

Master Thesis

DESIGNING FOR USABILITY A STATISTICAL DISCLOSURE CONTROL TOOL FOR MICRODATA SETS

ANSHIKA RAWAT

This page is intentionally left blank

Designing for Usability: A Statistical Disclosure Control Tool For Microdata Sets

Master Thesis

by

Anshika Rawat

to obtain the degree of **Master of Science**
in **Management of Technology** at
The Faculty of Technology, Policy and Management
Delft University of Technology

To be defended publicly on August 27th, 2020

Student number: 4904362

Thesis committee: Prof. dr. ir. M.F.W.H.A. (Marijn) Janssen, TU Delft, Chair & First supervisor
Dr. F. S. (Seda) Gürses, TU Delft, Second supervisor
Dr. ir. M. (Mortaza) S. Bargh, Research and Documentation Centre (WODC),
Ministry of Justice and Security,
External supervisor

Acknowledgements

Before you lies the master thesis “Designing for Usability: A Statistical Disclosure Control Tool for Microdata Sets”. It has been written to fulfil the graduation requirements for the master’s programme in Management of Technology at the Delft University of Technology.

The project was undertaken as part of an internship at the Research and Documentation Centre (abbreviated as WODC in Dutch) of the Dutch Ministry of Justice and Security. The research was a part of the Privacy-Utility – Tools 2.0 (PU-Tools 2.0) project.

This thesis wouldn’t have been possible without the excellent guidance and support of my supervisors. I would like to express my warmest gratitude to Prof. Janssen who recommended me to WODC for the internship and was ever so kind enough to provide me with valuable feedback at every stage of my thesis. Many thanks to Dr. Mortaza without whom this project would have been incomplete. His constant support and commitment to go beyond his way to help me is deeply appreciated. I am forever thankful to Prof. Seda for always bringing a new perspective to our discussions and challenging me to widen my horizons. Her insights helped me refine the finer aspects of my work.

I gratefully acknowledge all my friends who were always there to lend a helping hand and provide much needed distractions. Thank you to all my dear friends who are spread across different time zones.

Finally, I would like to thank my family. Words are not enough to express my gratitude to you. Thank you for being my biggest cheerleaders and always believing in me. This journey wouldn’t have been possible if it weren’t for you.

*Anshika Rawat
Delft, August 2020*

Executive Summary

Governments across the world looking to implement Open Government Data (OGD) initiatives undergo many problems. One such problem is the risk to privacy from opening data sets as most of the data is at a microdata level which corresponds to specific individuals. A solution to such a predicament is the application of Statistical Disclosure Control (SDC) techniques on microdata. SDC methods anonymize microdata that reduces the risk of disclosure while also maintaining the value of the data.

SDC methods can be applied by using software tools, however, these tools are designed from the perspective of experts or for the purpose of demonstration. Moreover, ongoing research has led to the slow progress in not only the development of these tools, but also their adoption. Resulting in limited support material and even smaller user base. As a consequence, individuals or organizations looking to adopt these tools to satisfy their data privacy objectives cannot use them.

Out of many SDC tools, ARX is a stable application that equips its users with an arsenal of techniques to anonymize microdata sets. It also undergoes regular updates, thus keeping pace with the current developments in the field of SDC techniques. Despite this, ARX is not widely used due to its perceived complexity.

This thesis addresses the problem of the complexity that is associated with ARX which makes it difficult to adopt them to anonymize their data sets. The thesis provides a solution to this problem by developing a prototype tool which reduces the complexity of SDC techniques through a simplified, user-friendly approach to data anonymization. The thesis does not aim to enhance privacy methods or improve the functionalities of already existing tools by proposing a replacement.

This problem statement is tackled by answering 5 research question. They are:

- *What are the typical challenges of using SDC tools?*
- *What are the usability problems with ARX in practice?*
- *What are the requirements to overcome these problems?*
- *How are the requirements translated into a design?*
- *How is the resulting design perceived by users of ARX?*

The research methodology first follows a systematic literature review to gain insights on the challenges that could be associated with the use of SDC tools. The literature on user acceptance models indicates that the challenges could be related to factors that motivate a person to use a technology. Theory on software usability suggests that the usability of a tool stems from how the tool is designed. If software guidelines are not followed while designing a software and keeping user needs in mind, then the resulting software design could be dismissed by the users. The literature also suggests employing techniques like use-case modelling to identify user needs and their interactions with the system. Also, QUIM (Quality in Use Integrated Map) model identifies specific measurable software criteria that can be measured through a Likert scale survey.

This was followed by a mixed-method approach to investigate problems users faced while

using ARX and to provide better method triangulation. The demographic of the subject group was identified via a stakeholder analysis. The research resulted in identifying six problems associated with ARX that were observed in both the qualitative and quantitative analysis.

These problems were then translated to user requirements and formed part of the software requirements needed to develop the prototype. The problem descriptions along with the user requirements are given below:

Problem Area	Description	Solution (User requirement)
Minimal Memory Load	The user is required to keep minimal amount of information in mind in order to achieve a specified task	Minimalistic design to avoid visual clutter
		Consistent interface elements based on existing mental models
		Offloading tasks by using default values or visual clues for decision making
Self-descriptiveness	The desktop application conveys its purpose and give clear user assistance in its operation	Intrinsic methods to relay information
		Use of simple, unassuming language
		Providing contextual functions and information
		Instinctive placing of visual metaphors
User Guidance	The user interface provides context-sensitive help and meaningful feedback	Principle of tunnelling and selective attention through multi-step pathway forms with inline validation for task completion
Navigability	It is easy to navigate the desktop application in an efficient way	Defining a clear primary navigation area
		Minimal hierarchical structures that embrace predictability such as a left-hand side navigation menu
Minimal Action	The desktop application helps the user achieve their task in minimal number of steps	Streamlining and grouping similar task actions on one page/tab of the screen
Familiarity	The user interface has recognizable elements and interactions that can be understood by the user	Incorporating predictable design elements in pace with current trends

In addition, functional requirements were identified through SDC literature to define the domain-related features of the prototype. Privacy models formed the basis of the anonymization approach. General-purpose data utility metrics were selected as they are more applicable for open data initiatives and lastly, attacker model metrics were taken to analyse the risk. These functionalities were streamlined and reduced to avoid choice overloading leading to a simplified tool. Lastly, user interactions were defined through use-case modelling and the prototype called Danaamta was developed.

Finally, Danaamta was evaluated by conducting quantitative interviews to measure the contrast in its usability against ARX. The results revealed that the user group preferred Danaamta over ARX, but there was no significance in the results of the metric that measured whether Danaamta reduced the complexity that is associated with the theories of data anonymization process. It was perceived that the user could complete the specific task using either of the applications.

However, users found their task simplified and resulted in less confusion on what needs to be done which is often associated with a lack of supporting documentation. This is majorly attributed to a design decision that avoids choice overloading. Choice overloading is a concept that is mainly addressed in e-commerce applications. However, the research showed that this concept also applies to complex applications such as ARX where there are multiple functionalities to achieve a singular purpose. Therefore, when designing such applications, elimination of choice overloading should be taken into consideration as a software guideline.

Moreover, upon reflecting on the theories of software guidelines it can be concluded that applying them does not guarantee the success of a software. They have to be filtered according to user needs and the context in which the software will be perceived useful.

Despite the limitations of the study which included modifying the research approach with a small sample size given the COVID-19 situation, shortcomings of the prototype design and the QUIM model, the research provided some valuable insights and contributions.

The thesis tried to bridge the gap which implicitly occurs when privacy tools are designed from the perspective of experts. It is understood that the protection of private data should only be handled by experts. However, to build that expertise, people have to be introduced to simpler tools without being overwhelmed by the complexity that is immanent with concepts of SDC.

Contents

List of Figures	vii
List of Tables	viii
1 Introduction	1
1.1 Background	1
1.2 Problem Statement	3
1.3 Research Objective	4
1.4 Research Questions.	5
1.5 Research Methodology	6
1.6 Thesis Outline	6
2 Literature Review	7
2.1 Methodology	7
2.2 Stakeholder Analysis	7
2.3 Overview of Adoption Models	9
2.3.1 TAM	9
2.3.2 UTAUT	11
2.4 User Interface Design	12
2.5 Measuring Software Usability	13
2.6 Conclusion	16
3 Investigating Usability of ARX	18
3.1 ARX	18
3.2 Purpose	19
3.3 Interview Setup	19
3.3.1 Interview Method and Content	19
3.3.2 Interviewee Selection.	21
3.3.3 Method of Analysis	22
3.4 Interview Results	22
3.4.1 Qualitative	22
3.4.2 Quantitative	24
3.4.3 QUAL + QUAN	26
3.5 Conclusion	27
4 Prototype Requirements	28
4.1 Software Requirements.	28
4.1.1 User Requirements	28
4.1.2 Functional Requirements	33
4.2 Use Case	36
4.3 Conclusion	37
5 Prototype	38
5.1 General Description	38
5.2 Page Description	39
5.2.1 Dashboard	39

5.2.2	Configure settings.	39
5.2.3	Analyse Results	40
5.3	Evaluation.	41
5.3.1	Purpose	41
5.3.2	Setup	41
5.3.3	Method of Analysis	41
5.3.4	Results	42
5.4	Conclusion	45
6	Discussion	46
6.1	Reflection	46
6.2	Contributions	47
6.2.1	Theoretical Contribution	47
6.2.2	Practical Contribution.	47
6.3	Limitations	48
6.4	Future Research.	49
6.5	Recommendations.	49
6.6	Relevance to the MOT programme	50
7	Conclusion	51
	Bibliography	53
	Appendices	60

List of Figures

1.1	Crime map of burglaries within a 5 kilometer radius of Amsterdam [4]	2
1.2	Problem Statement	4
1.3	Research Objective	5
1.4	Research Design	6
2.1	Stakeholder mapping on a Power/Interest Matrix	8
2.2	Overview of Adoption/Acceptance Models	10
2.3	Technology Acceptance Model (TAM) [33]	10
2.4	Unified Theory of Acceptance and Use of Technology (UTAUT) [98]	12
2.5	Measurable criteria in QUIM [84]	15
2.6	Relationship between factors and criteria in QUIM [84]	16
3.1	Graphical User Interface of ARX	19
3.2	Convergent Parallel Design	22
3.3	Graphical representation of the quantitative results	25
3.4	Convergence of qualitative and quantitative findings	26
4.1	Screen on startup in ARX	31
4.2	Multiple tabs in ARX	32
4.3	Knob button in ARX	33
4.4	Functional steps involved in k-anonymity privacy model	36
4.5	Use case for anonymizing a data set	37
5.1	Sitemap of Danaamta	39
5.2	Graphical representation of the quantitative results for ARX	42
5.3	Graphical representation of the quantitative results for Danaamta	43
A.1	Screen on startup, part 1	79
A.2	Screen on startup, part 2	79
A.3	Raw data is uploaded on the dashboard	80
A.4	Configure settings page	80
A.5	Specific configurations are loaded for k-anonymity	81
A.6	Anonymized data is displayed on the dashboard	81
A.7	Analyse data utility page	82
A.8	Analyse data risk page	82

