DATA AND DE-ANONYMIZATION: AN OBSERVATION OF ANONMYZING TECHNIQUES AND THEIR SECURITY ABILITY WITHIN AN APPLICATION

by

Daniel Dixon

DIX16602092

A dissertation submitted in partial fulfilment of the requirements for the degree of

BSc (Hons) Computer Science

School of Computer Science

University of Lincoln

2019

Contents Page

1. Acknowledgements

I would like to thank Phillip Carlisle for plentiful help as a project supervisor;

Yvonne James for her assistance with certain cyber security topics;

Lastly my family and friends for their assistance with editing and moral support.

2. Abstract

Anonymization techniques are used in an attempt to preserve the anonymity of collected data. De-anonymization techniques can be used to undo this anonymity and identify the owner of said data. Both of these techniques have been researched but are not well understood in reference to each other and have not been fully implemented in code. The aim for this Project is to evaluate anonymization and de-anonymization techniques in use today and attempt to make quantifiable comparisons. This will be done so anonymizing techniques can be evaluated against de-anonymizing counterparts before being implemented on the data, allowing for future creation of bespoke options that can depend solely on the needs of the data rather than the technique most commonly used.

3. Background and Literature Review

Following the creation of the internet, computer users were given the ability to transport data across great distances between numerous and varying types of computers. The amount of data being transmitted is hard to quantify but is believed to have an approximate size of "45.7 billion pages in the Google index on January 5, 2015 " (Bosch et al, 2016), not accounting for the internet in its entirety but rather just the web pages that "are directly accessible to the everyday user". This amount is increasing every year as the world's capacity for bidirectional telecommunication grows by 28% per year while the increase in globally stored information increases by 23% (Hilbert and Lopez, 2011).

Companies have formed specifically to handle both the connections and the influx of new information that came with the internet. While doing so, these companies have started to store user information both when a user initially creates an account with the company and also throughout the account's lifetime. In the modern day, it has become impossible for a business to "not collect or hold personally identifying information" such as names and addresses (Geistfeld, 2017).

This mass collection of data creates several issues open to exploitation.
Firstly, the information may not be anonymized or stored securely by the company and hence is left open to retrieval by malicious attackers who may use it to commit identity theft or as stepping stones to enacting a data breach. These are both major issues as in the UK it is estimated that more than 100,000 people are affected by identity theft every year (Concepcion-Sanchez et al, 2018) and that data breaches are also numerous and increasing with 553 reported data breach incidents for 2008, leading to the compromise of 83 million records of personal data. If compared to 11 reported incidents and 6 million compromised records in 2003, this is a significant increase over just 5 years (Roberds and Schreft, 2009)(Edwards et al, 2016).
Secondly, user Credentials may be funnelled by the conglomerate who owns the platform into software specifically set up to help impart suggestion/influencing techniques to users and thereby sell a product or idea which may be ethically and morally wrong.
Lastly, the data collected may be sold to third parties who may also experience all of the above exploitations, leading to an exponential increase in vulnerabilities, this happened recently when "Facebook gave unfettered and unauthorized access to personally identifiable information (PII) of more than 87 million unsuspecting Facebook users to the data firm Cambridge Analytica" which then used the data to promote targeted political propaganda (Isaak and Hanna, 2018).

Attempts to solve these data security issues include using anonymization. Anonymization is a data manipulation technique that can be used to obscure input data so that key personal details can't be recovered from or linked to it. Anonymization has become increasingly common for data storage but recent techniques have been developed that allows even this anonymized data to be de-anonymized, opening it to the same problems mentioned previously. This creates vulnerabilities which can be exploited and so research into newer ways to anonymise data without the restrictions and vulnerability of older techniques has begun.

De-anonymization can be used to identify a user even if many identifiable features are removed, this is especially true if "there are no "similar" records in the multi-dimensional space defined by the attributes" (Narayanan and Shmatikov, 2008). This means that with anonymising techniques such as k-anonymity, there is "a tradeoff between anonymity and utility" (ji et al, 2015), making the data usable while also stopping the ability for it to be de-anonymized.

Anonymization has previously been compared to de-anonymization but is usually done in a theoretical sense (ji et al, 2015) or the technique is compared to previous algorithms who also have only been evaluated theoretically (Macwan and Patel, 2018). The reason for this lack of practical comparisons is unknown. Possible reasons for this include anonymization research being carried out by those whose primary field of study is not based on computing, that it is not simple to compare these techniques in a coded context or that it is seen as much easier to evaluate theoretically.

A physical coded comparison would be able to give specific processing details and would allow for a more informed decision when choosing an anonymizing technique. To discover whether this is an effective means of comparing the techniques, an application will be created that provides the details needed for such comparisons.

4. Methodology

4.1 Project Management

This project is designed to consider the generic data collected by companies and whether the anonymization of this data can be easily combined with de-anonymizing techniques in an application. To efficiently and effectively do this, a specific management style for handling the project must be selected that fits the demands specified.

The demands for the project are:

- Acquire data input similar to those given by users and have several input test cases
- Research into the implementation of several anonymization and de - anonymization techniques
- Attempt to implement and document found techniques
- Create a way to suitably show this process

Several management methodologies have been developed in the past to help enable effective project management, a number of these were researched and the findings will be discussed below.

The possible methods include both agile (XP, Scrum) and non-agile (Waterfall) methods, the former being a style of project management focussed on quick iteration and heavy discussion, Basing itself on humans not processes while also being versatile to change (Rubin, 2013). The latter focusses on planning the entire process from start to finish, staying rigid and allowing for easy measuring of progress. An outline table has been created below indicating the benefits and limitations of each method in relation to the project to show why the final decision was made:

Methodology	**Advantages**	**Disadvantages**
Waterfall	Structured and fully planned out, allowing efficient use of the limited resources by focussing on overall completion.	Once planned there is a lot of rigidity allowing for little deviation. Updates will be necessary throughout the process and thus planning would not be effectively completed.
XP (Extreme Programming)	"In XP, requirements are expressed as scenarios (called user stories), which are implemented directly as a series of tasks" (Sommerville, 2016, 77), this allows for developing based on the user and so a more usable product.	The code written for this project needs to be checked thoroughly for functionality. XP does not allow for this. Documentation is not the main concern of XP and so this report would be lacking and unexpected issues would not be trackable.
Scrum	Useful when prototyping and so is perfect for a proof of concept design. Work is broken down into manageable time increments (sprints), this could be implemented to coincide with meetings with the project supervisor. Versatility is necessary for this development to succeed and scrum being an Agile method allows this. Testing and evaluation occur during development and so will minimise the amount needed at the end.	Not all techniques used in Scrum are usable in the context of the project as a whole team is necessary to use them, whereas the project will be completed by a singular developer. Scope Creep caused by new ideas being thought up or researched can increase work to a point that it can't be completed in the time given.

The optimal method was decided as Scrum and this will be mixed with Waterfall to gain the best results. Waterfall will be used initially to decide a singular goal and to plan how it will be broken down, these smaller objectives will then be subdivided further into sprint goals for use during scrum sprints. Scrum techniques will also be used verbatim or tweaked to help keep development on track while also still being usable with a single person development team.

