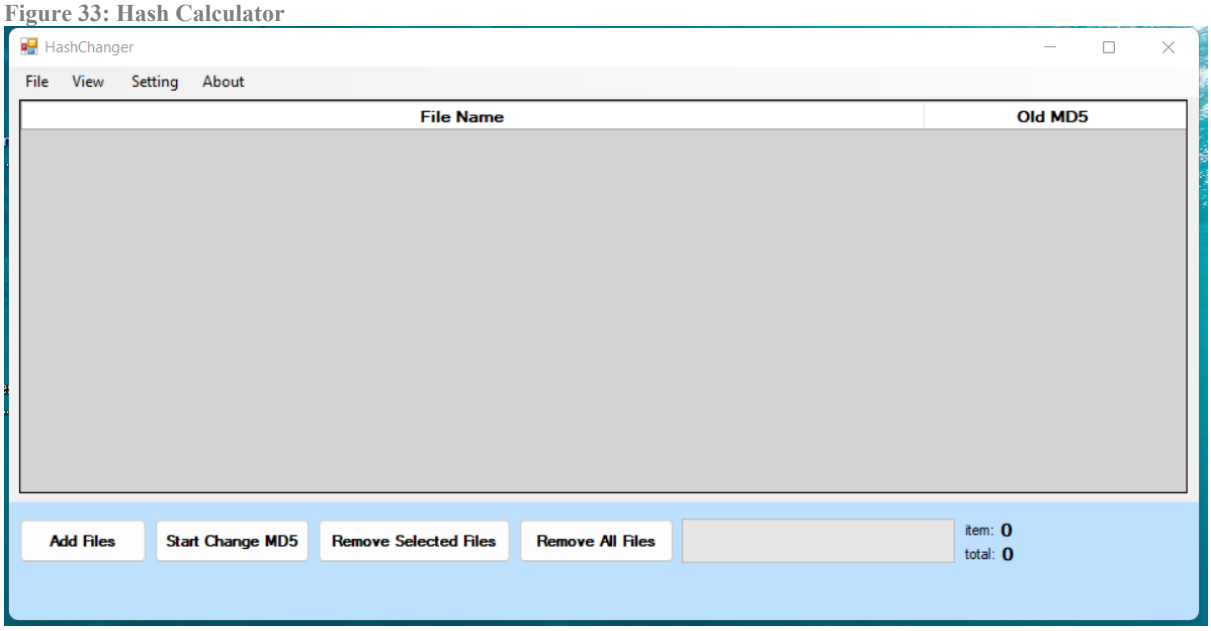


Figure 34: Hash Changer

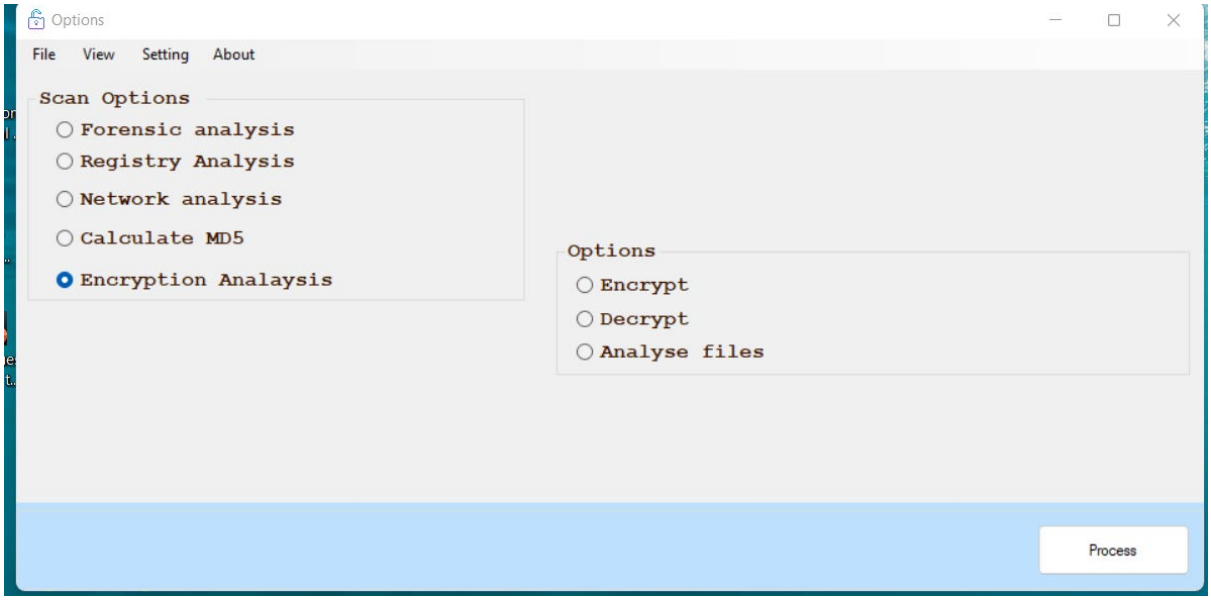


Figure 35: Encryption Analysis

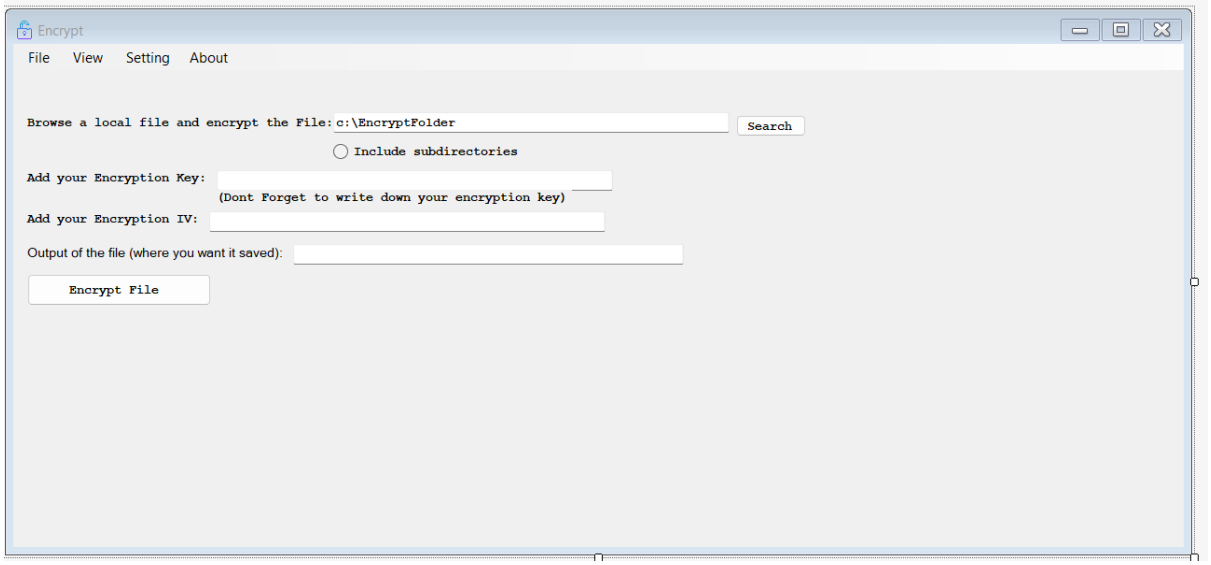


Figure 36: Encryption

2.5. Testing

I used visual studios built in testing framework MSTest for my testing plan. This testing framework is easy to use and supports various types of tests such as unit, integration, and acceptance tests.