List of Tables

2.1	Guidelines organized under six functional areas of user-system interaction [89]	13
3.1	Results of qualitative data analysis	23
3.2	Results of quantitative data analysis	25
4.1	User requirement solutions for addressing the six usability problems of ARX . . .	29
5.1	Results of quantitative data analysis	44

Introduction

1.1. Background

The government treats information as an instrument of policy and is, therefore, an information collector, producer, provider, and user [34]. Each year, colossal amounts of data is collected and generated through administrative procedures by public organisations. With tremendous data at hand, governments are taking an initiative towards Open Government Data (OGD). OGD is defined as *government data* i.e any data and information produced or commissioned by public bodies, can be *opened* i.e freely used, re-used and distributed by anyone [97]. Thus simply put, these initiatives aim at opening the data produced by governmental institutions to the public. The purpose of OGD initiatives is to create value from data. These initiatives specifically seek to improve the understanding of the public into the operations of the government, empower them to make informed decisions and participate in governance processes, and most importantly to generate social and commercial value to simulate the economy through innovation [6]. Ultimately, the end goal is to maximise the value of data. As Clive Humby said, “*data is the new oil*” [16], just like oil data is valuable only if it is refined.

Across the globe, governments are seeking to enhance their transparency, accountability and efficiency by setting up an OGD initiative that aims at stimulating the sharing of its data sets with the public or with other organisations [19]. The data sets pertain to information that is gathered by the different branches of the government to perform their task. Public bodies are said to be among the largest creators and collectors of data across different domains [48]. These data domains range from public health, crime, social security, climate, ecology, tourist information, statistics to a myriad of other different topics.

However, the implementation of these initiatives can be undermined by the challenges it entails. Technical, legal and financial restrictions, among others, may limit data accessibility and valuable ways to re-use data [97]. One of the most common challenges is the opening of data responsibly for microdata sets. Microdata refers to data at the individual respondent level. Opening of these data sets carries an inherent risk to privacy. Each individual in a microdata file can be identified by two kinds of variables called key variables and sensitive variables. Key variables are direct (e.g., name) and indirect identifiers (e.g., phone number) of an individual whereas sensitive variables contain information that is sensitive to an individual (e.g., criminal record) [35]. By default, it is understood that key variables should be foremost removed from the data set before publishing. However, there is always a risk of re-identification from sensitive data if a person looking to misuse the published data attempts to link a target person

with the record on the released microdata. Therefore, it is a challenge to make them accessible without disclosing the identity or any other confidential information about the respondents of the data [88].

Disclosure of personal data is considered as one of the main threats for data opening and therefore, only a small proportion of a selective microdata set can be made available to the public. An example of this is the Crime Map. It is an initiative of the Dutch Police that provides a list of burglaries per region of the past three months, refer Figure 1.1. These burglaries are shown as dots, which is in the midst of a four-digit postcode area on the map, to not correspond to a specific home ensuring the privacy of the victim(s) [4].

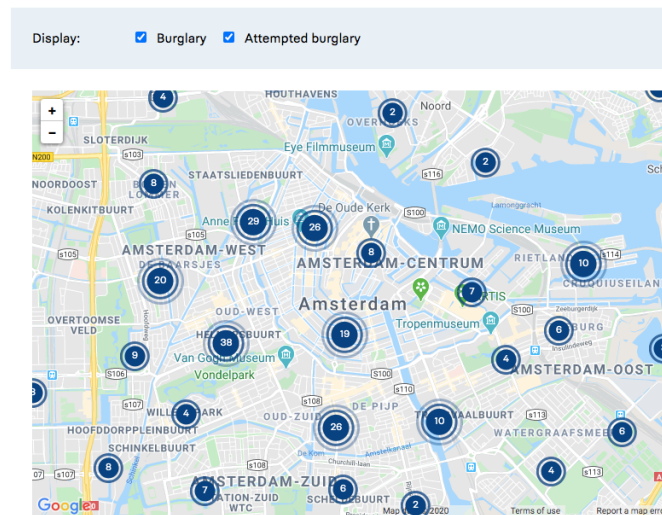


Figure 1.1: Crime map of burglaries within a 5 kilometer radius of Amsterdam [4]

When publishing anonymized microdata, one has to defend against all kinds of adversaries as there may exist an adversary who knows all or some attributes of an individual enough to make a linkage [57]. Thus, in most cases, depending on the type of microdata it is not possible to disclose it to the public. Data such as 'registered rape victims' or 'offence characteristics of suspects of a crime' can be regarded as highly sensitive in nature. And any attempts at removing the sensitivity in such records can render the data useless. This impacts the transparency of these organisations and impedes value creation as the full potential of these data sets cannot be achieved. Moreover, in the European Union (EU), data controllers have to be compliant with laws like General Data Protection Regulation (GDPR) before sharing and opening their data sets as it is of paramount importance to protect the privacy of the respondents of the data [19].

Disclosure risk in microdata is generally presented as:

1. **Identity disclosure:** It refers to the identification of an entity (such as a person or an institution) [54].
2. **Attribute disclosure:** It refers to an intruder finding out something new about the target entity [54].
3. **Membership disclosure:** It means that through data linkage an attacker can determine whether or not data about an individual is contained in a data set [1].

These risks are measured based on the perception of an intruder, an entity who attempts to link a respondent to a microdata record or make attributions about particular population units

from aggregate data [5].

To minimise this risk to privacy Statistical Disclosure Control (SDC) techniques are applied on microdata sets to protect personal data while also maintaining its usefulness [36]. SDC refers to methods that try to prevent statistical data, such as microdata, from disclosing confidential information about specific respondents, who may be individuals or enterprises [36] while still maintaining some of its value [46]. Thus, in broad terms, SDC is a discipline that deals with reducing disclosure risk and information loss in a given data set.

Within SDC techniques, there are different methods to anonymize data sets. These methods can be categorised as perturbative and non-perturbative methods. Perturbative methods falsify the data before publication by introducing an element of error purposely for confidentiality reasons [46]. Non-perturbative methods reduce the amount of information or the level of detail by data suppression or other methods, but preserves truthfulness [37]. However, these methods do not specify any mechanism to assess what is the disclosure risk remaining in the transformed data set [90]. This drawback is addressed by privacy models which specify some properties that the data set must satisfy to limit disclosure risk, but they leave it open which SDC technique can be used to satisfy these properties [90]. Hence, the use of privacy models is appealing and quite popular among academics and statistics bureaus.

These methods are made available for use through software tools(referred throughout the text as SDC tools/applications) as functions that can be applied on privacy-sensitive microdata sets, that else would not have been disclosed. Therefore, the adoption of these tools is vital for public or private organizations across the world looking to open their data sets. There is an organizational need to understand and master these tools in order to anonymize data for the purpose of sharing it with the public or other organisations.

1.2. Problem Statement

There are quite a few SDC tools that are available, notably μ -Argus [46], sdcMicro [94], ARX [77], PARAT [7], Cornell Anonymization Toolkit (CAT) [2] and UTD Anonymization Toolbox [3], but their accessibility is not devoid of challenges. Majority of these tools are either underdeveloped or are not open to the public or were developed only for the purpose of demonstration. ARX, on the other hand, is an open-source software desktop application that undergoes continuous development. It supports a wide variety of privacy and risk models, methods for transforming data and for analysing the usefulness of output data [1] making it the preferred tool of choice for anonymizing microdata sets.

Despite the elegance and extensiveness of this tool, it is not widely used by data processors in public organisations in the Netherlands. The complexity of the use of the tool and the underlying expertise needed for understanding its functionalities is a likely factor that impacts the adoption of SDC methods in organizations, leading to sharing of data without appropriate protections. The complexity of this tool can be described in terms of the knowledge needed to use the tool effectively and its GUI design. Often SDC tools can only be used by experts, whereas a large-scale use by public organizations for opening data sets would require the use by non-experts. This is also supported by Prasser et al. who posit that the limitation of these tools relates to either scalability issues when handling large data sets, incomplete support of privacy criteria and methods of data transformation and/or, most significantly, these tools require complex configuration by Information Technology (IT) experts [77].

Moreover, the people who implement these techniques, especially statistical agencies, do not widely share, in substantial detail, their knowledge and experience using SDC and about the process of creating safe data with other agencies [20]. This makes it difficult for those

organisations who are motivated to implement this solution but are new to the process to get all the relevant information they need to apply these techniques in practice.

There are benefits to be gained by making these tools more accessible and easy to use for newcomers who are looking to delve into the field of SDC methods, especially for practical applications.

The problem statement can thus be summarised as microdata sets have to undergo anonymization methods before they can be opened to the public or shared with other organizations. This anonymization process can be done by using SDC software tools. But these tools are quite complex in their use, to begin with, especially for people within public organisations who are not SDC experts or researchers in this field. Thereby impeding the accessibility and adoption of such tools in practice. Figure 1.2 illustrates the problem statement.

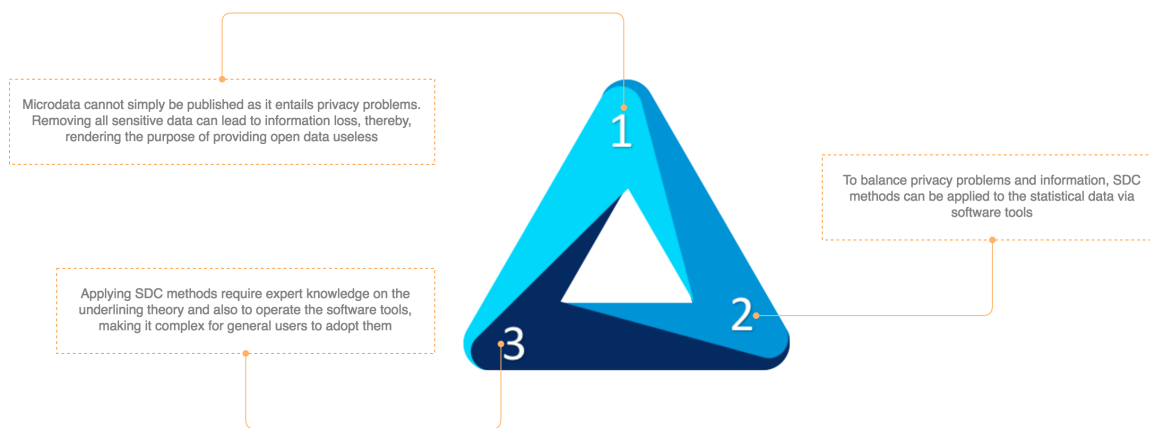


Figure 1.2: Problem Statement

1.3. Research Objective

As stated earlier, there is a need for SDC tools that are not developed for expert users, but can address the demands of anonymizing microdata sets while stripping away some of the complexity that these tools pose for non-expert users. Taking ARX as a reference, this complexity has to be first explored to arrive at a solution for the said problem.