4.2 Software Development

This project should output an artefact that is similar to a learning tool, it will place both anonymizing and de-anonymizing techniques into the same destination and explain what is happening to the user. To develop this artefact several features need to be implemented that are in relation to the previously discussed requirements. They will be considered the main milestones and will include:

- Importing CSV files into an internal database
- Allowing users to choose the anonymizing technique they wish to use
- Display information on the anonymizing technique they have selected
- When activated, manipulate the database in alignment with the selected technique
- Have a further option to test a de-anonymizing technique on the data (anonymized or not) with the same approaches as above

To deal with the CSV input, SQL may be necessary as it is the most common way to store database information, although in this case a coding library that handles input data similar to SQL might be more useful if further implementation occurs that allows input of different file types.

Anonymizing techniques will be taken from the algorithms found while carrying out research and they will be completed on the back-end while a front-end GUI or console display will tell the user what is occurring.

De-anonymizing will be completed in a similar way to the anonymization, displaying a description and output while the process runs in the background.

4.3 Toolsets and Machine Environment

When deciding on Tools to use the discussion mainly focussed around how to incorporate the chosen management style into the development process, what language should be used and whether a GUI was necessary.

Shown below is a table that was created to discuss the advantages and disadvantages of each option individually. Possible combinations for these tools are also discussed further down:

Name of Tool	Tool Development Area	Basic Description	Advantages	Disadvantages
Product Backlog	Management	Full Goal List	Able to see all milestones for the project.	The Size of the list can be continually increased.
Sprint Backlog	Management	Broken down goal list	Divides the task into more manageable sizes. Gives a sense of progression.	By itself it does not give an accurate account of how much time should be used on each smaller goal.
Sprint Reviews	Management	An evaluative look at how the previous sprint progressed	Due to interaction with the Project Supervisor being on a weekly basis, scrum sprints and reviews are perfect for this development, allowing for effective updates and keeping development on track.	If not completed correctly or is too harsh then the development team may become indifferent to the project.
Pair Programming	Management	Testing while coding	Can gain an objective view of the development from a secondary source. Completes testing while the code is taking place, reducing a need for debugging.	Necessary to have more than one project participant who is knowledgeable about code for the process to be useful.
Meeting List	Management	Schedule for interaction	Gives accurate recording of when interactions happened for easier evaluation.	Not entirely necessary as it has been stated the meetings took place weekly, in-between sprints consistently.
Gantt Chart	Management	Weekly Planner	Allows the ability to plan out the entirety of development before beginning. Works well when connected to the more flexible sprint backlogs.	Will not be accurate as it can't predict all future necessary revisions or outside factors that may affect development.
Python	Development	Coding Language	A higher-level design makes the language simple to implement. Large number of useful libraries allow for modularity with code.	Inter-library functionality can be an issue as they are sometimes developed for different versions of the coding language. Error handling can be tricky as data types aren't

				usually explicitly declared.
C++	Development	Coding Language	A lower level language that allows for very deliberate implementations. Comes with a large amount of functionality and code customizability.	Does not have garbage collection unless coded in. Does not have some of the more useful higher-level elements.
Git	Development	Version Control	A complete record of development, allowing for easier evaluation of progress.	May not be necessary with sprint backlogs existing.
Eclipse	Development	Programming IDE	Has several plugins, allowing for different coding languages to be used. Can allow for either a console application or GUI.	Even with plugins the IDE is highly focussed on C++ development.
Spyder	Development	Programming IDE	Allows for quick iteration and is simple to use.	Only useful for Python development.
Qt Creator	Development	GUI Creation IDE	Creates full integration of C++ code within a GUI. Large amounts of support from company and community.	Entirely a C++ IDE. Proprietary code and datatypes only useful to the IDE
wxWidgets / wxPython	Development	GUI Creation Library	Fully Cross-Platform.	Very rigid in design and focus may be taken away from completing the task to work heavily on GUI placement.

Combinations considered were as follows:

Combination	Used in Project	Reasoning
Product Backlog, Sprint Backlog, Sprint Reviews, Meeting List, Gantt Chart, Python, Git, Spyder, wxPython	Was not used for project	Python and its development environments are easy to use and previous experience in this area has been obtained, especially in GUI creation. Research found that there can be problems with both debugging and interdependencies of libraries however and so this combination was not used.
Product Backlog, Sprint Backlog, Sprint Reviews, Gantt Chart, C++, Git, Eclipse, Qt Creator	Chosen for project	Using Qt Creator means interacting with the restrictive licensing imposed by Qt, this licensing is less imposing for open source solutions however and so will not be an issue for this project. Use of Eclipse for non-GUI coding is to ensure if issues occur further in development that revision is possible. This option provides the most efficient solution while also being flexible and so was used.
Product Backlog, Sprint Backlog, Sprint Reviews, Meeting List, Pair Programming, C++, Git, Eclipse, wxWidgets	Was not used for project	This combination was created for the possibility of failure of other tools. The meeting list was added for if meetings with the project supervisor could not occur frequently, pair programming for if the development team grew in size and wxWidgets for if Qt was not functional. Implementation was not necessary on any counts.

4.4 Research Methods

The output of comparing data manipulation techniques, such as anonymization and de-anonymization, is influenced by the input data being manipulated. For this project, attempts will be made to base data being inputted on data that is collected by modern companies. Research was conducted into how using targeted marketing techniques, a part of many modern companies, involves collecting user data and collating it for a more accurate record on the user. This is then used for attempting to sell a product or service more effectively. The data used in this process can be split into four categories:

Identity Data, Quantitative Data, Descriptive Data and Qualitative Data (Roberts, 2013)

Of these categories, the main focus will be on recreating Identity data as it contains information which enables unique identification which is the basis for anonymization. This category usually contains both nominal and ordinal data as the data is always clearly labelled and the order of this data can have an effect. The data is also not clearly defined and cannot be effectively compared to each other by empirical means so will not contain interval or ratio data.

All evidence collected during this project will eventually be collated into a comparing model which will in part be built upon data collected from previous research and partly collected from the artefact. When adapting this model and acquiring the artefact portion of the evidence, data will be collected in an objective manner and will be empirical in nature to then combine with the theoretical data found during research. This empirical data could include giving the work and span complexities of each technique, timing the amount of time needed to complete a technique or the amount of data that must be anonymized for it to then not be able to be de-anonymized. The finished artefact will eventually be released as open source so critique from users of the artefact can critique it and development can be focussed on rectifying and improving based on this. For an effective critique the data must be able to build on previous findings and users must be able to objectively verify this; meaning the artefact must provide either interval or ratio data. This is in line with the empirical nature given to the data and so should be easily developed.