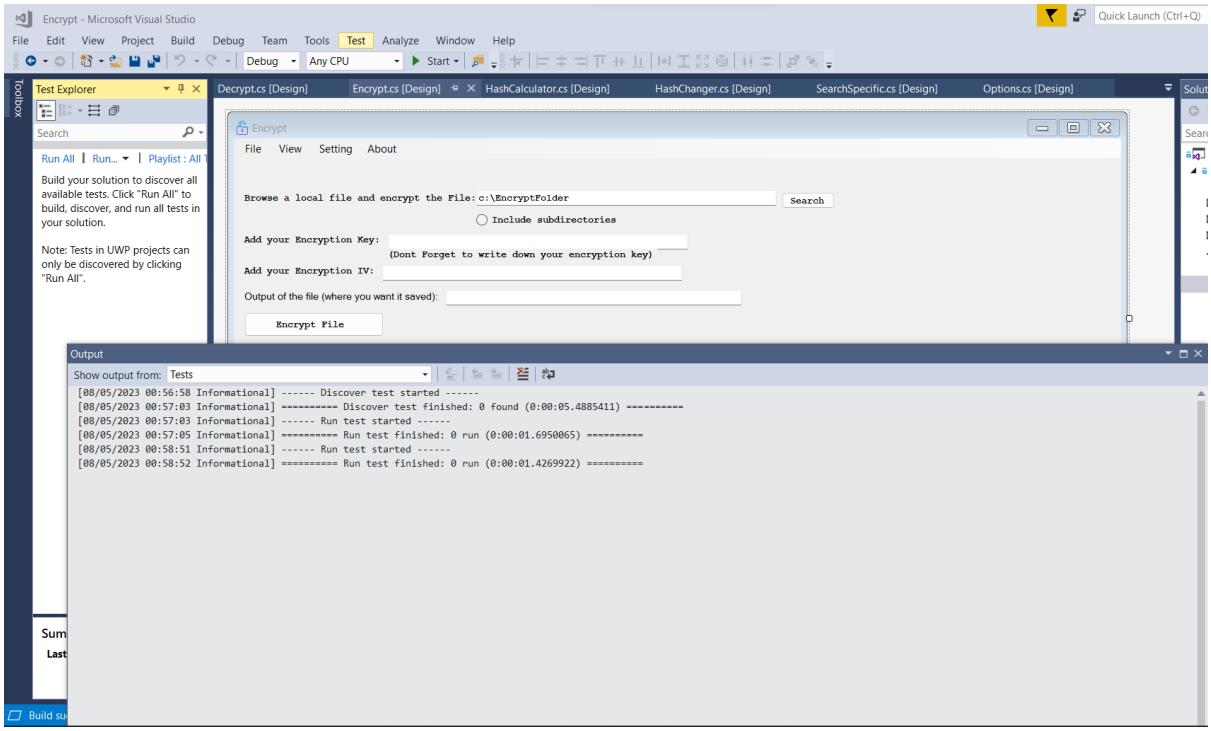


Figure 37: Visual Studios Built in testing framework.

In Scope: All the components in Encrypto which were defined in the requirement specs were tested.

Out of Scope: The features that weren't included in the testing procedures include:

- Hardware interface
- Media files
- Web functionalities
- Communication interface
- Application Security

The test objectives are to **verify** the functionality of Encrypto, the project should focus on testing the **analysis operations** such as File analysis, registry analysis, network analysis and MD5 etc. This testing needs to be done in order to **guarantee** all these operations can work **normally** in a real-world environment.

Test Case

Test Case Type	Description	Test Step	Expected Results	Status
Functionality	Can access the file directory.	Search for all path directories.	All paths in a directory can be accessed.	Fail
Security	Hash files can be created	Create a new file in file manager or in the application.	A new file can show a hash or MD5 details.	Pass
Usability	Ensure all buttons can be accessed.	Have users click on various page forms.	All the buttons can be accessed and can allow functions to work.	Pass

Error Handling	The applications system returns all errors to the user.	Insert wrong or null data into a textbox.	The users get an error message box, and the system doesn't crash.	Pass

This type of testing done was:

Unit testing - Unit testing is a type of testing that focuses on individual components (such as classes or methods) of a software system. Unit testing frameworks in visual studios can be used to automate unit testing and ensure that each component works as expected. A positive outcome from the test is by getting a successful build from a specific component.

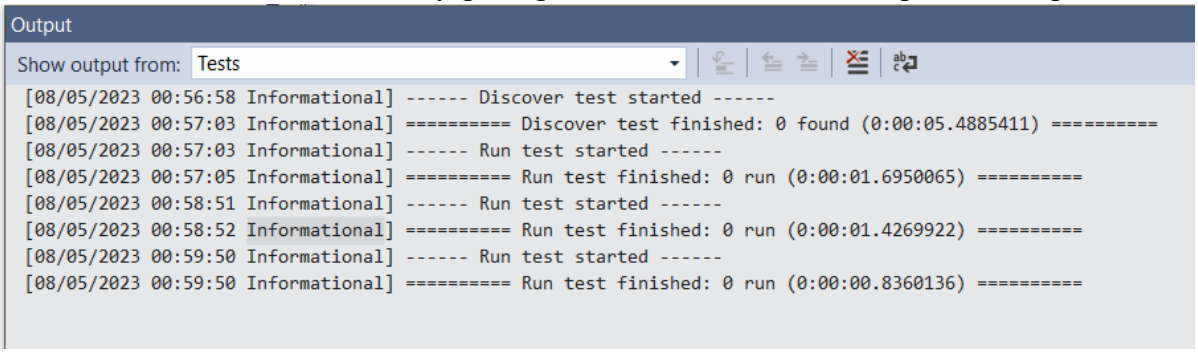


Figure 38: Example of a successful test on file analysis

Integration testing – I used integration testing to focus on testing how different components of a software system work together. Integration testing frameworks (Selenium) can be used to automate integration testing and ensure that all components work together correctly.

End user testing – I used this testing method to focus on testing the software system from the perspective of the end user. End user testing can be carried out manually. The results of end user testing can help to identify any issues or usability problems that need to be addressed. I used a voluntarily third party to test out my application and give a unbiased feedback on the application

2.6. Evaluation

In this section of the document, I will evaluate the data usage and performance of my application. Using the data from the testing phase, I was able to collect and calculate the following quantitative results and usage data. The data was put into the table below.

Evaluation Focus: Usage

Metrics Measured	**Test Plan**	**Pass Percentage**	**Results**
Number of active users per day	Usage data is collected by logging user actions within the Encrypto application.	85%	Average number of active users per day: 5
Number of files analysed per day	Usage data is collected by logging user actions within the Encrypto application.	95%	Average number of files analysed per day: 100
Number of registry analyses performed per day	Usage data is collected by logging user actions within the Encrypto application.	70%	Average number of registry analyses performed per day: 50
Number of network analyses performed per day	Usage data is collected by logging user actions within the Encrypto application.	55%	Average number of network analyses performed per day: 25
Number of encryption/decryption processes performed per day	Usage data is collected by logging user actions within the Encrypto application.	45%	Average number of encryption/decryption processes performed per day: 10

In summary, based on the usage data collected, the Encrypto application appears to be seeing moderate use, with an average of 5 active users per day. The most commonly used features are file analysis and registry analysis, which are performed approximately 100 and 50 times per day. Network analysis and encryption/decryption are used less frequently, with approximately 25 and 10 processes performed per day, respectively. This information may be useful for determining which features to prioritize for further development or optimization.

Evaluation Focus: Performance.

Metrics Measured	**Test Plan**	**Test frequency**	**Results**
Response time (in milliseconds)	Conduct load testing on the Encrypto application	10	Average response time: 300ms Maximum response time: 500ms
Throughput (in requests per second)	Simulate 500 concurrent users accessing the application.	15	100 requests per second
CPU utilization (as a percentage)	Measure the response time, throughput, and CPU utilization.	20	70% utilisation of CPU

In summary, based on the performance metrics measured, the Encrypto application appears to be able to handle a moderate load of 500 concurrent users. However, the peak response time is relatively high, indicating that the application may struggle to handle sudden spikes in traffic. Further optimization may be necessary to improve the application's performance under heavy load.