It can likely be that the perceived complexity of this tool affects how this tool is used. If this tool is adopted in an environment where people are not familiar with concepts of data science and SDC methods, there is a possibility that the tool's potential might not be fully realised due to the user's perceived complexity of the tool. Meaning that the data transformation results would not be the best possible outcome in terms of controlling disclosure risk and maximising data utility. It could also be that the extensiveness of the functionalities provided by ARX especially concerning risk and utility measures might lead to low user adoption as it leads to choice overloading in users. Choice overloading can drive users to arrive at an inferior solution as people tend to do things, they are familiar with rather than exploring for better results [83]. Therefore, instead of taking the research approach of attenuating the user adoption process of ARX, it is deemed suitable to propose an alternative, not a replacement to ARX, but rather a simpler SDC tool. To make it easier for organizations or individuals to be introduced to the complexities of the SDC methods and their application.

Thus, the research objective of this master thesis is to analyse, design and evaluate a low fidelity SDC tool prototype that addresses the usability problems of ARX to increase

user adoption and reduce usage complexity. Developing a prototype that adheres to SDC guidelines while also providing a user-friendly, time-efficient and intuitive GUI which simplifies the process of data anonymization for its intended target user groups. Thereby, in the hopes of bridging the gap that is inherently created because of complex tools like ARX which are quite hard to use in practice.

The target user groups of this tool are defined as data controllers and data processors within the public organisations, and not experts or academics proficient in the theory of data anonymization.

The scope of this thesis also addresses the typical challenges in SDC tools, majorly usability problems in ARX, and to provide a solution for overcoming these problems. The thesis does not aim to enhance privacy methods or improve the functionalities of the said tool. The scope of this research only spans the pre-processing section, namely the *data harmonisation* step, in the standardised government data life-cycle [17]. This step involves only the preparation of the data to conform to the publishing standards established by the government [17] which is done through a software tool. The proposed tool is a standalone application software which can be used within an organization or used privately by individuals. It is not an integrated or mediating software to manage the business processes needed to realise the open data initiative of an organisation. And lastly, while the problem statement is observed universally the region of focus is narrowed down to EU member states.

Figure 1.3 shows the positioning of the thesis in line with the problem statement.

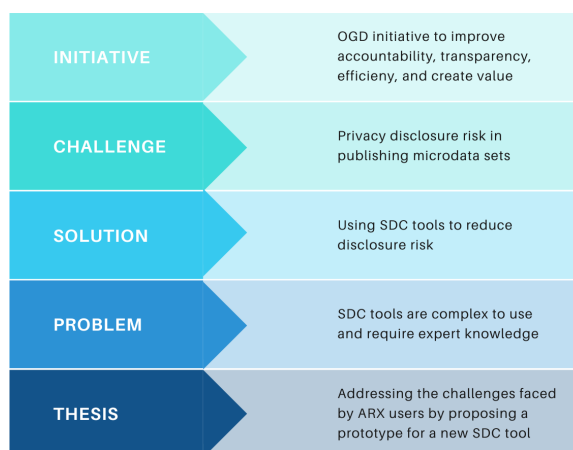


Figure 1.3: Research Objective

1.4. Research Questions

To achieve this, the main objective of this project can be realised by the following six research questions:

1. What are the typical challenges of using SDC tools?
2. What are the usability problems with ARX in practice?
3. What are the requirements to overcome these problems?
4. How are the requirements translated into a design?
5. How is the resulting design perceived by users of ARX?

1.5. Research Methodology

The nature of this thesis requires understanding the challenges users face while using SDC tools, namely ARX, and then overcoming them by designing a new prototype. A research methodology for answering the research questions is illustrated in Figure 1.4.

The first step is to identify the typical challenges users face while using SDC tools. This can be answered through a literature review. Next, the second research question can be answered by investigating the problems people face while using ARX. This can be achieved through qualitative and quantitative data (a mixed-methods approach) collection and then analysing the results. The answer for the third research question is a collective result from the first two research questions and requirement analysis through use case modelling. The fourth research question can be answered by developing a low fidelity SDC tool prototype. Lastly, the prototype can be evaluated through quantitative interviews to answer the last research question. The results of the evaluation can be used as a feedback loop to improve the prototype.

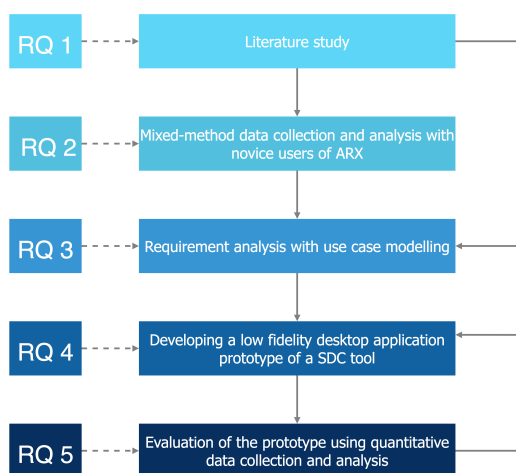


Figure 1.4: Research Design

1.6. Thesis Outline

Following the introduction in this chapter, the thesis is structured along these lines: Chapter 2 explores the literature needed to answer the first research question by reviewing topics on technology adoption models and software usability. Chapter 3 lays the foundation for answering the second research question through interviews to investigate the usability of ARX and the challenges for general users. In Chapter 4 said challenges and usability are addressed by defining the requirements for a prototype design of a new SDC tool. Chapter 5 explains the design of the prototype in detail and evaluates it against ARX. Chapter 6 reflects on the thesis by discussing different aspects of the research. Lastly, the thesis is concluded in chapter 7.

2

Literature Review

In this chapter, the literature relevant for this thesis is reviewed. Section 2.1 describes the methodology used for the literature review. This is followed by a stakeholder analysis in section 2.2 to understand the different actors at play. Section 2.3 provides an overview of adoption models. Next, Section 2.4 provides a discussion on effective UI design. This is followed by Section 2.5 which sheds light on how software usability can be measured. Lastly, the chapter is concluded in section 2.6 by answering the first research question and giving a brief description of how the literature will be used in the subsequent chapters.

2.1. Methodology

The starting point of this literature review stems from grey literature provided by WODC. They conducted a study on SDC technologies that can be used to enable the sharing of microdata sets to the public. Because of the extensiveness of this report, it gave an overview of the different topics involved in the theory surrounding SDC. Next, the general literature on SDC was searched to have an overview of not only the technologies but also the related concepts, models and practical applications of SDC. This step was again met with grey literature published by different organisations. This literature was filtered by scanning the table of contents and those materials were selected that provided a more rounded and well-structured outline. Moreover, a separate literature search was done on technology acceptance models, theories on user adoption, usability studies and lastly design guidelines for UIs.

2.2. Stakeholder Analysis

Using SDC software tools to realise the OGD initiative of the government is a confluence of many different stakeholders. At first glance, it can be assumed that the government and the respondents of the data are the primary actors involved in the initiative. However, there is more to this than meets the eye. Adopting SDC tools for anonymizing microdata to make it available for public release is an intricate project in itself. It is surrounded by many actors that can influence to impede or facilitate its progress.

Identifying these actors and locating their goals can help in understanding the levels of influence they can have to answer the problem statement of this thesis [23]. These stakeholders are identified and are analysed using the *Power/Interest Matrix* [66]. This matrix identifies stakeholder based on their ability to influence an organization's strategy or project resources and how interested they are in the organization or project. It allows grouping the stakeholders into four zones based on the relationship they hold with the project. This is

illustrated in Figure 2.1, followed by a description of each stakeholder.

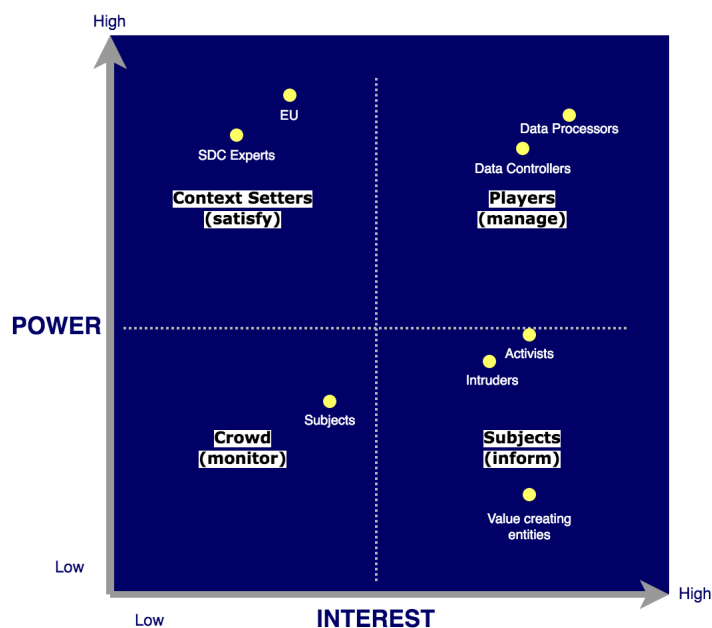


Figure 2.1: Stakeholder mapping on a Power/Interest Matrix

- **The European Union:** The EU plays a major role in the stakeholder analysis as it is responsible for laws like GDPR which directly influence the design and implementation of OGD initiatives. The EU laws define parameters for privacy and personal data and lay down constraints for publishing such records. They are the context setters and have high power. It is important to keep them satisfied.
- **Data Controller:** In this case, this term is defined as a public authority which determines the purpose and means of the processing of personal data. Here, public authority refers to the government, namely, the policymakers who are responsible for fulfilling the goals of the government. They decide which data should be made accessible, how it should be made accessible and to whom. Therefore, they are the key players in this scenario.
- **Data Processor:** This refers to those people who process personal data on behalf of the data controllers. In this context, data processors might be people working within the government or a separate public department that processes data for all other departments of the government. These are the people who will apply SDC techniques through the use of software tools for anonymizing microdata. They are also identified as key players but have lower power than Data controllers.
- **SDC Experts:** This refers to experts, developers and researcher of SDC techniques and tools. These actors can greatly influence how SDC methods and tools are used in practice. Through their knowledge, they have the potential to improve existing practices or render them obsolete in the future. Thereby changing the use of such tools. This makes them the context setters but they have lower power than the EU as they are not policymakers.
- **Data Subjects:** These actors are the respondents of the data. If data privacy and security of the respondents is not protected then they might not be willing to give data to the government in the future or falsify the data thereby impacting the utility of the data.