5. Software Design, Development and Evaluation

5.1 Requirements

The requirements for this project were considered initially by creating a UML case diagram. This is shown in Appendix 1 and indicates that the user should be able to choose the specific actions that occur to a data file and have control of the anonymization and de-anonymization process. It also shows in detail how the input data should interact with the algorithms. This speculation is then turned into a product backlog which breaks down the goals stated earlier into smaller chunks, following the Scrum methodology. The backlog is shown below:

ID	Product Backlog Item	PBI Type	Priority	Effort Required
1	Literature research and review	Knowledge Acquisition	9	4
2	Decide final topic to cover	Knowledge Acquisition	9	2
3	Project management	Knowledge Acquisition	8	3
4	Software development	Knowledge Acquisition	8	3
5	UML case diagram creation	Knowledge Acquisition	5	1
6	Determine coding language and syntax	Knowledge Acquisition	7	2
7	Choose and research coding libraries	Knowledge Acquisition	6	2
8	Use coding language to read in CSV Files	Feature	8	5
9	Create and fill data files	Feature	7	3
10	Develop a GUI	Feature	8	4
11	Merge developed code with GUI IDE	Change	9	7
12	Fix issues with file input	Defect	6	7
13	Display anonymised data	Feature	9	6
14	Anonymise file input	Feature	8	7
15	Display de-anonymised data	Feature	9	6
16	De-anonymise file input	Feature	9	9
17	Implement extra techniques	Feature	6	8

5.2 Design

The design was initiated by recording the different techniques on offer that were found during the research stage. They include:

Name	Technique Type	Summary
k-Anonymity	Anonymization	A type of aggregation anonymization, all unique data is removed from the database
L-Diversity	Anonymization	An extension of k-Anonymity that aims to resolve the weaknesses created by it
Class Clustering	Anonymization	Grouping similar user data to reduce how identifiable the data is
Noise Addition	Anonymization	Taking precise initial data and adding imprecision
Substitution	Anonymization	Replacing an identifiable value with a value not linked to the original (e.g. height = 5'7 replaced with the colour blue)
Differential Privacy	Anonymization	Using statistical probability to manipulate data in such a way that the possibility of deanonymization is stopped while also being able to reverse the manipulation to retain utility
Singling out	De-anonymization	When data is found to be unique and so can be used to identify the user
Linkability	De-anonymization	Linking two or more records together by overlapping or similar information, this can come from more than one source
Inference	De-anonymization	Deducing new personal information not explicitly present in the original data through data known about the individual

(ji et al, 2015) (Dwork and Roth, 2014) (Gambs et al, 2013)

Initially k-Anonymity and Singling out would be the two techniques developed with noise addition and linkability as techniques to be implemented if time permits.

The next stage involved deciding where to acquire data for input. It was believed better to start small with a custom dataset before possibly importing a much larger set found in use today like other research papers have (Gambs et al, 2013). Research led to discovering a list of data types the social media company Facebook collects on its users (Dewey, 2016). This list is by no means complete but does give a large amount of options that would likely be kept by a social media platform and could be used as a guideline for creating custom data. The features given by the website were then input into Appendix 2. Next this appendix was subjectively evaluated based on the perceived vulnerability to exploitation and given a rating of low, medium or high vulnerability. Below is a small extract from Appendix 2:

Data Collected	Threat Level	Used In project	Possible uses/other information found if data is acquired
1. Location	High	Y	Movement Knowledge, Knowledge of future expenditures
2. Age	High	Y	D.O.B, Political views
3. Generation	Low	N	Age (range)
4. Gender	Medium	N	Social views, General political views
5. Language	Low	N	Country (if language is niche), region
6. Education level	Medium	N	Computer Literacy
7. Field of study	Medium	Y	Computer Literacy
8. School	Medium	N	Socioeconomic background, Political views, Computer literacy
9. Ethnic affinity	Medium	N	Socioeconomic background, Political views
10. Income and net worth	High	Y	Socioeconomic background, Political views, Account type

As shown several features were chosen based on their perceived threat level but also on the data category they fell under, defined in the research methods section (Roberts, 2013). The data used in these initial data sets would include:

ID, Firstname, Surname, County, Postcode, DOB, Field of Study, Job Title and Income

ID was not on the list in Appendix 2 but was chosen as a permanent identifier, with each entry given a number to locate them even if other attributes are lost in the anonymization process. Firstname and Surname were also not on the list but are defined as name information within the identity data list. Location data was split into County and Postcode to allow for more testing possibilities in the future by being able to partially anonymize postcode or to anonymize either county or postcode separately to see the results, they also both fall under the postal address information category. Age was displayed as the users Date of Birth (DOB), which is classified as person information and is stored in the form DD/MM/YYYY, again to increase opportunities while testing. Lastly, Field of Study, Job Title and Income were added as features that could be wholly anonymized and are created similarly to the Job Information section of the identity data category.

5.3 Development

Initial development focussed on completing the task shown in the Gantt Chart, created as a timeline of predicted events and shown within Appendix 3, this chart combined with the product backlog was then converted into a sprint backlog. This backlog is shown below:

Sprint No	Start Date	End Date	Tasks	Completed
1	29/10/2018	04/11/2018	Literature research and review	Literature research and review
2	05/11/2018	11/11/2018	Literature research and review	Literature research and review
3	12/11/2018	18/11/2018	Literature research and review	Literature research and review
4	19/11/2018	26/11/2018	Decide final topic to cover	Decide final topic to cover
5	03/12/2018	09/12/2018	UML case diagram creation, Determine coding language and syntax, Create and fill data files	UML case diagram creation, Determine coding language and syntax, Create and fill data files
6	10/12/2018	16/12/2018	Use coding language to read in CSV Files	
7	17/12/2018	16/01/2019	Develop a GUI, Use coding language to read in CSV Files	Develop a GUI, Use coding language to read in CSV Files
8	17/01/2019	23/01/2019	Merge developed code with GUI IDE	
9	24/01/2019	30/01/2019	Merge developed code with GUI IDE, Fix issues with file input, Display anonymised data	Merge developed code with GUI IDE, Display anonymised data
10	31/01/2019	06/02/2019	Fix issues with file input	
11	07/02/2019	13/02/2019	Fix issues with file input, Anonymise file input	Fix issues with file input
12	14/02/2019	20/02/2019	Anonymise file input, Display de-anonymised data	Anonymise file input
13	21/02/2019	27/02/2019	Display de-anonymised data	Display de-anonymised data
14	28/02/2019	06/03/2019	De-anonymise file input	
15	07/03/2019	13/03/2019	De-anonymise file input	
16	14/03/2019	20/03/2019	De-anonymise file input	
17	21/03/2019	27/03/2019	De-anonymise file input	De-anonymise file input

18	28/03/2019	04/04/2019	Implement extra techniques	

These sprints were also reviewed to evaluate and better understand what went well or badly, allowing for better decision making in future sprints. This Sprint Review is shown below:

Sprint No	Sprint Review Comments
1	Research was started into data security and integrity and attempts were made to focus on a single goal as the field is vast. Data scraping and de-anonymization research was started to observe and decide which ideas and algorithms could be converted into code.
2	Observed that social media data is the best way to access large amounts of data and it may be worth researching what data is stored by them and how it is stored/used. Data collection software was researched including Wireshark (TCP and other Packet Data) and Burpsuite (Html and Web page scraping). The work environments necessary to collect this data was also researched, showing that Kali Linux and Parrot OS both have the tools already installed but it was also found that installing them on a Linux distribution (such as the one in use already by the development team) is easily compatible.
3	This sprint focused heavily on de-anonymisation of data and a large amount of recent research was found on the subject. GitHub was then searched to discover whether implementations existed and if so which ones. Few existed and none could be found that weren't highly specialised. In view of the previous discovery that social media data is incredibly useful and prevalent, the possibility exists of showing data owners what happens to their data within businesses.
4	A decision has been made that data scraping cannot be set up realistically or safely in the time given and results would not show anything significant. This means focus has shifted to de-anonymization going forward and data will be taken from datasets created by others that are accessible to anyone or through creating custom datasets that are similar to those in real world use.
5	A UML case diagram was created to understand how a user might interact with the software. C++ has been selected as the language to use for development as it offers flexibility and interoperability.
6	Coding started using the Eclipse IDE. Errors occurred during parts of the development process including reading the file in from the wrong location.
7	GUI creation for initial page progressed well with no errors within the QtCreator IDE. The work on file input within Eclipse completed, attempts should be made in future to combine the two IDEs.
8	Merging has started. Qt creator uses proprietary variable types so a conversion of the Eclipse code to the Qt format must be made. There were particular issues with the reading files as the QFile library is quite different to the one used within the base C++ code.
9	All issues with merging were fixed and files imported correctly. The file can also display anonymised data when complete as the GUI has been developed for that page. Work on implementing the anonymizing function/s begins in the next sprint
10	Although the data can be retrieved from specific files, issues occurred when passing the data through to different GUI windows and a fix to the header files may be necessary
11	A workaround was found where instead of passing through the data, it can be set to a global value, accessible to the entirety of the application. It may be necessary to

	evaluate the security of this action. A K-anonymity function has been developed and helps show the proof of concept that can now be tested on the newly accessible data.
12	Tweaks were needed to the function for it to work for the dataset but nothing major was found so it was finished quickly. A similar style was taken to develop the De-anonymizing GUI window as the anonymizing one, which took longer than expected due to unforeseen circumstances.
13	The GUI display was quickly finished and a de-anonymizing algorithm was chosen.
14	Attempts made to create the function for de-anonymization, but because of no set algorithm and prior commitments, work has slowed. Time needs to be assigned to this work as a priority.
15	Time has been allocated each week to complete the de-anonymizing function. The output is not correct and trawling through the code has had no positive results. This must be completed during the next sprint.
16	After spending more time sifting through the code, an error has been spotted in how the function views different features. This will need to be corrected.
17	The function works correctly. More techniques can be added in due time but the proof of concept is completed.
18	Techniques were attempted, specifically implementing noise addition. Unfortunately, issues occurred with the complexity of allowing users to add the noise and being able to manipulate the original data.

As shown, after the main research segment had been completed, the first step was then to fill the data files with attributes that fit features chosen in the design section. To do this in a more objective manner, much of the data was chosen based on several sources that provided the information. ID was assigned in numerical order and consequently no source was needed, similarly the first name and surname were picked based on names chosen by the development team. The county of the user data was based upon the counties in England only. Postcode was based on those in use within the previous stated counties. DOB was based on a random calendar generator between the years 1930 and 2000. Field of Study was based on condensing down ten major fields found within the Canadian Census Dictionary (Statistics Canada, 2010). They were condensed as shown:

Major Fields	**Condensed version**
educational, recreational and counselling services	Education
fine and applied arts	Art
humanities and related fields	Humanities
social sciences and related fields	Social Science
commerce, management and business administration	Business
agricultural, biological, nutritional, and food sciences	Agriculture
engineering, applied sciences technologies and trades	Engineering

health professions and related technologies	Medicine
mathematics	Mathematics
computer and physical sciences	Computing

Job title and income, like first and surname, was assigned by the development team.

Because all information used has previously shown to be based on two sources both of which discuss the data usage of companies, the information used has fulfilled the purpose of fundamentally being similar to data that would be collected had those features been actually used by companies.

After filling the datasets, development turned to the techniques discussed in the research and design section, the implementation of which was briefly explained within the sprint reviews and will be discussed in more detail here.

Initially, the file input was developed using a data file and splitting the data using commas as the delimiters. This works perfectly with CSV (Comma Separated Variable) files. The Limits on importing are that the file must include a top row of headers within the file and the data must be filled in for each column to ensure correct transfer. Testing of this will be discussed later in this report.

For anonymization the k-anonymity algorithm was chosen initially, this involved allowing the user to choose which columns to anonymise and then initialising a loop that replaced the data in those columns with asterisks.

Singling out unique data points was how the k-anonymized data was deanonymized, making the user aware of every column in the data and which has unique and therefore identifiable elements within it.

5.4 Testing

A battery of tests was completed for each of the main three stages of the artefact. The first set of tests was completed against the file loading system, they are shown below:

Test Number	Testing Parameters	Expected Results	Actual Results	Changes Made
1.1	Import a Jagged data file (some rows have more values than headers)	Only data that falls within the range of the headers will be displayed	As expected, data outside header range was not shown	N/A
1.2	Other data types than string data is entered into the data file	All imported data is converted to string so no issues should occur	Results as expected.	N/A
1.3	Import a file with a .txt extension but still have data separated by the comma delimiter	Data still uses delimiter so should import correctly	Because a check on extension was in place the file was not imported	Removed .csv extension as a prerequisite, fixed issue
1.4	Import a file that is not in the parent	Issues may occur with input of	No issues occurred; file	N/A

	directory but rather a child directory	backslashes into the file locator	imported with no issues	
1.5	Import the file from a completely different location by entering the entire address into the search location	Again, there may be issues when using backslashes but also	Location had been tagged onto the default file location and so would not load	Remove exact searching and have the user fully enter the location

Secondly, testing was enacted upon the anonymization GUI and techniques used:

Test Number	Testing Parameters	Expected Results	Actual Results	Changes Made
2.1	Use the k-Anonymity function on the inputted data	The columns chosen to be anonymized have the data within them replaced by a "*"	Results as expected	N/A
2.2	Return to original screen by pressing the back button	The window returns to the main window	This was not possible as including the main window header crashes the document	Several attempts were made to find a fix for the issue but none could be found
2.3	After implementing an anonymization function upon the input data, click on a different technique	The description should show for this separate function	Results as expected	N/A
2.4	Run an anonymizing function before running a different one	No data should be left on the GUI apart from the manipulation of the input data from the initial function	Data from the initial function was left on the GUI	Remove all references to the original function on the GUI when choosing a new function

Lastly, the testing of the de-anonymization GUI and techniques used:

Test Number	Testing Parameters	Expected Results	Actual Results	Changes Made
3.1	De-anonymize the data using the "singled" function	Columns with unique data will be displayed	Results as expected	N/A
3.2	Input a dataset with multiple columns with unique data	The code will identify and display all columns	Results as expected	N/A
3.3	Identify tables that have been anonymized	There is no check in place to do so, leading to it not being found	Results as expected	A feature was implemented that keeps track of anonymized data
3.4	Run a de-anonymization function before running a separate instance of a different function	No data should be left on the GUI apart from the manipulation of the input data from the initial function	No data was left over	N/A

5.5 Operation and maintenance

To operate the artefact, the user runs the executable and is brought to the window shown in Appendix 4, they are shown the file location that the application will search, giving the user knowledge about where to store their own files when using the application themselves. Once chosen the data will be imported and displayed on screen and the user can ensure that it has imported correctly before moving on to choose the anonymization technique. The GUI, shown in Appendix 5, displays different data depending on the anonymizing technique chosen, for example, the description and options will change when switching from k-anonymity to any other technique. The next few steps are very selective depending on the technique used but they will ask the user questions about the data and let them verify which parts they wish to be anonymised. After clicking the button to anonymise they can further manipulate the data with more techniques or instead move onto de-anonymising, taking us to the final GUI, Appendix 6. When using this window, the user will perform similar actions as those in the previous window but instead of the questions focussing on the original inputted data, they instead work from the basis that the data is already anonymized.