3.0 Conclusions

In conclusion, Encrypto provides a convenient and efficient way to analyse data on a computer. The advantages of this application include the ability to quickly and thoroughly scan and search for relevant data, encrypt and decrypt sensitive information, and perform calculations to assist in investigations. The registry editor provides deep access to the system's configuration and can be used to track software installations, user activity and system modifications. The MD5 converter and calculator provide a way to verify the integrity of files and data. The encryption and decryption tool can be used to protect sensitive information.

However, the application does have some limitations. Firstly, it may not be able to find files that are restricted access or encrypted from other devices. Secondly, the application may not be compatible with all operating systems or file formats. Additionally, it requires a certain level of expertise and knowledge to use effectively. Another disadvantage, it may not be able to identify or access encrypted data or encrypted files with very strong encryption. The last point is that it’s important to note that this tool alone cannot replace the skills and expertise of a trained forensic analyst.

Despite these limitations, Encrypto provides valuable assistance to forensic analysts and normal users in their efforts to gather evidence, verify data integrity, and analyse information in a device. The strengths of this tool lie in its comprehensive scanning capabilities, data encryption and decryption features, and MD5 converter and calculator. It is an effective tool that can significantly streamline digital investigations, system security and is a valuable asset for forensic analysts.

4.0 Further Development or Research

If I was given additional time and resources, I could develop the application more and take it into several directions to improve its functionality and usability.

One direction could be to improve the user interface, making it more intuitive and user-friendly. This could involve incorporating more visual aids and streamlining the scanning and search processes to make it easier for users to quickly identify relevant data.

Another direction could be to expand the tool's capabilities to include more advanced data recovery and analysis features. This could involve incorporating machine learning algorithms to analyse large data sets and identify patterns and anomalies. Additionally, the application could be enhanced with more advanced algorithms for data recovery and file reconstruction.

Another area of development could be to improve the applications ability to analyse and decrypt encrypted data. This could involve incorporating more advanced decryption algorithms or creating new tools specifically designed to identify and analyse encrypted data.

Finally, the application could be expanded to include support for more devices and operating systems. This would make the tool more flexible and enable it to be used in a wider range of digital investigations.

5.0 Appendices

5.1. Project Proposal

Project Proposal

Encrypto

30/10/2022

Bachelor of science (Honours) in computer Science

Cybersecurity

2022/2023

James Kawala

X19330941

X19330941@student.ncirl.ie

Contents

1.0 Objectives

The objective of this project is to create a windows application that analyses, reports, and administrates encrypted data and software.

The name of the application is called Encrypto. This application is a forensic kit tool that is mostly used by business that deal with sensitive data or security.

The secondary features that I plan on adding to this application is memory analysis, Full disk encryption/decryption, windows password recovery, privacy checker, security recommendation features and password manager features.

The application will be compatible with 340 file types (MS documents, PDF, Cip and RAR etc) and will also be able to look through window compatible software. The application will be able to detect all encrypted files and hard disk images and report the type of encryption and the complexity of the decryption. The password recovery tools, and encryption tools have a 70% success rate.

2.0 Background

The reason why I choose to undertake this project is because of my personal interest in encrypted data and personal data security. I saw a gap in the market for an application that provided users/ businesses with information and tools for encrypted software. After researching about the topic of cryptography for a couple of days, I decided I would build a project that could inform the consumer if the files and software contained in their device was secure from third party apps or took steps to encrypt their files or software.

The way I plan on meeting my objectives is by splitting each task and function to a week-by-week time requirement. Giving myself enough time to initiate, plan, execute and monitoring and closing the project.

3.0 State of the Art

The closest competitors I found when researching ideas for this project was Passware, Ophcrack and Trininty Rescue Kit.

- Passware is an all-in-one Password recovery and encrypted evidence discovery application.
- Ophcrack is an open-source program that crack through window file passwords.
- Trinity Rescue Kit is a windows and Linux application that handles offline operations for windows systems such as rescue, repair, password resets and cloning.

The reason why my project stands out is because my application has a feature that can check for encrypted software and can also recommend the consumer security apps that would make there device more secure.

My competitors are either outdated or only carry out a specific function. My application is updated with the latest forensic tools and has a slick new design, that’s what makes it different from the rest of the competition.

4.0 Technical Approach

I will use the AGILE methodology when creating my project. During the development of my project, I will break it down 5 sections: requirement, design, testing, deployment, and review.

These sections will make it easier for me to identify my key requirements and give myself enough time to create tasks for myself.

- **Requirements**: Narrowing down the elements and function of my project, then creating a roadmap of my project and organising the tools I'm going to use for my project. During this time, I will be communicating back and forth between my project supervisor and start creating a Trello board.

  I will identify the requirements by splitting the project into two categories, functional and non-functional requirements.

  The functional requirements include scanning the system, memory analysis, Full disk encryption/decryption, windows password recovery, privacy checker, security recommendation features, password manager features.

  The non-functional requirements include application loading time, number of times the system can be scanned, data integrity, data management, data security, serviceability of the application, recoverability of the application and the accessibility of the user's data.

- **Design:** During this period, I will create a UI using Draw.io and create an interface for each feature I'm going to add to my application. Making skeletal interface for my application before developing it.
- **Testing and development:** I will begin to code the Interface of my project and then later begin developing the key features of the applications such analyses and the encryption reporting system. For each feature I finish programming, I will begin to test the features utility, integrity, and check for any bugs
- **Deployment**: Deploy application on a suitable platform and begin sending out surveys for feedback
- **Review:** Collect the surveyed data and review the apps features and requirements before developing and releasing the final version of the product.

5.0 Technical Details

The programming language used code this application is going to be Java, C, python, and C++. I will be using IntelliJ Community Edition and atom as programming software.

I also plan on using Trello (organising app), Dropbox (cloud storing application), Microsoft Teams, GitHub, Browser Stack (automated testing application), Draw. Io (App design) and Monkey survey (Survey website) for different project management phases while doing my project.

A principal library that would help to analyse large amounts data for encrypted data is Apache Spark. This library is designed for Python and Java programmes.

An important algorithm that I plan on using is the Quicksort Algorithm and hash functions. Quicksort is an efficient sorting algorithm that performs two to three times faster than most algorithms. I need this algorithm to sort items of the files for which a less than relation is defined. This algorithm would be essential for me when I am coding in Java and python.

A hash function is the most common encryption-based algorithm that helps convert a numerical input value into a compressed numerical value. I would use this algorithm to detect for any

compressed or encrypted files on the computers system. This algorithm is mostly used in C++ or C.

6.0 Special Resources Required

At this point in time, no special resources will be required for the operation and development of this project. The only resources used in this project is basic human effort, basic software licenses and computer hardware.

7.0 Project Plan

Stage 1 (24/10/2022): Planning stage

During this week I will be brainstorming project ideas and writhing down the key requirements needed for my application. I will also begin communicating with my project supervisor and begin discussing my applications features.

Project proposal and pitch will be submitted and waited on for review and feedback.

Stage 2 (31/10/2022): Designing a blueprint.

Creating a User Interface for my application using Draw.io. A blueprint of all the features will be designed and reviewed. Create an organisational and planning board using Trello.

Stage 3 (07/11/2022): Encryption and analyses system

Begin the development of the analyses and reporting tools for Encrypto. Start with basic GUI during the first few days and then begin coding the front end of the programme. During the second week begin developing the backend system of this feature.

After the features show function ability, test them out to avoid clashing with other features in the windows applications.

Stage 4 (21/11/2022): Password recovery and management features

During this period, I begin to programme the password recovery tools and management features. Front end design and implementation will be focused on during the first week and back end combability will be implemented during the second week. Testing will also be done to avoid any system crash or conflict with other feature of the window's applications.

Stage 5 (05/11/2022): Recommendation features

Create an app recommendation feature that will tell the user what apps will improve its device security. Using Java and C++ to create a simple front-end feature that scans the computer and recommends from the web. Start working on non-requirement features such as loading speed and data integrity to improve the app for consumer.