Release of sensitive information also impacts the accountability of the government. They have low power but may or may not have some interest in the project. They are part of a crowd and thus need to be managed.

- **Value creating-entities:** This can refer to individuals or organisations such as business entities or non-governmental organisations who can use the published data for creating value. Sometimes organisations have the resources to process the data which can be used for improving products and services, thereby adding to the economy of the nation. Therefore, they have high interest but directly no powers to influence, putting them in the zone of subjects that need to be kept informed.
- **Intruders:** Intruders or attackers can refer to an individual or a private organization. They are also receivers of the government data and it is undetermined how this data will be used by them. But there is a possibility of people with malicious intent to use the data to cause harm to the respondents of that data. Thus, they have high interest and slightly more power to influence the project than value-creating entities.
- **Activists:** This refers to civil rights, political or social activists that can disrupt the actions of the government if they feel that the government is violating their beliefs. For example, civil rights activists can question the government if personal data is not protected as it directly violates people's right to privacy and dignity. They have high interest and like intruders, they have slightly more powers of influence.

Data processors are relevant for this thesis work as they are the people who will directly deal with SDC tools and hence, would be required to learn and operate them. Their background and demographics can be important for finding the right candidates for the interview process during the research.

2.3. Overview of Adoption Models

Why users accept or reject a particular technology is a question that has received much attention from academics and professionals alike. Answering this question brings the prospect of predicting the impact of technology on human behaviour. Understanding this challenging issue can help the business organization develop successful products, giving them a competitive advantage over other organizations. Additionally, it can also help in comprehending the acceptability of SDC tools.

Several studies and approaches have been conducted by researchers on understanding this particular behaviour in humans. This has given rise to many theories and models on technology acceptance and adoptions. Figure 2.2 gives an overview of some of these theories/models. For the literature review, the focus will be on the most relevant and commonly used models. These are the Technology Acceptance Model (TAM) [31] and the Unified Theory of Acceptance and Use of Technology (UTAUT) [98].

2.3.1. TAM

TAM was first introduced by Fred Davis in 1985. It is an extension of another theoretical model the Theory of Reasoned Action (TRA) [31]. TRA posits that a person's behavioural intention is a function of the person's attitude towards performing the behaviour and normative beliefs that influence the individual's subjective norm about performing the behaviour [59]. However, TRA is considered to be "a general model as it does not specify the beliefs that are operative for a particular behaviour" [33, p. 983]. This gave rise to TAM which does not include TRA's subjective norm as a determinant of a person's behavioural intention (BI) [33].

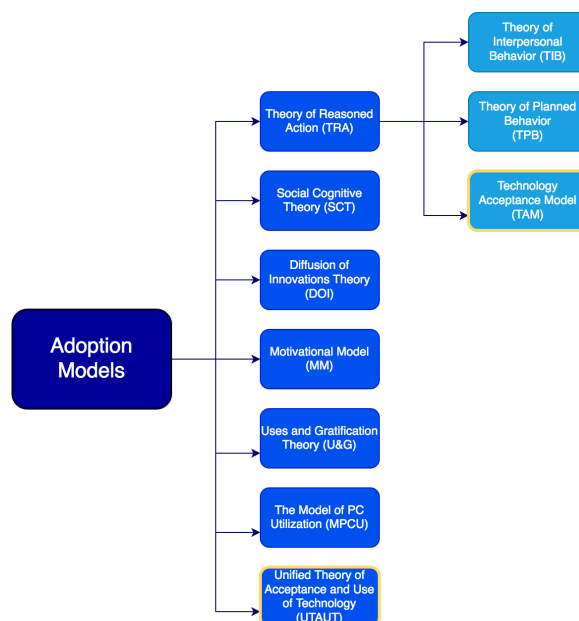


Figure 2.2: Overview of Adoption/Acceptance Models

TAM explains the motivations or behavioural intention of a person to use a technology based on three factors [31]. They are:

1. **Perceived Usefulness (U):** The degree to which whether or not a person perceives that a technology will be useful to enhance the person's performance.
2. **Perceived Ease of Use (E):** The degree to which a person believes that the technology will be easy to use. This factor has a positive effect on *U*.
3. **Attitude toward Use (A):** The person's attitude towards using a technology. Both *U* and *E* positive effect in the person's attitude toward using a particular technology.

TAM also differs from TRA in computing BI which has a positive effect on the actual system use. It is jointly determined by *A* and *U* [31]. Sometimes, factors called external variables are also considered in the TAM model. These external variables influence *U* and *E*. This is illustrated in Figure 2.3.

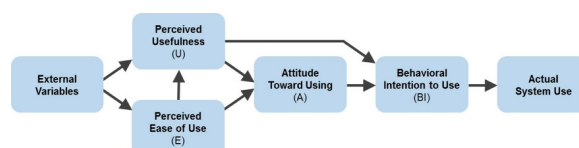


Figure 2.3: Technology Acceptance Model (TAM) [33]

The literature review revealed studies on critical external variables that have an impact on the adoption intention of users of software application technologies. Thong, Hong & Tam identified three categories of external variables in their research – Interface Characteristics, Organizational Context and Individual Differences [95]. The authors found that interface characteristics such as clarity in screen design and terminologies can have a positive effect on *E*. Individual differences like computer experience and domain knowledge were shown to also have a positive effect on *E*. Whereas, organizational context like system accessibility and

relevance were shown to have a positive effect on both U and E .

Another study conducted by Miller & Thomas focused on elements of human-computer interactions (HCI) for identifying behavioural issues [64]. In this paper, the authors categorised the external variables as System Characteristics (e.g., performance, on-line information) and Interface Characteristics (e.g., dialogue style, graphics). Their study also targeted users that were not computer professionals like programmers or system engineers but rather general users. They concluded their study by saying that the list of external variables was not exhaustive and that there were, in fact, more variables that can impact a user's behaviour to adopt a particular software application [64].

Though the TAM is widely accepted and has been the focus of more supporting research which has led to the development of its extensions. It has come under criticism over its practical application such as in a work environment. Critics of TAM suggest that to obtain a proper understanding of the factors which promote the use of a technology in a work environment, it is necessary to have a comprehensive theoretical and practical knowledge of the frameworks and models through which the use of that technology is investigated [61].

Ajibade explains that underlining behaviour cannot be reliably quantified in an empirical investigation, owing to several different subjective factors such as the norms and values of societies and personal attributes and personality traits [12]. Subjective norms like interpersonal influence through a friend can apply to personal use of a technology but may not be applicable in a work environment. Consequently, as an organization attains maturity it may establish policies and processes to use technology that is provided by the organization itself. Accordingly, critics suggest that behavioural expectations should be used instead of BI concerning the levels of compliance but not solely based on the perceptions of employees [61].

Some of these limitations of TAM are addressed in UTAUT which is explained in the next section.

2.3.2. UTAUT

UTAUT is a unified model that integrates eight other acceptance models to determine user acceptance of IT [98]. Much like TAM UTAUT identified constructs and variable that influenced BI and subsequent usage. The authors identified four key constructs [98]. They are:

1. **Performance Expectancy:** The degree to which a person believes that using the application will enhance their job performance.
2. **Effort Expectancy:** The degree of ease associated with the use of the application.
3. **Social Influence:** The degree to which the user believes it is important that others believe the user should use the application.
4. **Facilitating Conditions:** The degree to which a person believes that technical and organizational infrastructure exists to support the application.

These four constructs are further influenced by moderating variables such as gender, age, experience and voluntariness of use [98]. The relationship between the constructs and variable on BI can be understood in Figure 2.4.

UTAUT gives a better measure of the factors involved in understanding why some technologies are accepted by users and some are not. The constructs can be measures to further analyse the impact on the BI. Whereas, the moderating variables are an interesting addition to the

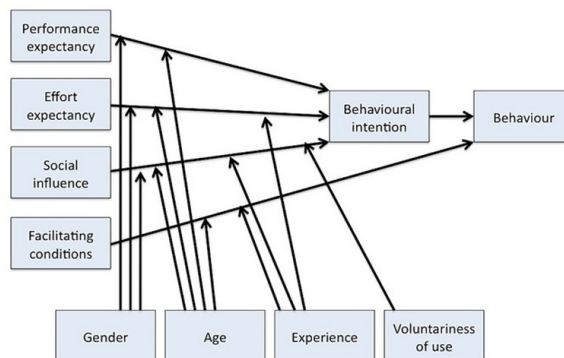


Figure 2.4: Unified Theory of Acceptance and Use of Technology (UTAUT) [98]

model that influence the constructs in turn [93]. Overall UTAUT helps in understanding how the user demographics and role of an organisation can influence user adoption behaviour.

However, UTAUT is also not devoid of criticism. One interesting criticism of UTAUT is using moderators to cause high predictability in other factors is unnecessary in practical applications [55]. The author suggests that a good predictive power can be achieved even with simpler models in organizational research design under practical business settings [55].

TAM and UTAUT both give a holistic approach to the factors that influence user adoption behaviours which can be critical especially in introducing new application technologies in an organisational workplace environment. These factors can be considered while preparing questions for the interview process in the later chapters.

2.4. User Interface Design

Designing a UI for a software often involves a considerable investment of time and effort which can be reduced by adhering to previously established design guidelines [89]. These guidelines can serve as a starting point for establishing software requirements for developing a prototype.

Smith and Mosier provide a very extensive report on UI guidelines in six functional areas: data entry, data display, sequence control, user guidance, data transmission and data protection [89]. The authors highlight that many times guidelines make rudimentary references to UI design, with general statements like 'the system should be easy to use' [89]. They add that this leads to the absence of effective guidance, in both the design and implementation of UI software and thus, becomes the responsibility of programmers who are unfamiliar with operational requirements [89]. This results in the detection and correction of design flaws that occur only after the prototype has been developed and making software changes becomes difficult. To overcome these problems the authors provide detailed guidelines with examples on a range of topics identified under the six identified functional areas. These guidelines are meticulously detailed and summarising them would be out of scope for this review. Table 2.1 lists the number of guidelines provided under each functional area.