Because the techniques used for anonymization and de-anonymization are constantly being developed, updates to the software would need to be constant. This is not so much an issue as the time taken to research and develop techniques will be in the range of months to years.

6. Conclusion

This paper aimed to answer the question "Can Anonymizing and De-anonymizing techniques be merged into a singular application and be compared when doing so". Modern methods of anonymization and de-anonymization were identified through research and implementation was attempted. Then these techniques were then collated into a coded application that could be used as either a teaching tool to those less knowledgeable on the subject or to more easily compare the techniques when using them for personal or corporate datasets.

After investigating the techniques in use today there seems to be less of an issue than initially thought, these newer techniques are developed based on the vulnerabilities of previous techniques, allowing anonymizing techniques to be continually improved and remove the possibilities of de-anonymization. This could mean that a comparison is not necessary but instead a list of improvements over previous iterations would better serve to answer the aim of this project.

Although the aim was to create a full comparison model, this was not possible with restraints that occurred during the development process, this unfortunate outcome may be rectified with further development however and the development that did occur allowed for the techniques to be merged and compared even if not to the standard hoped for.

Overall, the artefact is a sufficient starting point towards fully developing a system for comparing anonymizing and de-anonymizing data manipulation techniques meaning it fulfils the goal set out during the research stage.

7. Reflective Analysis

When looking back on the project, several areas could have been improved.

Initially, it was difficult to define and manage the scope efficiently; this was because the project topic was chosen based on personal interest in an area of study that was largely unexplored personally, leading to motivations changing as more was learned. If a concrete goal could have been set initially and the large field of study the topic came from was understood in a shorter timeframe this problem may have been avoided.

Although the file input worked well for CSV files, there were several constraints on inputting different file extensions. The formatting of the input only allowed for files where a comma was used as a delimiter and so as shown in testing .txt files can be imported in the same way as a CSV file would. This was not tested for any other extension types and so further testing of the range of extensions that are usable with the application would eventually allow for inclusivity of most or all file types.

A definite improvement that could be made would be the inclusion of more techniques for fuller and more compelling comparisons and collation of those comparisons into a model. An example of this is the introduction of techniques such as l-diversity, an extended version of k-anonymity, allowing for a comprehensive look at the improvements that have been made between the two and the de-anonymization possibilities for both.

Much of the techniques used were focussed on purely relational data, using social media platforms and the data they provide would have given a more complicated but also more in depth look at how data interacts. Social networks have recently become a large part of de-anonymizing research and so interest in physically implementing this area was also of interest (Su et al, 2017) (Lee et al, 2018). Because this type of technique is focussed on data that is created within the social media platform, it wouldn't be possible to compare algorithms within the artefact unless access to this data could be easily acquired and that the interactions that occur could be converted into a usable database format, making for a very difficult task to complete.

8. References

Bansal, S. Kumar, P. Rawat, S. and Choudhury, T. (2018) Analysis and Impact of Social Media and it's Privacy on Big Data. In: *International Conference on Advances in Computing and Communication Engineering*, Paris, France, 22-23 June. IEEE, 248-253. https://ieeexplore-ieee-org.proxy.library.lincoln.ac.uk/stamp/stamp.jsp?tp=&arnumber=8458066

Beigi, G. Shu, K. Zhang, Y. and Liu, H. (2018) Securing Social Media User Data - An Adversarial Approach. In: *29th ACM Conference on Hypertext and Social Media*, Baltimore, MD, USA, 9-12 July. ACM, New York, NY, USA, 165-173.
https://arxiv.org/pdf/1805.00519.pdf

Bosch, A.v.d. Bogers, T. and Kunder, M.d. (2016) Estimating search engine index size variability: a 9-year longitudinal study. *Scientometrics*, 107(2) 839-856.

Concepcion-Sanchez, J.A. Molina-Gil, J. Caballero-Gil, P. and Santos-Gonzalez, I. (2018) Fuzzy Logic System for Identity Theft Detection in Social Networks. In: *4th International Conference on Big Data Innovations and Applications*, Barcelona, Spain, 6-8 August. IEEE, 65-70.

Dewey, C. (2016) *98 personal data points that Facebook uses to target ads to you*. Available from https://www.washingtonpost.com/news/the-intersect/wp/2016/08/19/98-personal-data-points-that-facebook-uses-to-target-ads-to-you/?noredirect=on&utm_term=.ea0920be13a7 [accessed 14 December 2018].

Dwork, C. and Roth, A. (2014) The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4) 211-407.

Edwards, B. Hofmeyr, S. and Forrest, S. (2016) Hype and heavy tails: A closer look at data breaches. *Journal of Cybersecurity*, *2(1)* 3–14.

Gambs, S. Killijian, M. and Cortez, M. N. d. P. (2013) De-anonymization Attack on Geolocated Data. In: *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, Melbourne, Australia, 16-18 July. IEEE, 789-797.

Geistfeld, M.A. (2017) Protecting Confidential Information Entrusted to Others in Business Transactions: Data Breaches, Identity Theft, and Tort Liability. *NYU Law and Economics Research Paper*, 17(30) 385-412.

General Data Protection Regulation (GDPR). (2018). General Data Protection Regulation (GDPR) – Final text neatly arranged. Available at: https://gdpr-info.eu/ [Accessed 23 October 2018].

Hilbert, M. and Lopez, P. (2011) The World's Technological Capacity to Store, Communicate, and Compute Information, *Science*, 332(6025) 60-65

Hilbert, M. and Lopez, P. (2012) How to Measure the World’s Technological Capacity to Communicate, Store, and Compute Information Part I: Results and Scope. *International Journal of Communication*, 6 956–980

Isaak, J. and Hanna, M. J. (2018) User Data Privacy: Facebook, Cambridge Analytica, and Privacy Protection. *Computer*. 51(8) 56-59

Ji, S. Li, W. Mittal, P. Hu, X. and Beyah R. (2015) SecGraph: A Uniform and Open-source Evaluation System for Graph Data Anonymization and De-anonymization. In: *24th USENIX Security Symposium*, Washington, D.C., 12-14 August. https://www.princeton.edu/~pmittal/publications/SecGraph-USENIX15.pdf

Lee, W. Liu, C. Ji, S. Mittal, P. and Lee, R.B. (2018). Blind De-anonymization Attacks using Social Networks. *Computing Research Repository*.
https://arxiv.org/pdf/1801.05534.pdf

Loker, M. (2018). CONVENIENTLY EXPOSED: HOW THE CONVENIENCE OF THE INTERNET IS EXPOSING YOU TO IDENTITY THEFT. *Journal of Internet Law*, 22(2) 3-8.