Stage 6 (16/11/2022): Privacy checker

Create the front side features for a privacy checker and then do the back-end development and testing for feature. Check for any errors and bugs that might pop and cause friction with the other elements of the application and fix the issues. Start working on the midpoint evaluation document and meet with my project supervisor. Write up the reflective report before the end of the month.

Stage 7 (02/01/2023): Testing and debugging.

Use automated testing and unit testing to check for any errors inside the applications. Fix these errors and debug the application. Refine the features to be more specific and efficient. Review the application with my project supervisor and note any features that need to be dropped or refined.

Stage 8 (23/01/2023): Prototype released, and surveys started.

Release the first version of the application to a group of volunteers via social media and email. Send out the survey along with the prototype for third party testing. Wait for all the survey data to be sent out during a two-week period.

Stage 9 (06/02/2023): Surveys collected, and feedback analysed.

Start collecting the data and sort it into useful information that can be given out in a short report. Meet with project supervisor on how the product can be improved and refined.

Stage 10 (12/02/2023) Development of final requirements

Start developing the final version of the application. Make sure that all initial features and requirements still work. Start checking off the points that need to be improved or highlighted by surveyed feedback. Add final visual and non-essential requirements such as colour, application sound and application loading screen etc.

Stage 11 (26/02/2023): Deployment and analysis

Fix any final problems and deploy the application onto a suitable platform. Finish all computing project documentation and meet project supervisor for final review and meeting. Analyse the outcomes the product receives outside a controlled environment.

Stage 12 (10/03/2023) Reviewing the final project.

Final review, submission of the project and all accompanying documentation

8.0 **Testing**

I will use the application Browser Stack to automate the testing scenarios from the client's perspective to evaluate the applications usability and performance. This program can set up framework and create test script that automate users action required for testing a windows app. An automated test will be able to show that all test scenarios can go through most of the basic requirements. An example of an automated test is running scripts that check if the program can analyse the hard drive for any encrypted file, then create a report of the number of files in the hard disk. At the end of the procedure a public key should be created, and a description should be listed of the test should be shown of the results.

Another method I can use is by getting a third party participate (volunteer) to run Unit testing on the application. By using unit testing I verify if each feature is meeting its original functional requirement. This is a good method to catch any unnoticed errors. After writing tests to ensure that the basic requirements are met, I would then need to write test cases that would catch any

component that goes outside the scope of features and make sure that the errors are noted and handled with.

Another test I can use is the integration test, this type of testing can test all the components of an application at the same time. It makes sure that certain features like model view, controller, and admin which the GUI uses can be utilised seamlessly.

The data used in this test will not be needed from user to satisfy any testing criteria. The application will only use and store data that's contained in the computer's hardware. An external source or server is not used, the program can work offline to meet its basic requirements.

5.1. Ethics Approval Application

National College of Ireland

Ethical Guidelines and Procedures for Research involving Human Participants

SEPTEMBER 2022

1. Introduction

All research involving human participants that is conducted by students or staff at the National College of Ireland should be done so in an ethical manner. The college has therefore developed an Ethics Committee, which acts as a sub-committee of the Research Committee, to ensure that ethical principles pertaining to research involving human participants are upheld and adhered to. All researchers intending to use human participants as part of their projects are thus required to reflect upon any potential ethical issues and submit their research proposals for ethical review before commencing data collection.

This document gives an overview of the core ethical principles guiding research in NCI, while also documenting the procedures required for seeking ethical approval of research involving human participants.

Am I conducting research?

Research is defined as "the attempt to derive generalisable new knowledge by addressing clearly-defined questions with systematic and rigorous methods" (NHS Health Research Authority). Sometimes, we collect data in order to evaluate a service or practice we are engaged in ("service evaluation"). The main difference between research and service evaluation is in the aim: research is trying to create new generalisable knowledge, and service evaluation is trying to evaluate whether a delivered service/practice is working well. One project may have both aims included in it. It can be confusing if a service or intervention is involved, whether or not research is being conducted. If new or competing interventions are being evaluated, then it is likely to be research, whereas if an existing service is being conducted anyway, with an evaluative component, then it is likely to be a service evaluation. Research requires consideration of the below guiding principles, whereas service evaluation does not require approval from an ethics committee.

2. Guiding Principles

In line with other research institutions, there are three core guiding principles governing the ethical conductance of research involving human participants at NCI. These principles stem from the *Belmont Report* (1979) published by the National Commission for the Protection of Human Subjects of Biomedical and Behavioural Research. While it is recognised that these principles may be operationalised differently depending on the specific research discipline, it is recommended that these are consulted as a starting point for any research involving human participants.

2.1 Principle 1: Respect for Persons

This principle entails recognition that participants should be treated as autonomous individuals and hence should never be coerced or swayed into participating in a research project against their will. The participant's right to withdraw from a research study at any time should be respected, as well as their right to dignity and protection from harm.

Respect for individuals can often be implemented in practice via the process of informed consent, whereby potential participants are made fully aware of the requirements involved in

participation. While it is recognised that in certain cases deception (i.e. the withholding of certain information from participants) may take place, this should only occur when it is robustly justified for the validity of the research. In cases where deception is justified, researchers should ensure that any potential risk resulting from this measure is minimised. Participants should also be fully debriefed on the nature of the research after it has taken place.

The principle of respect also requires researchers to protect individuals from vulnerable groups who may have diminished autonomy (see section 4.2 for more detail as to what constitutes vulnerable groups). Where full informed consent is not possible for such population groups, consent may instead be sought from their guardians. In all cases however clear assent, or willingness to participate, should be demonstrated from participants.

2.2 Principle 2: Beneficence and non-maleficence

This principle specifically focuses on the need to protect the well-being of participants. Any potential risk to participants should be minimised, whether that be risk of physical discomfort or of any psychological, emotional or social distress, while possible benefits should be maximised. Researchers adhering to this principle should thus ensure that any potential benefits derived from carrying out the study (e.g. in terms of knowledge gained) should outweigh potential risks. Even in cases where there is only a slight potential risk of harm, participants should be provided with appropriate support to alleviate this.

2.3 Principle 3: Justice

This principle emphasises the need to employ fairness in the distribution of benefits and risks to participants. The way in which participants are selected to take part in research should relate to the purpose of the study, as opposed to other factors such as availability or manipulability of participants. The exploitation of vulnerable populations should be avoided.

Where applicable, researchers are encouraged to consult guidelines stemming from their own professional bodies (e.g. The Psychological Society of Ireland) in addition to the general guiding principles above when planning their research. Researchers should also be sensitive to those issues which are specific to the population under investigation and the methodology that is employed in the project (e.g. qualitative methodologies involving the recording of data may raise issues relating to participants' right to anonymity, as well as the ethical management and use of data). Detailed consideration should be given to all these issues when planning research and when completing the Ethical Review Application form.

3. Ethics Committee

The NCI Ethics Committee was established by the Academic Council in 2012. Acting as a sub-committee to the Research Committee, its role is to oversee ethical issues arising from all research involving human participants that is conducted by students and staff of the college.

The key purpose of this committee is to safeguard against any potential harm to participants, and to ensure that their rights are recognised in line with the guiding principles outlined above.

The Ethics Committee reviews all research proposals posing ethical risk to the participants involved, however the decision as to whether projects pose ethical risk is firstly made via the appropriate Filter Committee which operates at School level (see organisational structure in Figure 1 below). The Filter Committees may review and approve research proposals which are of low ethical risk, while referring those of high ethical risk to be considered by the Ethics Committee (see categories of ethical risk in section 4.1).

While the Filter Committees are made up of staff members with subject-specific knowledge, membership of the Ethics Committee should comprise of no less than five representatives from both the School of Computing and the School of Business, including representatives from the Research Committee.