However, not all guidelines can be applied to the design and have to be filtered for tailoring the design to fit the needs of the target group. Broadly speaking, most guidelines deal with HCI which has a fundamental objective to make systems usable, useful, and to provide user experience (UX) fitting their specific background knowledge and objectives [42].

A technique that can be used for capturing and describing the functional requirements of a

Section	Functional Area	Number of Guidelines
1	Data Entry	199
2	Data Display	298
3	Sequence Control	184
4	User Guidance	110
5	Data Transmission	83
6	Data Protection	70

Table 2.1: Guidelines organized under six functional areas of user-system interaction [89]

software tool is use-case modelling [14]. Use-cases describe all those scenarios in which a user can interact with a system [80]. Writing effective use-cases can help in realising the goals of the different stakeholders. It can also lead to stakeholder driven requirement analysis taking into account the possibility of conflicting requirements [25].

Moreover, Morris & Dillon argue in their paper that developers can gather inputs on user perception of the usefulness or ease of use of the system based on preliminary designs of software tools [65]. The paper suggests that these early formulations of user perception of a system influence whether users will actually use that system [65]. The literature also suggests that capturing predictive measures of user acceptance, even before the user has an opportunity to interact with the software, can lead to correlations between perceived usefulness and eventual user adoption of the software [32].

Additionally, Fischer observes that "the challenge in an information-rich world is not only to make information available to people at any time, at any place, and in any form, but specifically, to say the *right* thing at the *right* time in the *right* way" [42, p. 65]. Complex software systems usually cater to this notion by providing its users with more options. However, Iyengar and Lepper challenge the implicit assumption that having more choices is necessarily more intrinsically motivating than having fewer choices [47]. Termed as the 'choice paradox' studies have shown that choice overloading in economic applications leads the user to decision fatigue which in turn deteriorates the quality of decisions made by the user [75]. Moreover, having too many choices can draw the attention of the user away from the main content, making it difficult for the user to focus on just one piece of the displayed content [75].

Additionally, sometimes software fails to get adopted by the user despite been designed by adhering to guidelines and analysing the user needs. Gould & Lewis suggest that the different components of a software – operating environment, user platform and reference manuals or materials usually fail to interact cohesively to create a conception that the user eventually deals with, as these components are designed separately [45]. The UI is not the lone component in the environment the user uses the system, it is part of a much bigger picture. Constantine and Lockwood purpose use-case modelling where the attention is on the usage rather than the user, specifically to implement UIs that are complex or extensive in design [28]. They believe that such an approach can enhance the UX of the users and focus on the bigger picture.

In short, UI design guidelines can only point you in the right direction, defining requirements and assessing how a system will be used by the user is equally important.

2.5. Measuring Software Usability

To understand the issues users face while using SDC tools it is imperative to explore the literature on software usability. Understanding what is software usability and how it is measured can shed light on the usability of SDC tools and whether they fair well when their usability is measured.

Software usability does not have a consistent definition. It has been defined in different ways depending on the literature. One article suggests that usability relates to how a system interacts with the user and cannot be defined as a specific aspect of a system [41]. Regardless of how it is defined, measuring usability usually helps in determining the quality of the system.

Many techniques and models have been proposed to measure usability. One such model is called Quality in Use Integrated Measurement (QUIM). This model unified existing models into a single consolidated, hierarchical model of usability measurement [84]. The QUIM model proposes 10 factors on which the usability of the software can be assessed. Each of these 10 factors corresponds to a specific facet of usability that has been previously identified in an existing conceptual model (e.g., Metrics for Usability Standards in Computing [MUSiC]) or standard (e.g., ISO 9241, ISO/IEC 9126, IEEE Std.610.12) [84, p. 168-169]. The factors are:

1. **Efficiency:** Capability of the software to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use.
2. **Effectiveness:** Capability of the software to enable users to achieve specified tasks with accuracy and completeness.
3. **Productivity:** The amount of useful output that is obtained from user interaction with the software.
4. **Satisfaction:** Subjective responses from users about their feelings when using the software.
5. **Learnability:** Capability of the software to enable users to feel that they can productively use it right away and then quickly learn any subsequently new functionalities.
6. **Safety:** Capability of the software to meet the user requirements during normal operation without harm to other resources and the environment.
7. **Trustfulness:** The faithfulness a software offers to its users.
8. **Accessibility:** Capability of a software to be used by persons with some type of disability.
9. **Universality:** Capability of a software to accommodate a diversity of users with different cultural backgrounds.
10. **Usefulness:** Capability of a software to enable users to solve real problems in an acceptable way.

These factors are then further decomposed into 26 measurable usability criteria which is a metric through which the factors can be measured. The description of the usability criteria and their relationship with the factors can be found in Figure 2.5 and Figure 2.6 respectively.

On observing Figure 2.6, some measurable criteria affect more than one usability factor. For example, 'minimal memory load' affects six out of ten factors. This criterion can be considered to be significant as having this capability in a software can greatly affect its quality and usability through multiple factors. In a practical environment, it may not be possible for a software to satisfy all the criteria as there are resource restrictions when developing a software. In such a case, depending on the context, the measurable criteria can be prioritised and only those can be included that have the maximum impact on the usability factors.

Practical applications of QUIM suggest to measure the 26 criteria on a Likert scale and calculate the response by mapping the results according to the effect the criteria has on enhancing the usability factors. The developers of QUIM also propose the model to have practical applications not only for measuring quality standards of a system but also to serve as a guide for incorporating usability into a software design [84].

Criteria	Description
Time behavior	Capability to consume appropriate task time when performing its function
Resource utilization	Capability to consume appropriate amounts and types of resources when the software performs its function (ISO/IEC 9126-1, 2001)
Attractiveness	Capability of the software product to be attractive to the user (e.g., through use of color or graphic design; ISO/IEC 9126-1, 2001)
Likeability	User's perceptions, feelings, and opinions of the product (Rubin, 1994)
Flexibility	Whether the user interface of the software product can be tailored to suit users' personal preferences
Minimal action	Capability of the software product to help users achieve their tasks in a minimum number of steps
Minimal memory load	Whether a user is required to keep minimal amount of information in mind in order to achieve a specified task (Lin et al., 1997)
Operability	Amount of effort necessary to operate and control a software product
User guidance	Whether the user interface provides context-sensitive help and meaningful feedback when errors occur
Consistency	Degree of uniformity among elements of user interface and whether they offer meaningful metaphors to users
self-descriptiveness	Capability of the software product to convey its purpose and give clear user assistance in its operation
Feedback	Responsiveness of the software product to user inputs or events in a meaningful way
Accuracy	Capability to provide correct results or effects (ISO/IEC 9126-1, 2001)
Completeness	Whether a user can complete a specified task
Fault tolerance	Capability of the software product to maintain a specified level of performance in cases of software faults or of infringement of its specified interface (ISO/IEC 9126-1, 2001)
Resource safety	Whether resources (including people) are handled properly without any hazard
Readability	Ease with which visual content (e.g., text dialogs) can be understood
Controllability	Whether users feel that they are in control of the software product
Navigability	Whether users can move around in the application in an efficient way
Simplicity	Whether extraneous elements are eliminated from the user interface without significant information loss
Privacy	Whether users' personal information is appropriately protected
Security	Capability of the software product to protect information and data so that unauthorized persons or systems cannot read or modify them and authorized persons or systems are not denied access (ISO/IEC 12207, 1995)
Insurance	Liability of the software product vendors in case of fraudulent use of users' personal information
Familiarity	Whether the user interface offers recognizable elements and interactions that can be understood by the user
Load time	Time required for a Web page to load (i.e., how fast it responds to the user)
Appropriateness	Whether visual metaphors in the user interface are meaningful

Figure 2.5: Measurable criteria in QUIM [84]

The QUIM model can be used to measure the usability of ARX and those usability criteria that

Criteria	Factors									
	Efficiency	Effectiveness	Satisfaction	Productivity	Learnability	Safety	Trustfulness	Accessibility	Universality	Usefulness
Time behavior	+			+						
Resource utilization	+			+						+
Attractiveness			+						+	
Likeability			+							
Flexibility		+	+					+	+	+
Minimal action	+		+		+			+		
Minimal memory load	+		+		+			+	+	+
Operability	+		+				+	+		+
User guidance			+		+			+	+	
Consistency		+			+	+		+	+	
Self-descriptiveness					+		+	+	+	
Feedback	+	+							+	+
Accuracy		+				+				+
Completeness		+				+				
Fault-tolerance						+	+			+
Resource safety						+				
Readability								+	+	
Controllability							+	+	+	+
Navigability	+	+					+	+	+	
Simplicity					+			+	+	
Privacy							+		+	+
Security						+	+			+
Insurance						+	+			
Familiarity					+		+			
Loading time	+			+					+	+

Figure 2.6: Relationship between factors and criteria in QUIM [84]

measure low can subsequently be incorporated into the design of the prototype.

2.6. Conclusion

The review followed a stakeholder analysis to understand the involvement of different actors associated with the OGD initiative, focusing more on the use of SDC methods for opening data sets. Then, the literature needed to answer the first research question was reviewed.

The literature revealed that there could many challenges that a user could face while using SDC technology. User acceptance models indicate that the challenges could be related to factors that motivate a person to use a technology. TAM model suggests that users are motivated to adopt a technology if they perceive the technology is useful and is easy to use. Moreover, UTAUT goes further than TAM and defines these factors as constructs and adds that social influence and other facilitating conditions like technical infrastructure also influence user adoption behaviours. These constructs are further influenced by user demographic and skill levels. While this is a good estimate of where adoption problems could lie, it could be that the usability of a technology itself poses problems for the users.

The QUIM model provides a way to measure usability and rate the quality of the product. The challenges that users face could range from whether they find the technology attractive to whether they can complete a specific task using the technology. Moreover, problems could lie in the inherent design of the technology that hampers its usability. If guidelines are not followed while designing a software and keeping user needs in mind then the resulting software design could be dismissed by the users.

In conclusion, these could be some of the typical challenges that users can face while using SDC tools.

Further Use of the Literature

The literature is applied at different phases of the thesis. In chapter 3, ARX is investigated by structuring the qualitative interviews based on the insights gained from adoption models. QUIM model is used to assess the usability of a specific application in chapter 3 and chapter 5. Also, the stakeholder analysis is used to help in participant selection for the research. User interface design guidelines and concept of choice overloading are referred to in Chapter 4 where the requirements of the proposed prototype are identified. Additionally, to understand the user needs, use-case modelling is applied. Lastly, the literature is reflected on in the concluding chapter of the thesis.