Macwan, K. and Patel, S. (2018). k-NMF Anonymization in Social Network Data Publishing. *The Computer Journal.* 61(4) 601–613.

Statistics Canada (2010) *Major Field of Study (MFS).* Available from https://www12.statcan.gc.ca/census-recensement/2006/ref/dict/pop063-eng.cfm#tphp [accessed 08 February 2019]

Narayanan, A. and Shmatikov, V. (2008) Robust De-anonymization of Large Sparse Datasets. In: *IEEE Symposium on Security Privacy (sp 2008)*, Oakland, CA, USA, 18-22 May. IEEE, 111-125. Available from https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4531148

Rihaczek, K. (1991). Data protection in networks. *Studies In Health Technology And Informatics*. 1(1) 249-270.

Roberds, W. and Schreft, S. (2009). Data breaches and identity theft. *Journal of Monetary Economics*, 56(7) 918-929.

Roberts, J. (2013) *A comprehensive checklist for auditing different data types in a CRM or Email marketing system*. Leeds: Smart Insights. Available from https://www.smartinsights.com/customer-relationship-management/customer-privacy/types-customer-data/ [accessed 05 April 2019].

Romanosky, S. (2016). Examining the costs and causes of cyber incidents. *Journal of Cybersecurity.* 2(2) 121–135.

Rubin, K.S. (2013) Essential Scrum: a practical guide to the most popular Agile process, Boston: Addison-Wesley.

Selznick, L. and LaMacchia, C. (2018). Cybersecurity Liability: How Technically Savvy Can We Expect Small Business Owners to Be?. *Journal of Business & Technology Law. 13(2).*

Sommerville, I. (2016) *Software Engineering*. Boston: Pearson Education Limited.

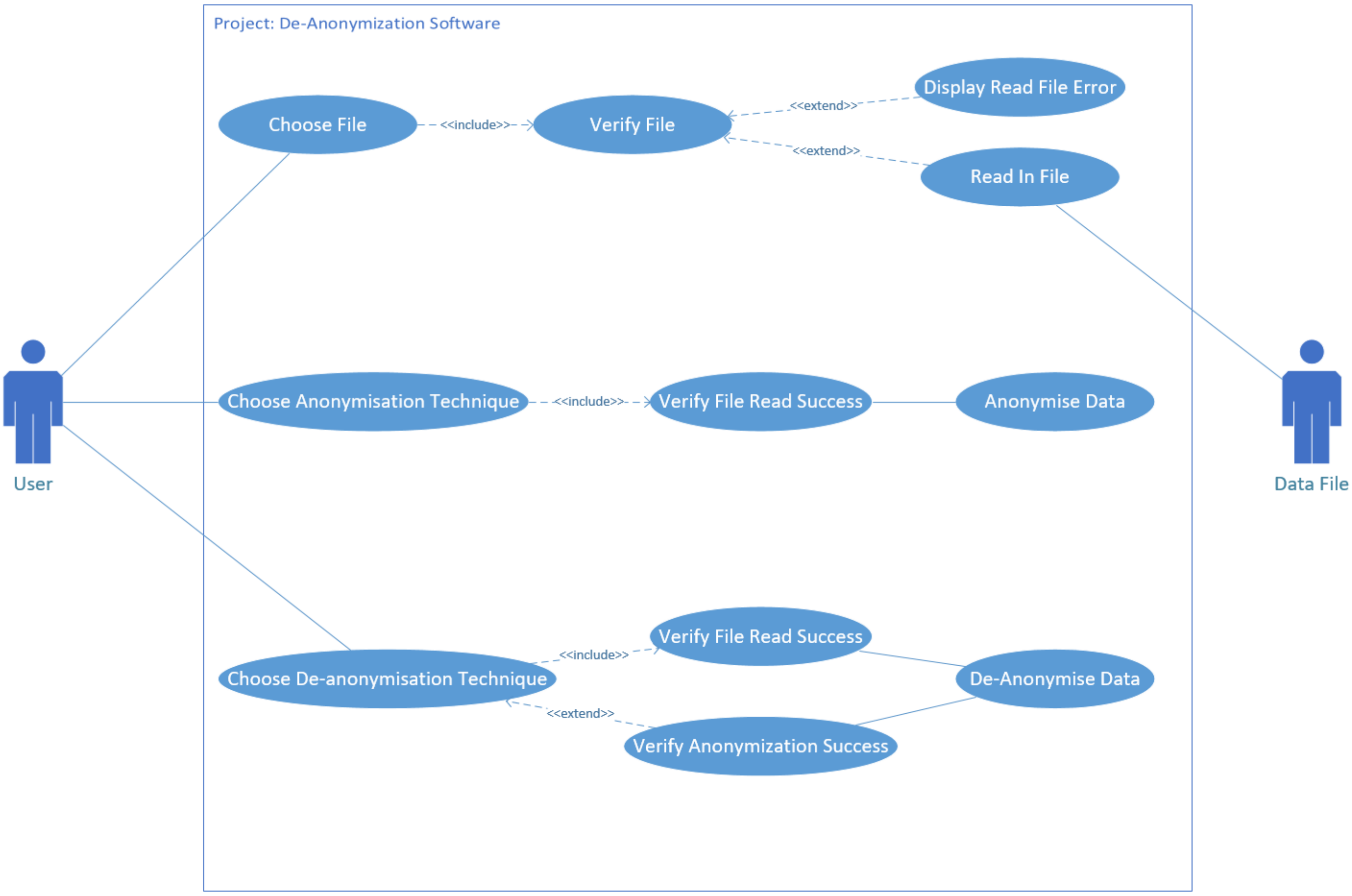
Su, J. Shukla, A. Goel, S. and Narayanan, A. (2017) De-anonymizing Web Browsing Data with Social Networks. In: *26th International World Wide Web Conference*, Perth, Australia, 3-7 April. Perth, Australia: IW3C2, 1261-1269.

Tian, W. Mao, J. Jiang, J. He, Z. Zhou, Z. and Liu, J. (2018). Deeply Understanding Structure-based Social Network De-anonymization. *Procedia Computer Science.* 129, 52-58.

9. Appendices

All Appendices are shown below or attached to this report in supporting documentation.