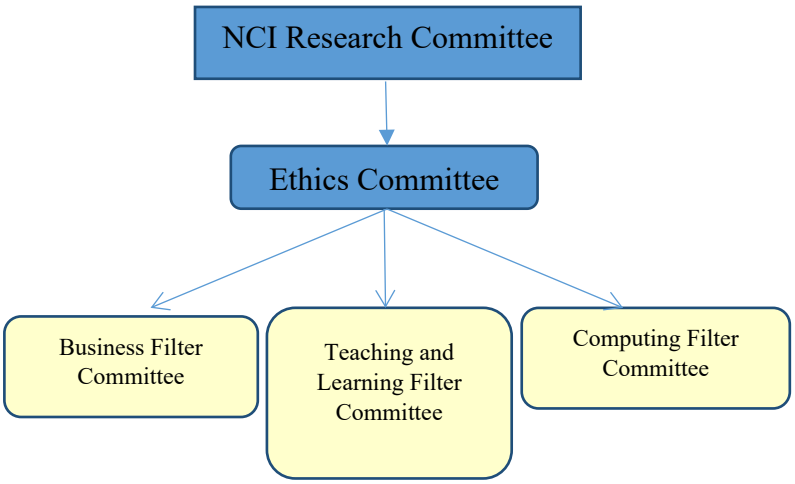


Figure 1: Committee Structures.

4. Review Process

Any staff or student of NCI wishing to conduct a study involving human participants should first submit the Ethical Review Application Form (included at the end of this document), to the relevant School Filter Committee at proposal stage. This initial review will result in a graded categorisation of ethical risk, as outlined below.

4.1 Categorisation of Ethical Risk

Research category A
Research in this category poses little ethical risk to the participants involved. Specifically, it refers to research involving human volunteers, but **excluding** studies involving:

- therapeutic interventions
- new research methodologies
- vulnerable populations (see section 4.2)

- deception of the participants
- any other significant physical, social or psychological risk to participants

Research category B
Research in this category involves human volunteers **including** studies involving:

- therapeutic interventions
- new research methodologies
- vulnerable populations (see section 4.2)
- deception of the participants
- any potentially significant risk to participants

Research Category C
This specifically refers to research involving human volunteers who are service users, patients, staff, records, etc., within the sphere of the HSE or similar setting (but not including clinical trials of investigative medicinal products).

4.2 Vulnerable groups

There are a number of participant populations that may fall under the heading of 'vulnerable groups'. These groups require consideration of unique ethical challenges regardless of the nature of the project. Research involving such populations should therefore always be reviewed by the Ethics Committee.

Groups that may be classed as vulnerable include, but are not limited to:

- Children (under 18 years of age)
- The older old (aged 85+)
- People with an intellectual or learning disability
- Individuals or groups receiving help through the voluntary sector
- Those in a subordinate position to the researcher (e.g. employees)
- Any other groups who might not understand the research and consent process

Note: in addition to the Ethical Review process, any researchers intending to work directly with children will be required to undergo Garda Vetting in advance of the proposed research.

4.3 Exemption from Full Ethical Review

In certain limited cases, researchers can apply for an exemption from full ethical review. In such cases, the Ethical Review Exemption form should be completed, explicitly detailing why the exemption is sought.

In completing this form, researchers must declare that the research does not involve any of the following:

- Vulnerable groups

- Sensitive topics
- Risk of psychological or mental distress
- Risk of physical stress or discomfort
- Any other risk to participants
- Use of drugs or invasive procedures (e.g. blood sampling)
- Deception or withholding of information from participants
- Conflict of interest issues
- Access to data by individuals or organisations other than the researchers
- Any other ethical dilemmas

4.4 Outcomes of Review Process

Following consideration of research projects submitted for Ethical Review, each Filter Committee will submit a report to the Ethics Committee summarising the applications considered and the decisions made.

For research that is deemed to fall under Research Category A (low ethical risk), a favourable outcome at the relevant Filter Committee will be sufficient to secure ethical approval. Research falling under the other two categories must however be considered by the Ethics Committee before approval may be granted.

On the basis of this review, four key outcomes may arise:

1. Research proposal approved (no recommendations)
2. Research proposal approved pending minor revisions (to be accepted by the Chair and Research Supervisor)
3. Research proposal approved pending major revisions (to be resubmitted and approved by the Ethics Committee)
4. Research proposal rejected (resubmission necessary)

A summary of the processes involved in applying for ethical approval can be seen in Figure 2.

Appeals

Appeals against the Committee's decision may be made within ten working days. In this case, at least three members of the Ethics Committee, none of whom will have reviewed the initial application, may review this along with any additional information submitted by the applicant.

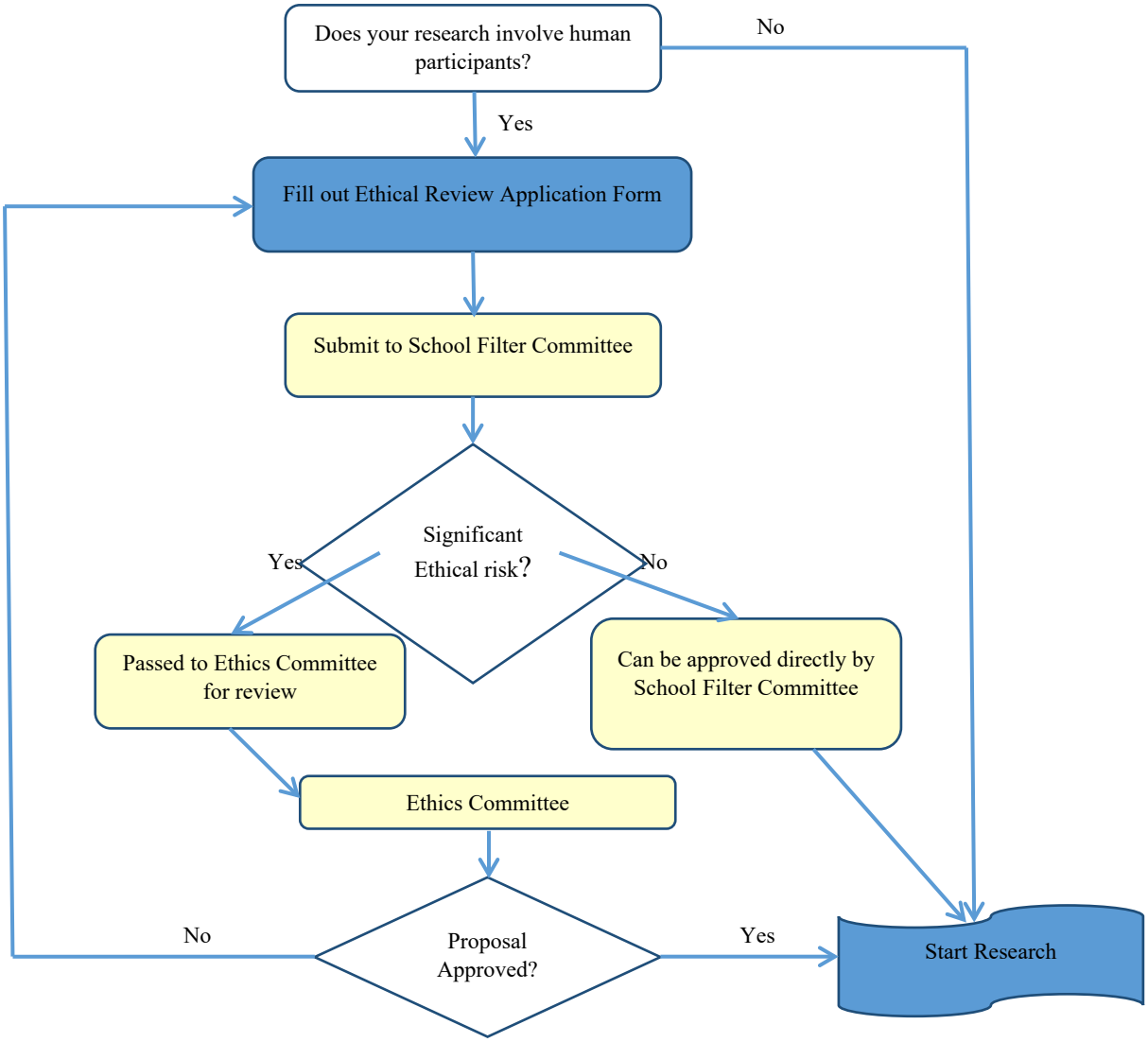


Figure 2: Process chart for seeking Ethical Approval

Ethics Application Checklist

To be submitted alongside the ethics application.