3

Investigating Usability of ARX

This chapter details the steps taken to answer the second research question by conducting a mixed-methods research approach. Since, the focus of the thesis is concerned with understanding the problems users face while adopting ARX, the application itself is first described in section 3.1. and how the tool is viewed by its developers. Followed by section 3.2 which explains the purpose of conducting such a research. Section 3.3 elaborates on the interview process such as the method, content, selection of interviewees and method of analysis. Next, section 3.4 shows the results of the data collection and analyses. The results are further interpreted in this section. The chapter concludes with Section 3.5 where the second research question is answered.

3.1. ARX

ARX is an open-source data anonymization tool that was developed mainly by Fabian Prasser and Florian Kohlmayer to use in the domain of health data privacy [1]. The tool undergoes continuous development, testing and documentation updates from its contributors.

It supports most privacy models, data transformation models and data quality models [1]. It also provides additional features such as creating data transformation rules, analysing risk and utility [1]. These features are made accessible through a GUI and also through a java software library that contains APIs (Application Programming Interface) which provide access to all features implemented in ARX. The ARX GUI can be seen in Figure 3.1.

Due to its active development, it has been transformed into a flexible tool that can be extended to support almost all arbitrary combinations of a wide range of techniques in a salable manner [78]. The developers supported this by conducting an extensive experimental comparison to show how ARX outperforms related solutions in terms of scalability and output data quality while being compatible with a broad range of techniques [78].

The developers of ARX believe that its flexibility and a relatively intuitive, easy-to-use interface are the key factors that contribute to the software's success [78]. This has helped ARX to find its way into official policies and guidelines of European Union Agency for Network and Information Security (ENISA) and UK Anonymization Network (UKAN), numerous research projects, and data publishing activities that have made use of the software [78].

However, they also emphasise that the methods implemented by the software are complex from a mathematical and statistical perspective and, as a consequence, anonymization in

real-world settings can usually only be carried out by experts [78]. They posit that different anonymization techniques should be applied depending on the context and that one must be aware of the intended use of the data set to ensure data remains useful and is reliably protected [78].

Moreover, they highlight that despite ARX's flexibility, it can anonymize only medium-sized data sets with up to a few million rows [78]. In a practical application, data controllers often deal with gigabytes and terabytes of data [38]. This is one challenge the developers of ARX hope to address in their future work.

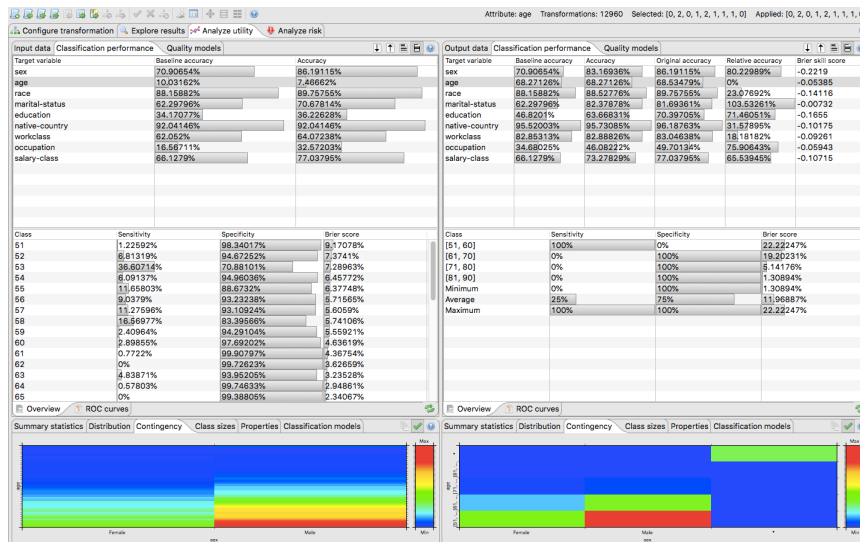


Figure 3.1: Graphical User Interface of ARX

3.2. Purpose

The purpose of the interviews is twofold. First, there is a need to understand the way users interact with ARX. Focusing on the user experience can help us understand the user itself, their needs, what they value, their abilities and their limitations [44]. It can also lead to understanding how the user perceives the product [44]. These insights are critical and can be used to improve the user experience.

Second, it is possible that we may uncover challenges that ARX itself poses for its usability. Such challenges greatly reduce the quality of the product itself and might also limit its usability. Understanding such challenges can help us to improve on its overall design.

3.3. Interview Setup

3.3.1. Interview Method and Content

The data collection process followed a mixed-methods approach. It is argued that mixed-methods research is one of the three major “research paradigms” i.e quantitative research, qualitative research, and mixed-methods research [50]. Such an approach involves more than one form of data collection. It uses both qualitative data collection methods and quantitative data collection methods. A reason for selecting such an approach is that mixed-methods strategy is also associated with method triangulation. Method triangulation is a technique in which confidence in the results can be increased if using multiple methods of data collection and analysis leads to the same results [85]. Another reason is that the studies on the usability of SDC tools have never been conducted before. Therefore, it is possible

that one type of data collection method might not be enough to answer the proposed research questions. Moreover, a software such as ARX has a very small user base. It is not widely used and hence, it is not often a topic of discussion or scrutiny. Using a mixed-methods approach might help in further explaining the results of the initial study [29]. The mixed-methods strategy adopted for this thesis puts more emphasis on qualitative data collection.

For data collection, interviews were conducted one-on-one through a video conferencing application (Skype). The duration of the interviews lasted for about 30 to 45 minutes. The structure of the interview process was divided into two parts – qualitative data collection and quantitative data collection.

The first part concentrated upon collecting qualitative data. The participants were asked a series of open-ended questions to facilitate a discussion like atmosphere. The questions ranged from basic to more specific topics, loosely based on the QUIM model, and took about 15 to 20 minutes of the interview time. Basic questions were asked to ease the participants into the discussion. These questions were simple and warranted only brief answers which helped in setting the pace of the interview for the interviewees as well as the interviewer. Whereas, more specific questions related to their experience with ARX application itself. The interviewees were also asked follow up questions when they related their feelings about a particular topic. They were asked to elaborate on these feelings in hopes of making an abstract subject more tangible. This proved useful during the data analysis phase as it was easier to identify similar themes and patterns that were observed in different participants.

The participants were also asked to do a heuristic evaluation of the ARX application. A heuristic evaluation is one of the methods included under usability inspection where a UI design is evaluated for usability problems in a cost-effective way [68]. During a heuristic evaluation, participants are asked to comment on the UI [70]. This method allows for finding major usability problems in a cheap and intuitive way especially when conducted individually with more than a single participant [67]. Other advantages of such an evaluation also include that it does not require planning in advance and it is easy to motivate people to carry out such an evaluation [70]. In this case, participants were asked to open the ARX application on their respective systems and evaluate it by navigating to its different aspects and elements. The time allotted for this evaluation was 5 to 10 minutes

The second part of the interview consisted of a quantitative data collection method to confirm the data collected in the previous steps. This included an online survey based on the QUIM model identified in the literature review. To measure the usability of ARX, only 18 out of the 26 usability criteria were considered. The reason for doing this was that some of the criteria mentioned did not apply to ARX as it is a desktop application whereas QUIM is designed to measure the quality of all kinds of software products (e.g, Operating Systems, Web Applications). Therefore, criteria like 'insurance' and 'resource safety' could not be considered. Other criteria like 'time behaviour' and 'resource utilization' that were related to optimising software performance were not considered as they were not within the scope of this thesis. The relevant 18 usability criteria were presented on a 5 point Likert scale to measure the level of agreement or disagreement the respondents of the survey had for the statements. The time required to complete the survey was estimated to be not more than 5 minutes.

The transcripts of the interview process involving qualitative data collection can be found in Appendix A. For quantitative data collection involving the online survey refer to Appendix B.

3.3.2. Interviewee Selection

A selection criterion was established for shortlisting the participants for the interview process. Since the research focuses on addressing the problem of the complexity of SDC tools in practice which makes it hard for public organisations to adopt them. The stakeholder analysis, conducted in the previous chapter helped in identifying the actual users of these tools which are the data processors. These users are not experts but rather employees who would be expected to have some basic knowledge or learn on the go to get the task done.

Thus, the interview participants were selected based on their levels of proficiency with ARX. These users can be categorised as entry-level users. They do not have in-depth knowledge or ample hands-on experience with using SDC tools and are thus non-experts in this particular field. However, they are cognizant of the theories involved in data anonymization to a beginner or medium level. They can, therefore, use ARX to anonymize a data set and analyse the impact it has on the data utility and risk.

The participants were also selected on their varying technical skill levels with an age range from 20 to 45 years. Participants also had different levels of academic background (e.g., Bachelor's, Master's, PhD's) and study fields (e.g Business IT, Computer Science). This was done to create a difference in the perspectives of how they view the ARX application. Triangulation also requires that the research is addressed from multiple perspectives [85]. Additionally, it was important to see if the same kind of challenges were faced by everyone despite the variation in their technical knowledge.

Moreover, taking a note from technology adoption models that were explored in the previous chapter, it was considered important to try and emulate real workplace settings. The participants selected were not users of ARX of their own volition but had been introduced to it because they had to use it to either pass a university course or conduct academic research which involved using this tool. Just like in a practical setting, an organization imposes the technologies its employees have to use to achieve the collective goal of the organization.

It was also deemed important to select the right number of participants for the research. Six & Macefield in their article compare different arguments put forth by researchers to determine the right number of participants needed to conduct a study for problem discovery concerning UIs [87]. They summarise these findings by concluding that a typical problem discovery study requires between three and twenty participants, with five to ten being a good baseline [87]. Similarly, in heuristic evaluation, it is recommended to carry out the study between three to five participants as the results grow rapidly but reach the point of diminishing returns around the point of ten participants [70].

Thus, bearing these points in mind, a candidate list was prepared. These candidates were known people who had used ARX and were acquainted with the internship supervisor at WODC. Eleven people were approached via emails (See Appendix G, 1) out of which 5 replied and agreed to be part of the research. A few participants were also asked to circulate the online survey among their trusted peers who were known to have used ARX to increase the response rate.

Hence, the mixed-methods data collection research was carried out with five participants. Two additional resources participated in only the quantitative data collection process. Thus, seven participants took the online survey.

3.3.3. Method of Analysis

Since this is a mixed-methods research approach, to analyse the results a convergent parallel design is used, refer Figure 3.2. It is the most widely used analytical approach [72]. In this approach, data analysis consists of merging qualitative and quantitative data which are collected concurrently and then the two sets of data and results are compared [52]. As stated earlier, this is done to exemplify the use of triangulation for drawing inferences from both sets of data.