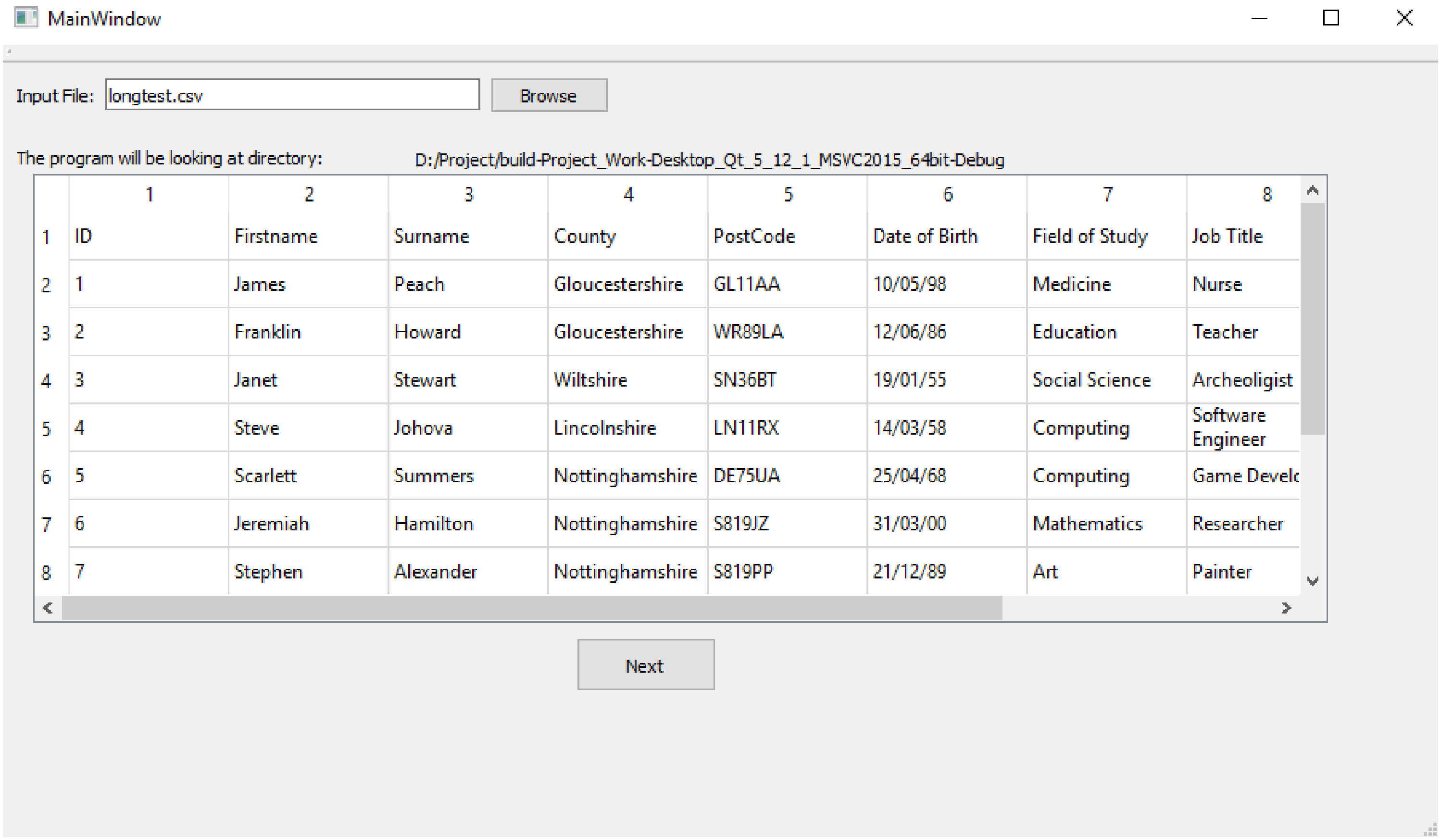
Appendix 1: UML Case Diagram



Appendix 2: Facebook Sold Data – Within Supporting Documentation

Appendix 3: Gantt Chart – Within Supporting Documentation

Appendix 4: Main GUI window

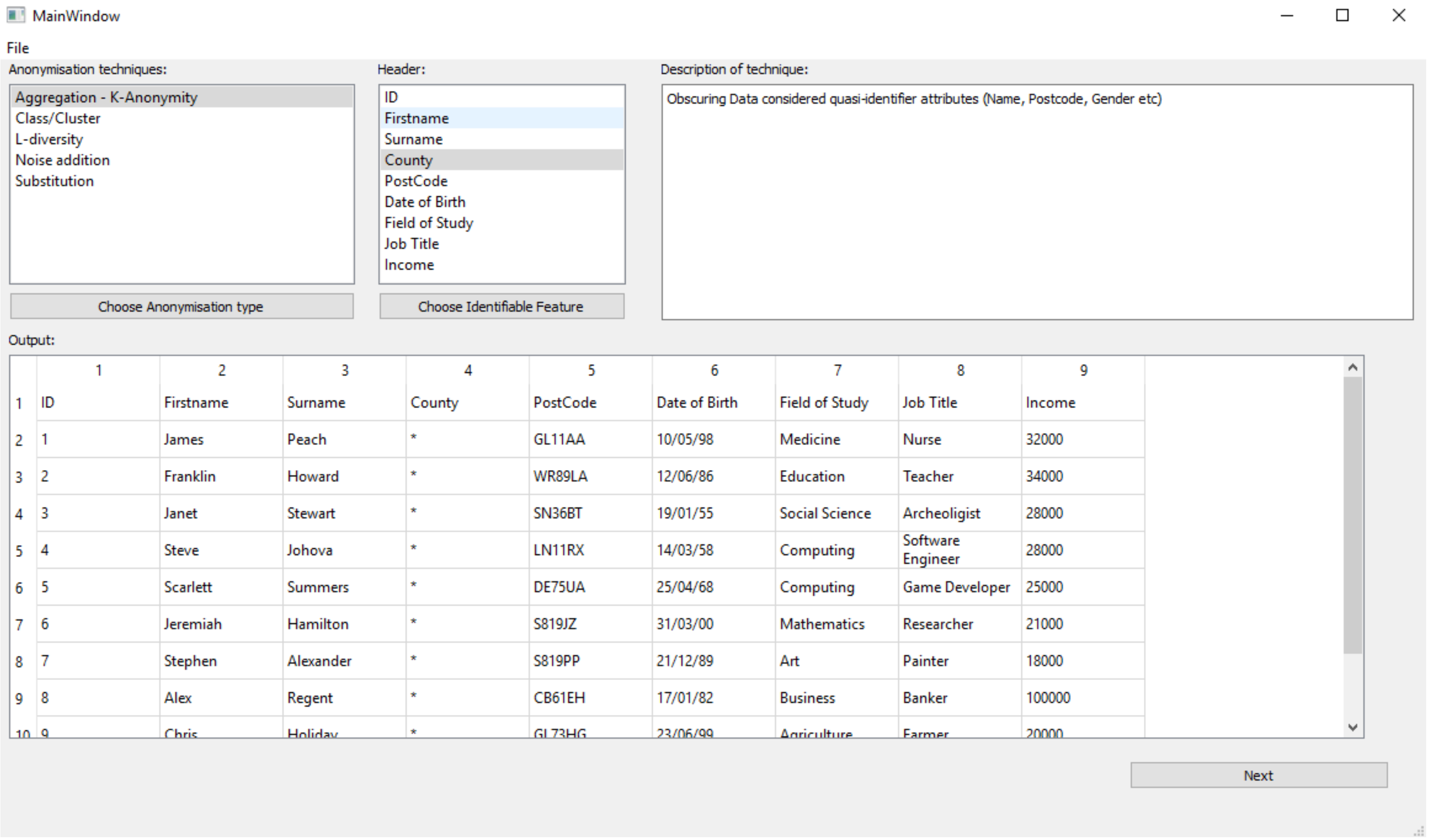


Input File: longtest.csv

The program will be looking at directory: D:/Project/build-Project_Work-Desktop_Qt_5_12_1_MSVC2015_64bit-Debug