Please complete the below checklist, ticking each item to confirm that it has been addressed.

1. I agree to obtain informed written consent from all human participants aged over 18 who are involved in this research (or if circulating digitally, I will ensure that informed consent is completed, and will have the participants indicate their informed consent by continuing with their study engagement).	□ □
2. I agree to obtain informed written consent from the parents of anyone aged under 18 in this research (or from the schools if appropriate), and informed written assent from those under 18 in this research.	
3. I include a letter of agreement from a clinically responsible individual agreeing to (where appropriate) help me recruit/provide clinical support in the event that participants become distressed/host the study data collection.	□
4. I append a letter of agreement from an external institution or organisation agreeing to host the study.	□
5. I agree to comply with NCI's Data Retention Policy.	
6. I have appended a) information sheet, b) consent form/assent form, c) debriefing sheet.	□
7. I have provided details of how non-anonymised data will be stored, in a safe and encrypted manner.	□
8. I have included my contact details and those of my supervisor (where appropriate). I have only included my NCI email address and not included any personal contact information.	□ □
9. I have given sufficient details on the proposed study design, methodology, and data collection procedures, to allow a full ethical review, and I understand that my failure to give sufficient detail may result in a resubmission being required.	□
10. I understand that if I make changes to my study following ethical approval, it is my responsibility to seek an ethics amendment if the change merits ethical consideration.	□
	□

National College of Ireland

Human Participants Ethical Review Application Form

All parts of the below form must be completed. However in certain cases where sections are not relevant to the proposed study, clearly mark NA in the box provided.

Part A: Title of Project and Contact Information

Name

James Kawala

Student Number (if applicable)

X19330941

Email

X19330941@student.ncirl.ie

Status:

Undergraduate□ X
Postgraduate □
Staff □

Supervisor (if applicable)

Mr Keith Maycock

Title of Research Project

Encrypto: An encryption application

Category into which the proposed research falls (see guidelines)

Research Category A □ X
Research Category B □
Research Category C □

Have you read the NCI Ethical Guidelines for Research with Human Participants?

Yes □ X
No □

Please indicate any other ethical guidelines or codes of conduct you have consulted

N/A

Has this research been submitted to any other research ethics committee?

Yes □
No □ X

If yes please provide details, and the outcomes of this process, if applicable:

N/A

Is this research supported by any form of research funding?

Yes □
No □ X

If yes please provide details, and indicate whether any restrictions exist on the freedom of the researcher to publish the results:

N/A

Part B: Research Proposal

Briefly outline the following information (not more than 200 words in any section).

Proposed starting date and duration of project

October 2022 – May 2023

The rationale for the project

As more people are getting concerned about data and encryption. Studies have shown that people are looking for trustworthy applications to inform and check what software and files are secure and encrypted. **Therefore, a web application that checks if a software and file are secure to be used on their device.**

The research aims and objectives

Encrypto is a windows application that's designed as a forensic encryption tool that allows people to scan files and software to check if its encrypted or vulnerable. The application will extract, report, and decrypt all password protected items on a computer. **The customer will be notified about the types of files in their system and will also help users to manage and recover any sensitive data that's been encrypted in there device.** **The aim of this project is to design, implement and test a forensic kit tool for a window's device.**

The research design

A few cybersecurity specialists that provide forensic tool services will be contacted to gather the requirements specification for this application. After doing that, the functionality of the application will be designed and then implemented. The application will be tested and checked for errors using Unit Testing. The evaluation involves human participants (+18 years old) that will provide feedback regarding the functionality of the programme.

Participants permission for data gathered in this project is to be used for research purposes is obtained.

An online survey will be used to get feedback. The survey will be anonymised and therefore will follow all GDPR requirements.

The collected data will be analysed and used to improve the application. An overall description of the answers received will be included in the report.

The research sample and sample size
Please indicate the sample size and your justification of this sample size. Describe the age range of participants, and whether they belong to medical groups (those currently receiving medical treatment, those not in remission from previous medical treatment, those recruited because of a previous medical condition, healthy controls recruited for a medical study) or clinical groups (those undergoing non-medical treatment such as counselling, psychoanalysis, in treatment centres, rehabilitation centres, or similar, or those with a DSM disorder diagnosis).

Encrypto will be available to download for anyone who chooses to participate in this project. The participates will be over the age of 18.

I'm aiming to get 20 participants to use the application and to fill out the survey.

Social media (e.g., Facebook) and email will be used to invite people to take part in this study. The participants will all be volunteers.

If the study involves a MEDICAL or CLINICAL group, the following details are required:

a) **Do you have approval from a hospital/medical/specialist ethics committee?**
   If YES, please append the letter of approval. Also required is a letter from a clinically responsible authority at the host institution, supporting the study, detailing the support mechanisms in place for individuals who may become distressed as a result of participating in the study, and the potential risk to participants.
   If NO, please detail why this approval cannot or has not been saught.
b) **Does the study impact on participant's medical condition, wellbeing, or health?**
   If YES, please append a letter of approval from a specialist ethics committee.
   If NO, please give a detailed explanation about why you do not expect there to be an impact on medical condition, wellbeing, or health.

The nature of any proposed pilot study. Pilot studies are usually required if a) a new intervention is being used, b) a new questionnaire, scale or item is being used, or c) established interventions or questionnaires, scales or items are being used on a new population. If no such study is planned, explain why it is not necessary.

N/A

The methods of data analysis. Give details here of the analytic process (e.g. the statistical procedures planned if quantitative, and the approach taken if qualitative. It is not sufficient to name the software to be used).

Yes/No questions on a 1 to 5 scale will be used in the survey. The quantitative data will be collected via the survey and will be analysed.

The reporting of the results is done after the data is aggregated across all participants.

No quotes or data will be specified to any individual participant.

Excel will be utilised for statistical analysis and graphical representation of the results.

Study Procedure

Please give as detailed an account as possible of a participant's likely experience in engaging with the study, from point of first learning about the study, to study completion. State how long project participation is likely to take, and whether participants will be offered breaks. Please attach all questionnaires, interview schedules, scales, surveys, and demographic questions, etc. in the Appendix.

Participates of the study will be given a brief description of what there getting involved at the beginning of the study. Their consent will be requested and collected before taking the study.

The participants will be asked to download the application and use it on their windows PC/laptop. The participants will be given a few tasks to perform that involve the use of the application.

Once the tasks are completed, they are asked to answer the survey.

The entire evaluation process of the application will not take more than 15 minutes.

The survey answers and consent requests will be provided in Appendix A and Appendix B.

Part C: Ethical Risk

Please identify any ethical issues or risks of harm or distress which may arise during the proposed research, and how you will address this risk. Here you need to consider the potential for physical risk, social risk (i.e. loss of social status, privacy, or reputation), outside of that expected in everyday life, and whether the participant is likely to feel distress as a result of taking part in the study. Debriefing sheets must be included in the appendix if required. These should detail the participant's right to withdraw from the study, the statutory limits upon confidentiality, and the obligations of the researcher in relation to Freedom of Information legislation. Debriefing sheets should also include details of helplines and avenues for receiving support in the event that participants become distressed as a result of their involvement in this study.

Its not envisaged that there is any risk of harm or distress to participants.

Participants will be allowed to withdraw from this study anytime if they wish to do so.

Do the participants belong to any of the following vulnerable groups?
(Please tick all those involved).