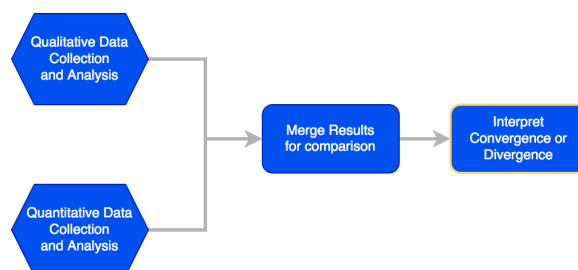


Figure 3.2: Convergent Parallel Design

Qualitative data analysis is done by following three steps – data reduction, data display and drawing conclusions [85]. First, data reduction is done using a streamlined codes-to-theory model [81]. The model follows a step-by-step process from data i.e the interview transcripts being coded to then being combined into groups with similar attributes through axial coding [91]. The groups are called categories which are then combined into themes and then further abstracted into theories, assertions or theoretical propositions. This model is found to yield richer results when used in a mixed-methods research [27].

For categorizing the data, the codes and categories are developed deductively first and then inductively. Miles & Huberman support this by saying that when there exists some preliminary theory, one can construct an initial list of codes and categories from it, and, change or refine these during the research process as new codes and categories emerge inductively [63]. Therefore, the codes and categories are generated from the literature reviewed in the previous chapter. The benefit of the adoption of existing codes and categories is that one can build on and/or expand prevailing knowledge [85]. This approach is followed in the thesis. Next, the data is displayed in the form of a comprehensible table. Lastly, the data is interpreted and conclusions are drawn.

Next, quantitative data analysis of Likert scale has been a topic of much debate. Some researchers argue that the Likert scale is ordinal in nature, while others treat it as an interval scale [85]. Usually, a Likert scale is analysed at the interval measurement scale as it allows researchers to calculate composite scores like mean for central tendency and standard deviations (SD) for variability [21]. This is the approach that is best suited for this thesis as it helps in a better triangulation of the results. Hence, for quantitative data analysis mean scores are calculated for each question asked and then their SD is taken to calculate data dispersion. Lastly, the results are displayed on a diverging stacked bar chart which centres the neutral responses in the middle, making it easier to compare the different categories of the responses.

3.4. Interview Results

3.4.1. Qualitative

The qualitative data collected from the interviews were first transcribed into transcripts from the audio recordings. A total of approximately 6500 words were transcribed. These transcripts

are included in Appendix A. Codes were generated for each interview transcript. On studying the transcribed data, it was evident that there were some recurring themes in the content. Similar themes were noted down so that they could be grouped under a specific code. Coding for each interview can be found in Appendix C. Next, the codes were classified and grouped under a category which is displayed in Table 3.1.

It is important to specify here, that similar codes were identified for multiple instances but in different contexts and not how many times it was repeated in the same context. For example, the code 'limited information' was identified multiple times when the participant spoke about visual GUI elements like buttons but also for textual elements like user manuals/documentation of ARX.

Code	Category
Prior knowledge needed Guidance required	Minimal memory load
Limited information Not instructive Minimal explanation Not self-explanatory	Self-descriptiveness
Inconsistent	Consistent elements
Confusing Non-uniform structure Switching between tabs	Navigability
Cluttered Outdated Unattractive	GUI design
Effortful	Operability
Advanced language Cannot interpret data Language not helpful	Data display (textual)
Easy visualisation	Data display (visual)
Discouragement Difficult to use Robustness	Feelings invoked

Table 3.1: Results of qualitative data analysis

The data analysis revealed that almost all participants shared a similar set of challenges and experiences that they encountered while learning to use and operate ARX. First commenting on the physical aspects of the GUI, four out of five participants were of the opinion that the GUI was outdated or old. They described it as *"something from the 90s and that it has not been updated in a very long time"*. They also felt the GUI had a cluttered design which is not evident in most modern GUIs.

On further asking them how they felt about such a GUI they revealed it invoked feelings of discouragement to work with the tool or felt the tool would be difficult to work with as they associated old looking GUIs to having a low quality user experience. For example, one participant said *"any software which looks attractive, I immediately feel that it is fine or a nice tool to work with. But my first impression with ARX was that this is going to be difficult and a little bit outdated"*. This can also be explained by technology acceptance models where a person's motivation to adopt a technology is influenced by the degree to which he/she believes the technology is easy to use. Moreover, two participants also commented on the inconsistencies in the design elements of the GUI such as, text fields that seemed editable but were, in fact, not editable.

Moreover, all five participants said that prior knowledge or guidance was needed to operate the tool. They felt that the tool had many functionalities but they had to look them up simultaneously in order to use the tool. They commented that they felt the tool was developed for experts where the developers had assumed the user to have theoretical knowledge to operate the tool. To quote, *"the reason why most of the information is not given on the GUI because it is assumed the user knows about the parameters and their definitions"*.

A lack of self-descriptive capability of the GUI was also observed in all five interviews. Participants felt that they had to constantly look up terms or functions that were displayed on the screen. They mentioned that the screens had a lot of textual content but not much information to understand it. They gave examples of buttons with no description or explanation of functions that were explained in a single sentence which was not helpful. They also commented that the interface lacked navigability leading the user to feel confused or to constantly switch between different tabs without any sense of structure. Moreover, they commented that the design was not instructive in nature which added to them having to learn the tool by hit and trial approach. This can also explain the high instances of 'minimal memory load'. If the users had to look up every function and commit it to memory then they would feel that they are overloaded with information. Similarly, the GUI was not instructive and did not guide them to the next step of the anonymization process, meaning the participants had to remember the steps they had taken for completing a task.

In addition to this, participants commented that ARX had a lot of textual content. One would normally think that more text would lead to more clarity. However, this was not the case. Participants felt that they could not interpret the textual data because either the language used was too advanced or that it was too minimal for them to get any useful information from it. For example, one participant said *"the manual had some difficult words in it which you had to know beforehand to use ARX"*. This also adds to the assumption that the participants made earlier that ARX might have been designed for a target user group consisting of people well versed in data anonymization techniques. Interestingly, data that was visually represented as gauge charts (speedometer chart), in 'Analyze Risk' tab, was found to be easier to interpret. Two participants shed light on this by commenting that they did not have a strong background in statistics but when data was presented to them in a creative way such as the gauge chart they were able to easily analyse the risk. They also commented that the data provided to assess the attacker model risks were clear and concise, meaning that they were not cluttered with too many numerical values.

Lastly, an observation that was not made during the interviews was whether the participants trusted the tool. One participant did comment on its robustness by saying that the functionalities in ARX are supported by scientific documentation. This can be explained by the demographic of the participants. An academic researcher can make such a claim as he may have read scientific documentation on the tool before using it and thus can make an estimation on whether the results of the tool are accurate. But most users are general users who are not academics, therefore, they rely on the tool or its user base to invoke feelings of trust. That's why most software applications now incorporate trust in their design elements [26].

3.4.2. Quantitative

The results of the quantitative data analysis in the Figure 3.3. are represented using a diverging stacked bar chart. At first glance it can be observed that all seven participants of the survey disagreed that ARX provides user guidance and requires minimal memory load for its use. Numerical proof for this data is displayed in Table 3.2. where the mean, variance and

SD are calculated for each question asked in the survey and taking the response of all seven participants. Refer Appendix D for individual responses to the survey.

The average mean calculated of all the responses gives the overall score to each usability criterion. For example, criterion minimal action has a mean score of 2.14, comparing this score on the values of the Likert scale it can be observed that 2.14 falls between 2 (=Disagree) and 3 (=Neutral), but closer to 2. With relatively a low SD value of 0.408 the data can be interpreted as participants disagreed that the software failed to possess the capability of minimal action.

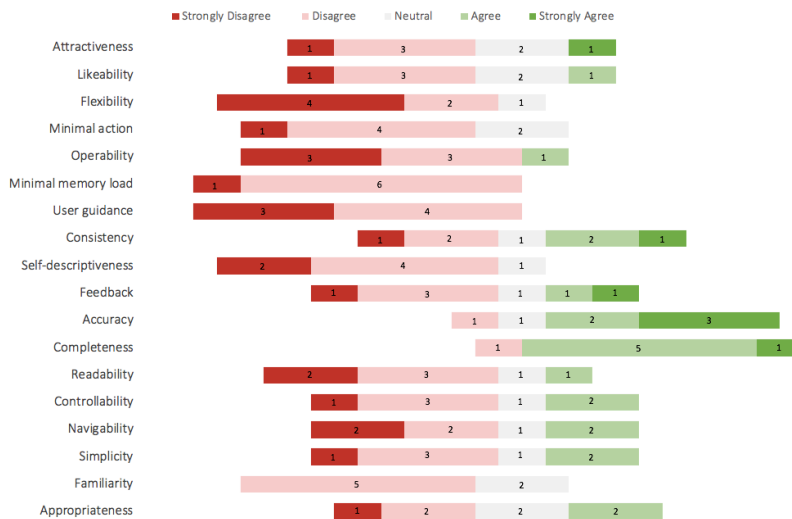


Figure 3.3: Graphical representation of the quantitative results

Strongly Disagree = 1, Disagree = 2, Neutral =3, Agree = 4, Strongly Agree = 5 *to 3 decimal places, sample size = 7				
S. No	Usability Criterion	Mean*	Variance*	SD*
1	Attractiveness	2.571	1.387	1.178
2	Likeability	2.428	0.816	0.903
3	Flexibility	1.571	0.530	0.728
4	Minimal action	2.14	0.408	0.638
5	Operability	1.857	0.979	0.989
6	Minimal memory load	1.857	0.122	0.349
7	User guidance	1.571	0.244	0.494
8	Consistency	3	1.714	1.309
9	Self-descriptiveness	1.857	0.408	0.638
10	Feedback	2.714	1.632	1.277
11	Accuracy	4	1.142	1.069
12	Completeness	3.857	0.693	0.832
13	Readability	2.142	0.979	0.989
14	Controllability	2.571	1.102	1.049
15	Navigability	2.428	1.387	1.178
16	Simplicity	2.571	1.102	1.049
17	Familiarity	2.285	0.204	0.451
18	Appropriateness	2.714	1.061	1.030

Table 3.2: Results of quantitative data analysis

Similarly, other measurable criteria for which the user disagreed on its usability with a low SD from the mean value are minimal memory load, user guidance, familiarity, self descriptiveness, minimal action and flexibility. The participants felt that ARX was rated low in usability for all these criteria.