	1	2	3	4	5	6	7	8
1	ID	Firstname	Surname	County	PostCode	Date of Birth	Field of Study	Job Title
2	1	James	Peach	Gloucestershire	GL11AA	10/05/98	Medicine	Nurse
3	2	Franklin	Howard	Gloucestershire	WR89LA	12/06/86	Education	Teacher
4	3	Janet	Stewart	Wiltshire	SN36BT	19/01/55	Social Science	Archeoligist
5	4	Steve	Johova	Lincolnshire	LN11RX	14/03/58	Computing	Software Engineer
6	5	Scarlett	Summers	Nottinghamshire	DE75UA	25/04/68	Computing	Game Develc
7	6	Jeremiah	Hamilton	Nottinghamshire	S819JZ	31/03/00	Mathematics	Researcher
8	7	Stephen	Alexander	Nottinghamshire	S819PP	21/12/89	Art	Painter

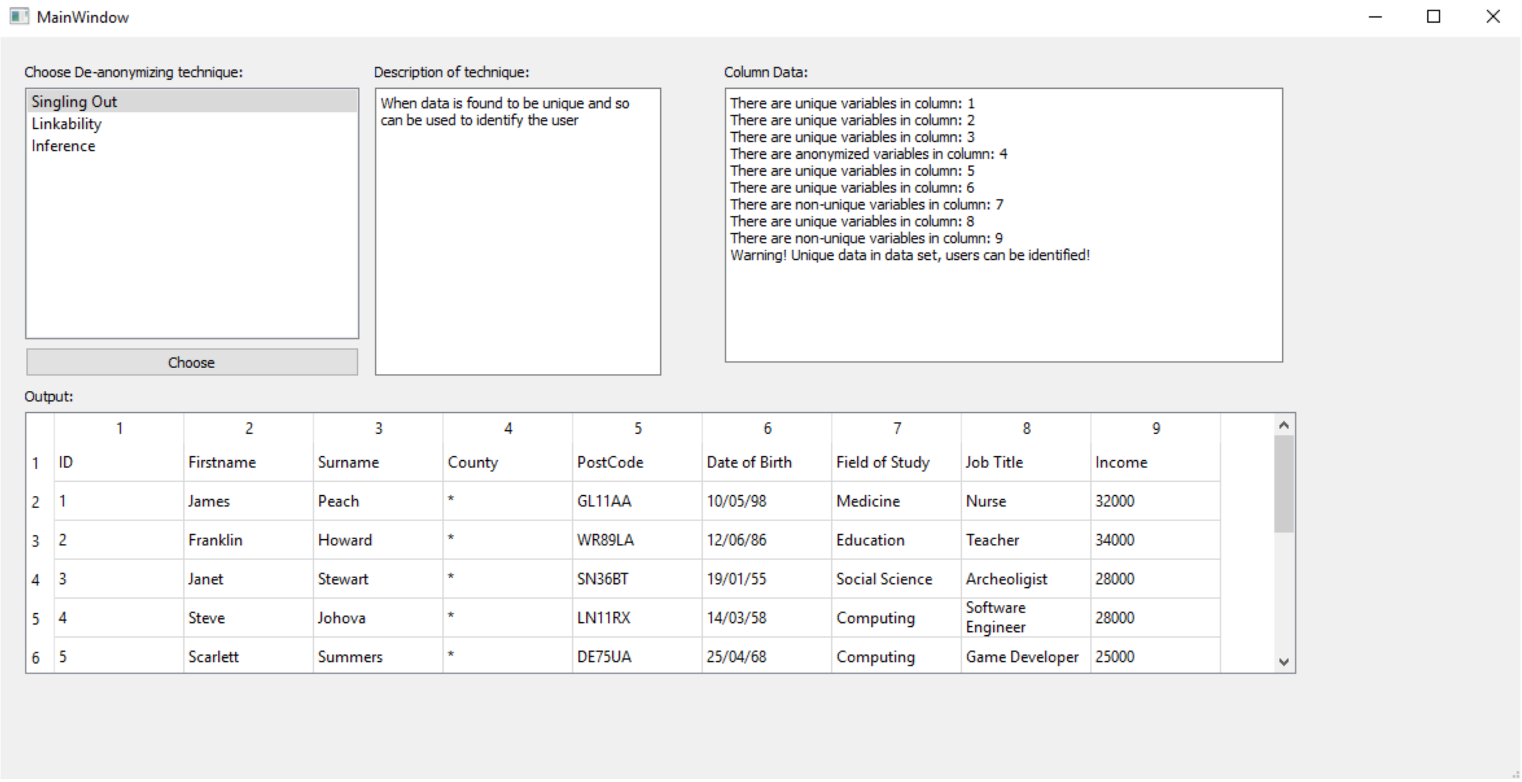
Appendix 5: Anonymizing GUI window



Obscuring Data considered quasi-identifier attributes (Name, Postcode, Gender etc)

	1	2	3	4	5	6	7	8	9
1	ID	Firstname	Surname	County	PostCode	Date of Birth	Field of Study	Job Title	Income
2	1	James	Peach	*	GL11AA	10/05/98	Medicine	Nurse	32000
3	2	Franklin	Howard	*	WR89LA	12/06/86	Education	Teacher	34000
4	3	Janet	Stewart	*	SN36BT	19/01/55	Social Science	Archeoligist	28000
5	4	Steve	Johova	*	LN11RX	14/03/58	Computing	Software Engineer	28000
6	5	Scarlett	Summers	*	DE75UA	25/04/68	Computing	Game Developer	25000
7	6	Jeremiah	Hamilton	*	S819JZ	31/03/00	Mathematics	Researcher	21000
8	7	Stephen	Alexander	*	S819PP	21/12/89	Art	Painter	18000
9	8	Alex	Regent	*	CB61EH	17/01/82	Business	Banker	100000
10	9	Chris	Holiday	*	GL73HG	23/06/99	Agriculture	Farmer	20000

Appendix 6: De-anonymizing GUI window



When data is found to be unique and so can be used to identify the user

```
There are unique variables in column: 1
There are unique variables in column: 2
There are unique variables in column: 3
There are anonymized variables in column: 4
There are unique variables in column: 5
There are unique variables in column: 6
There are non-unique variables in column: 7
There are unique variables in column: 8
There are non-unique variables in column: 9
Warning! Unique data in data set, users can be identified!
```

	1	2	3	4	5	6	7	8	9
1	ID	Firstname	Surname	County	PostCode	Date of Birth	Field of Study	Job Title	Income
2	1	James	Peach	*	GL11AA	10/05/98	Medicine	Nurse	32000
3	2	Franklin	Howard	*	WR89LA	12/06/86	Education	Teacher	34000
4	3	Janet	Stewart	*	SN36BT	19/01/55	Social Science	Archeoligist	28000
5	4	Steve	Johova	*	LN11RX	14/03/58	Computing	Software Engineer	28000
6	5	Scarlett	Summers	*	DE75UA	25/04/68	Computing	Game Developer	25000