- □ Children;
- □ The older old (85+)
- □ People with an intellectual or learning disability
- □ Individuals or groups receiving help through the voluntary sector
- □ Those in a subordinate position to the researchers such as employees
- □ Other groups who might not understand the research and consent process
- □ Other vulnerable groups

How will the research participants in this study be selected, approached and recruited? From where will participants be recruited? If recruiting via an institution or organisation other than NCI please attach a letter of agreement from the host institution agreeing to host the study and circulate recruitment advertisements/email etc.

Social media and email will be used.

What inclusion or exclusion criteria will be used?

+18-year-old participants will be used

How will participants be informed of the nature of the study and participation?

Invitation email or message via social media describing the study will be sent out

Does the study involve deception or the withholding of information? If so, provide justification for this decision.

N/A

What procedures will be used to document the participants' consent to participate?

At the beginning of the survey a section of it will ask the user for consent .

Can study participants withdraw at any time without penalty? If so, how will this be communicated to participants?

Yes, they can withdraw. This will be communicated in the invitation email/text.

If vulnerable groups are participating, what special arrangements will be made to deal with issues of informed consent/assent?

N/A

Please include copies of any information letters, debriefing sheets, and consent forms with the application.

Part D: Confidentiality and Data Protection

Please indicate the form in which the data will be collected.

□ Identified □ Potentially Identifiable **X □ De-Identified**

What arrangements are in place to ensure that the identity of participants is protected?

No personal data will be collected in the survey.

Will any information about illegal behaviours be collected as part of the research process? If so, detail your consideration of how this information will be treated.

It wont be collected

Please indicate any recording devices being used to collect data (e.g. audio/video).

No recording devices

Please describe the procedures for securing specific permission for the use of these recording devices in advance.

N/A

Please indicate the form in which the data will be stored.

□ Identified □ Potentially Identifiable □ **De-Identified**

Who will have responsibility for the data generated by the research?

The applicant for the Ethics approval

Is there a possibility that the data will be archived for secondary data analysis? If so, has this been included in the informed consent process? Also include information on how and where the data will be stored for secondary analytic purposes.

No

If not to be stored for secondary data analysis, will the data be stored for 5 years and then destroyed, in accordance with NCI policy?

X □ Yes □ No

Dissemination and Reporting

Please describe how the participants will be informed of dissemination and reporting (e.g. submission for examination, reporting, publications, presentations)?

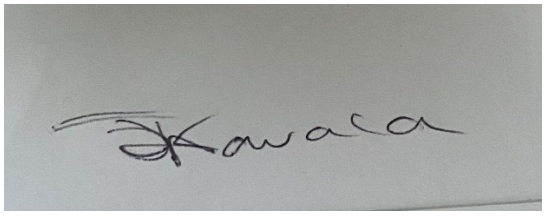
The participants will not be informed about the project report

If any dissemination entails the use of audio, video and/or photographic records (including direct quotes), please describe how participants will be informed of this in advance.

N/A

Part E: Signed Declaration

I confirm that I have read the NCI Ethical Guidelines for Research with Human Participants, and agree to abide by them in conducting this research. I also confirm that the information provided on this form is correct (Electronic signature is acceptable).

Signature of Applicant 

Date **30/10/2022**

Signature of Supervisor (where appropriate):

Date _____

Any other information the committee should be aware of?

N/A

National College of Ireland

DECLARATION OF ETHICS CONSIDERATION

School of Computing

Student Name: …James Kawala……

Student ID: …x19330941……..……

Programme Bachelor of Science (Honours) in Computer Science **Year:** …4..

Module: ……Computing Project……..……

Project Title: ……Encrypto: data encryption and analyse……..……

Please circle (or highlight) as appropriate

This project involves human participants	Yes / No

Introduction

Secondary data refers to data that is collected by someone other than the current researcher. Common sources of secondary data for social science include censuses, information collected by government departments, organizational records and data originally collected for other research purposes. Primary data, by contrast, is collected by the investigator conducting the research.

A project that does not involve human participants requires ONLY completion of Declaration of Ethics Consideration Form and submission of the form on module's Moodle page

A project that involves human participants requires ethical clearance and an Ethics Application Form must be submitted through the module's Moodle page. Please refer to and ensure compliance with the ethical principles stated in NCI Ethics Form available on the Moodle page.

The following decision table will assist you in deciding if you have to complete the Declaration of Ethics Consideration Form or/and the Ethics Application Form.

Public Data	Y	Y	Y	Y	N	N	N	N
Private Data	Y	Y	N	N	Y	Y	N	N
Human Participants	Y	N	Y	N	Y	N	Y	N
Declaration of Ethics Consideration Form	x	X	x	X	X	X	x	
Ethics Application Form	X		X		X		X	

Please circle (or highlight) as appropriate

The project makes use of secondary dataset(s) created by the researcher	Yes / No
The project makes use of public secondary dataset(s)	Yes / No
The project makes use of non-public secondary dataset(s) Approval letter from non-public secondary dataset(s) owner received	Yes / No Yes / No

Sources of Data:

It is students' responsibility to ensure that they have the correct permissions/authorizations to use any data in a study. Projects that make use of data that does not have authorization to be used, will not be graded for that portion of the study that makes use of such data.

Public Data

A project that makes use of public secondary dataset(s) ***does not need ethics permission****, but* ***needs a letter/email from the copyright holder*** *regarding potential use.*

Some websites and data sources allow their data sets to be used under certain conditions. In these cases, a letter/email from the copyright holder is NOT necessary, but the researcher should cite the source of this permission and indicate under what conditions the data are allowed to be used. See Appendix I for examples of permissions granted by Fingal Open Data, and Eurostat website.

Where websites or data sources indicate that they do not grant permission for data to be used, you will still need a letter/email from the copyright holder. For example, see Appendix II for an example from the Journal of Statistics Education.

Private Data

A project that makes use of non-public (private) secondary dataset(s) must receive data usage permission from School of Computing.

An approval letter/email from the owner *(e.g., institution, company, etc.)* ***of the non-public secondary dataset must be attached to the Declaration of Ethics Consideration****. The letter/email must confirm that the dataset is anonymised and permission for data processing, analysis and public dissemination is granted.*

Evidence for use of secondary dataset(s)

Include dataset(s) owner letter/email or cite the source for usage permission

Data licenses / Data usage Restrictions
The data from some datasets may be subject to usage limitations or restrictions. See the datasets license metadata, which can be found on the datasets Data Access Form (.html) and Dataset Attribute Structure (.das) web pages. https://www.facebook.com/privacy/policy https://help.smapply.io/hc/en-us/articles/360003749013-SurveyMonkey-Apply-Data-Privacy#h_67b98262-f35c-451c-90cc-09a26c5560d0

CHECKLIST

Non-public/private secondary dataset(s) -Owner letter/email is attached to this form ***OR*** Citation and link to the web site where permission is granted – provided in this form	Yes / No Yes / No

ETHICS CLEARANCE GUIDELINES WHEN HUMAN PARTICIPANTS ARE INVOLVED

The Ethics Application Form must be submitted on Moodle for approval prior to conducting the work.

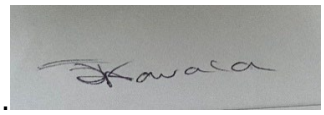
Considerations in data collection

- Participants will not be identified, directly or through identifiers linked to the subjects in any reports produced by the study
- Responses will not place the participants at risk of professional liability or be damaging to the participants' financial standing, employability or reputation
- No confidential data will be used for personal advantage or that of a third party

Informed consent

- Consent to participate in the study has been given freely by the participants
- participants have the capacity to understand the project goals.
- Participants have been given information sheets that are understandable
- Likely benefits of the project itself have been explained to potential participants
- Risks and benefits of the project have been explained to potential participants
- Participants have been assured they will not suffer physical stress or discomfort or psychological or mental stress
- The participant has been assured s/he may withdraw at any time from the study without loss of benefit or penalty
- Special care has been taken where participants are unable to consent for themselves (e.g children under the age of 18, elders with age 85+, people with intellectual or learning disability, individuals or groups receiving help through the voluntary sector, those in a subordinate position to the researcher, groups who do not understand the consent and research process)
- Participants have been informed of potential conflict of interest issues
- The onus is on the researcher to inform participants if deception methods have to be used in a line of research

I have read, understood, and will adhere to the ethical principles described above in the conduct of the project work.

Signature: 

Date: ……31/09/2022……

Appendix I

1) Fingal Open Data: http://data.fingal.ie/About

Licence

Citizens are free to access and use this data as they wish, free of charge, in accordance with the Creative Commons Attribution 4.0 International License (CC-BY).

Note: From November 2010 to July 2015, data on Fingal Open Data was published in accordance with the PSI general licence.

Use of any published data is subject to Data Protection legislation.

Licence Statement

Under the CC-BY Licence, users must acknowledge the source of the Information in their product or application by including or linking to this attribution statement: "Contains Fingal County Council Data licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) licence".

Multiple Attributions

If using data from several Information Providers and listing multiple attributions is not practical in a product or application, users may include a URI or hyperlink to a resource that contains the required attribution statements.

2) Eurostat: https://ec.europa.eu/eurostat/about/policies/copyright

COPYRIGHT NOTICE AND FREE RE-USE OF DATA

Eurostat has a policy of encouraging free re-use of its data, both for non-commercial and commercial purposes. All statistical data, metadata,

content of web pages or other dissemination tools, official publications and other documents published on its website, with the exceptions listed below, can be reused without any payment or written licence provided that:

- the source is indicated as Eurostat
- when re-use involves modifications to the data or text, this must be stated clearly to the end user of the information

Appendix II

Journal of Statistics Education: http://jse.amstat.org/jse_users.htm

JSE Copyright and Usage Policy

Unlike other American Statistical Association journals, the Journal of Statistics Education (JSE) does not require authors to transfer copyright for the published material to JSE. Authors maintain copyright of published material. Because copyright is not transferred from the author, permission to use materials published by JSE remains with the author. Therefore, to use published material from a JSE article the requesting person must get approval from the author.

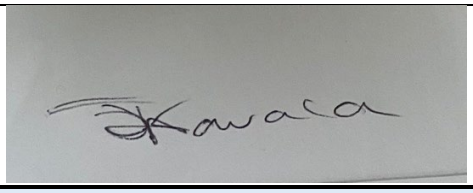
8.1. Reflective Journals

Supervision & Reflection Template

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: October

What? Reflect on what has happened in your project this month? I was researching my project idea and also deciding on the resources I was going to use. Created a design blueprint of what my project UI would look like.
So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? I researched about cryptography, and I also looked for information concerning security API. I will continue to develop my application and narrow down the features/ forensic tools to use for my windows applications.
Now What? What can you do to address outstanding challenges? Look for application that have implemented the same features that I will use for my framework.

Student Signature	

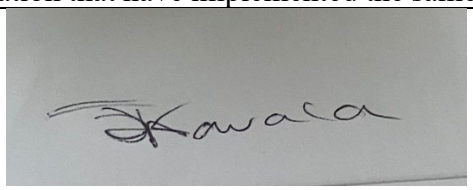
Supervision & Reflection Template

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: November

What? Reflect on what has happened in your project this month? I talked with my supervisor about my project on the 18th of November. We discussed about the API that I was going to use and to also be more specific about my applications features.

So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? I researched more about cryptography, and I also looked for information concerning API. I will continue to develop my application and narrow down the features/ forensic tools to use for my windows applications. I also finished creating my design framework for my application.
Now What? What can you do to address outstanding challenges? Look for application that have implemented the same features that I will use for my framework.

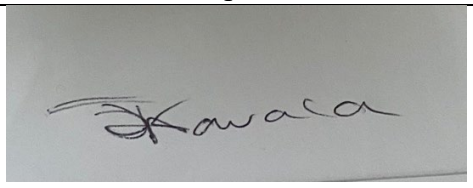
Student Signature	

Supervision & Reflection

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: December

What? Reflect on what has happened in your project this month? I submitted my prototype and uploaded my midpoint presentation
So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? At this current time, I created my first prototype and will continue to modify and narrow down my applications features and design for my 2nd updated version. My success was creating a functional application. The most challenging part of my project is integrating a unique and functional API into my system.
Now What? What can you do to address outstanding challenges? Look for application that have implemented the same features that I will use for my framework.

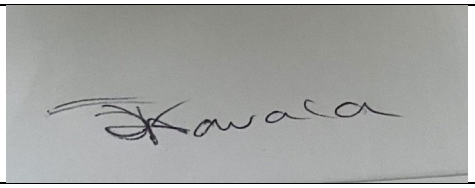
Student Signature	

Supervision & Reflection Template

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: January

What? Reflect on what has happened in your project this month? I created my file search tool and form for my quick search button that was in the main menu. I tested the feature to check if it could look through the user's directory. I was also creating the windows setting form. I changed the Menu GUI to make it more accessible for the user. Fixed a bug that didn't allow about page to appear because of a link
So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? I am able to search the user's systems directory and I have been able to finish 3 windows form for my application. I have also debugged the main menu problems that caused a lot of errors. The challenges that remain is creating a scan option form into my windows application and also adding my other tools such as the View and settings of the application.
Now What? What can you do to address outstanding challenges? Research and implement already existing information to better improve my application.

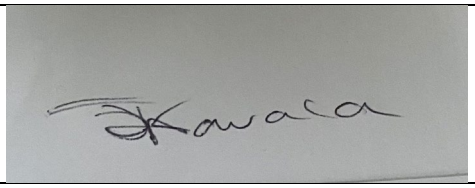
Student Signature	

Supervision & Reflection Template

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: February

What? Reflect on what has happened in your project this month? I meet with my supervisor at the start of the month to go over my project, midpoint presentation conclusion and what I need to improve on concerning my application. I created the SearchEngine and SearchOption internal class that specify whether to search for files in a directory and its subdirectories, or just in the specified directory. I also started on creating the GUI pages for the other scan options such as specific files and where to scan.
So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? A success that I made this month was to create a search directory box that allows the user to select a specific file to be scanned in the system. A challenge that I encountered was that I encountered this error "**System.UnauthorizedAccessException**" when trying to find specific files that were either zipped or encrypted.
Now What? What can you do to address outstanding challenges? Research on how to use frequency analysis or allowing the application to gain access to different types of files.

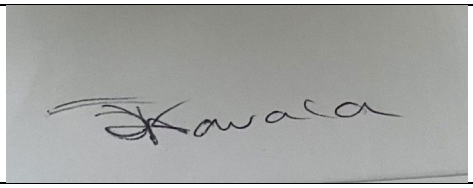
Student Signature	

Supervision & Reflection Template

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: March

What? Reflect on what has happened in your project this month? I meet with my supervisor at the start of the month to go over my project and what I need to improve on concerning my application. I revised and updated my SRS documentation for my project. I also started creating my applications settings features.
So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? A success that I made this month was to fix my applications bugs and errors. A challenge that I encountered was by creating a feature in the settings to change the applications view and creating a new session.
Now What? What can you do to address outstanding challenges?

Research on how to use frequency analysis or allowing the application to gain access to different types of files.
Student Signature 

Supervision & Reflection Template

Student Name	James Kawala
Student Number	X19330941
Course	BSHCYB4
Supervisor	Keith Maycock

Month: April

What? Reflect on what has happened in your project this month? • I meet with my supervisor at the start of the end of the month to go over my project. • Wrote up my project documentation. • Finished programming my Scanning options for my applications. • Created my showcase poster. • Automation testing and bug fixing • Started creating my presentation material.
So What? Consider what that meant for your project progress. What were your successes? What challenges still remain? • A success that I made this month was to fix my applications bugs and errors. • A challenge that I encountered was by creating a feature that recovered recently deleted files.
Now What? What can you do to address outstanding challenges? • Reduce the scope of project and simplify certain features to achieve deadlines.
Student Signature