Measures like attractiveness, likeability and readability have a high dispersion which could be a result of individual preferences of the participants. Moreover, the data suggests that the user agreed that ARX rated high when it came to the accuracy but this is followed by a high SD. This can be attributed to the varying knowledge levels of the users and hence, could not correctly judge the accuracy of their results. Similarly, consistency is also followed by a high SD and it could be a result of one or two odd cases where the participants felt the elements of ARX were not consistent in their design.

3.4.3. QUAL + QUAN

On merging the qualitative data analysis (QUAL) and quantitative data analysis (QUAN) it was found that some results were convergent. See Figure 3.4 for overlapping results.

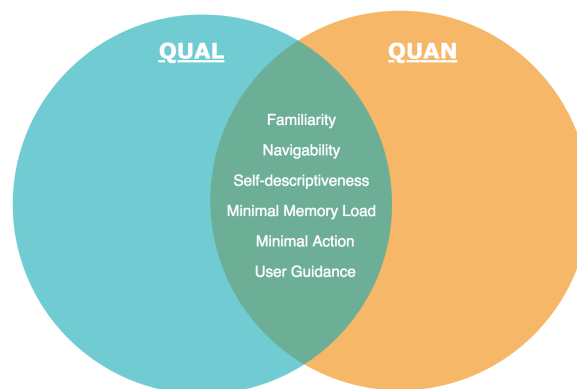


Figure 3.4: Convergence of qualitative and quantitative findings

In both the analysis the participants strongly agreed that ARX rated low on criterion concerned with minimal memory load, self descriptiveness and navigability.

Findings that were observed in both data set but with divergent results are consistency and attractiveness. In the QUAL a few participant commented on consistency between the GUI elements of ARX, however, this is not clearly supported in QUAN. No conclusions can be drawn from the latter analysis for consistency as the mean value indicated that people responded neutrally while rating it. It was also followed by a high SD and hence, consistency should not be considered in the analysis. Similarly, for attractiveness, in QUAL the participants agreed that the GUI was not attractive whereas in QUAN no firm conclusions can be drawn as attractiveness is rated to be neutral with a high SD.

Whereas, for findings that are observed in QUAN and are rated low by participants like user guidance, minimal action and familiarity can be explained in QUAN. User guidance can be explained in relation to self-descriptiveness as participants commented that ARX was not instructive in nature and needed guidance to operate. Similarly, minimal action and familiarity are interconnected and can be sub features or a by product of minimal memory load. If the user is required to keep very less information in memory while operating the system then a by product of this could be that the user has become familiar with the elements of the system and hence does not need to memorize each element and it's functionality. Consequently, the user will be able to perform a task in minimal number of steps.

Lastly, it is also important to comment on how data is presented to the users to ease some of the complexity of ARX. In QUAL, the participants were able to interpret the data if it was displayed clearly and concisely. These findings can partially be explained by the criterion readability in QUAN and can explain the relatively high SD. Readability cannot be considered

as a specific problem area of ARX as it takes into account many other features and how they are presented on the UI (e.g., file menu). Clear Data visualisation, on the other hand, can be a feature of self-descriptiveness and minimal memory load on how ARX uses language or any medium to express itself to the user without requiring the user to expend energy in understanding it.

3.5. Conclusion

Using a mixed-methods approach the challenges users face while using ARX were investigated. While some similar findings were observed in both types of data analysis, some findings were supported in one but were not supported in the other. In conclusion, the analysis helped in answering the second research questions by identifying six problem areas where the usability of ARX deteriorated. It is understood that some of these problems areas are interdependent on each other. One problem might be perceived as an enabler for another problem. The identified problem areas are summarized below:

- *Minimal Memory Load*: ARX requires its users to recall from memory concepts of data anonymization to complete a task using the application. Thereby, increasing the burden on memory rather than promoting recognition.
- *Self-descriptiveness*: ARX system lacks self-explanatory features. This is compounded by a lack of external supporting documentation.
- *User Guidance*: Low self-deceptiveness results in poor user guidance.
- *Navigability*: The design elements of ARX impede a smooth navigation experience for the user and adds to more confusion.
- *Minimal Action*: Lack of information and guidance leads to users finishing a task in more number of steps than actually intended.
- *Familiarity*: Given the extensiveness of ARX's features, its design such as content display does not invoke feelings of familiarity in the user, resulting in the user to constantly having to look up things.

ARX tries to provide a range of functionalities to anonymize microdata. This is what sets it apart from its peers. However, this is also its drawback. Without a large user base that can provide more supporting documentation and usability problems weighing down its potential, the users of ARX cannot fully maximise its value as a robust tool. Users are left to apply only those functionalities that they can easily comprehend and thus, this can lead to inferior anonymization results. Therefore, there is a need to simplify these functionalities.

4

Prototype Requirements

In this chapter, the process of designing the prototype is described. Section 4.1 details the software requirements needed for developing the prototype by drawing support from the results of the previous chapter. This section is divided into two subsections to categorize the different subsets of the software requirements. Next, section 4.2 defines the user interaction with the systems through use case modelling. Lastly, the chapter is concluded in section 4.3 summarising the requirements for the prototype to answer the third research question.

4.1. Software Requirements

Before designing any software application, the needs of the users, as well as their expectations from the application, have to be defined. These are called software requirements. Software requirements are the description of functions and features that a software system must provide and the constraints under which it must operate [8].

Specifying these requirements is an important part of any project. It is a mechanism by which one can identify, organize and justify the software requirement goals of all the stakeholders involved [15]. Moreover, McDermott & Shimeall stress upon specifying software requirements as it can contribute to the success of the project [62]. Therefore, for designing the prototype it is necessary to identify the requirements it will fulfil to meet the needs of the stakeholders and reduce the risk of failure.

In the previous chapter, ARX was evaluated and it was concluded that there are certain aspects of the tool which pose a problem for general users. These problems, leave plenty of room for improvement and have to be addressed. Thus, they contribute to a significant portion of the software requirements and are categorized as user requirements.

Additionally, the stakeholders require a tool that can help them achieve their OGD initiative. The tool should have the capability of anonymizing microdata while providing a means to balance information loss and the risk of disclosure. Such requirements are categorized as functional requirements and constitute the remaining portion of the software requirements.

4.1.1. User Requirements

User requirements are functional requirements that are developed from a user's perspective [100]. In the previous chapter, the usability problems with ARX were explored and the results of the evaluation explained to some extent why general users find the tool complex to use.

The identified problem area's of ARX are addressed here which form the rationale for defining the user requirements.

Referring to software guidelines that were identified in the literature review, the problems can be solved by applying certain software design rules. These rules are summarised in Table 4.1, followed by a detailed explanation.

Problem Area	User requirement
Minimal Memory Load	Minimalistic design to avoid visual clutter
	Consistent interface elements based on existing mental models
	Offloading tasks by using default values or visual clues for decision making
Self-descriptiveness	Intrinsic methods to relay information
	Use of simple, unassuming language
	Providing contextual functions and information
	Instinctive placing of visual metaphors
User Guidance	Principle of tunnelling and selective attention through multi-step pathway forms with inline validation for task completion
Navigability	Defining a clear primary navigation area
	Minimal hierarchical structures that embrace predictability such as a left-hand side navigation menu
Minimal Action	Streamlining and grouping similar task actions on one page/tab of the screen
Familiarity	Incorporating predictable design elements in pace with current trends

Table 4.1: User requirement solutions for addressing the six usability problems of ARX

Minimal Memory Load

Reducing memory load means that users should not be required to memorize or recall out of their heads a great deal of information to carry out tasks, this can, in turn, greatly enhance the user's performance [84]. According to the QUIM model, minimal memory load has an impact on efficiency, satisfaction, learnability, universality, accessibility and usefulness of a software [84]. Thus, it can be regarded as an important usability criterion to have in software design.

During the evaluation of ARX, participants felt that the tool was developed for experts. The developers of ARX are also of the belief that since the software deals with mathematical and statistical models, anonymization in a real-world setting can usually only be carried out by experts [78]. Therefore, the design of ARX does not take into consideration an in-depth description of the functionalities or a detailed explanation of the steps needed to be done to anonymize a data set as it is unintentionally assumed that the target user group consists of experts well versed in SDC techniques.

As a consequence, ARX provides a multitude of functionalities that can be used alone or in combination with other functionalities to anonymize and analyse a given data set. This can inevitably be a lot to comprehend especially for a non-expert user resulting in them recalling information and looking up explanations for features at each step of the data transformation process. Thereby, leading to an increase in memory load and conversely to the usability of the tool.

However, memory load can be reduced by eliminating some functionalities and following certain design guidelines, some of which are explained here which will be incorporated in the prototype:

1. Striving for an aesthetic and minimalist design can avoid visual clutter [18]. If a design

has irrelevant images or typography then it can slow a user down [99]. Similarly, overuse of meaningful data in the form of textual or visual aids can decrease the performance of the users. The users will spend much time focusing on different elements on the screen to understand their meaning, thereby, increasing their cognitive load. Instead, ensuring only relevant and necessary information is displayed on each component (e.g., modal windows, messages boxes) per page [86].

2. Designing the interface based on existing mental models people have about how a desktop application works, based on their past experiences with other desktop applications [99]. When users are presented with similar design elements that they would normally see in other commonly used applications it reduces the amount of learning they need to perform basic functions. Additionally, ensuring consistency between different elements of the GUI to not confuse the user whether a set of elements function in a similar manner or not [18]. This applies to textual and visual elements as well.
3. Offloading tasks is another way of reducing memory load [99]. Elements that require the user to input data or make decisions can be re-displayed as default values or visual clues. This can eliminate certain mental tasks for the user. Similarly, it is good to provide a mechanism for editing data item as close as possible to its display, also known as direct editing [10].

Self-descriptiveness

Self-descriptiveness of a software can come in the form of extrinsic methods such as user manuals and supporting software documentation (e.g., developer website) and intrinsic methods in which the software explains itself through its design alone [43].

General users of ARX found it difficult to understand the elements of the ARX interface because of inadequate explanations provided in the manual or its instant help feature. Explanations consisted mostly of generic single line sentences. They felt that sometimes the explanation only provided the definition but not what it meant in the scenario in which it was used or appeared. Given the small user base of ARX there weren't many application-related examples of ARX that the users could refer to and hence, they depended mostly on the tool itself for explanations. Also, given that the users did not understand the different implications of the statistics that were displayed, it was hard for them to analyse the results of the anonymized data.

To address this user requirement, the approach of intrinsic methods to increase self-descriptiveness will be adopted in the prototype. Self-descriptiveness can be improved by first ensuring clarity in data display. Complex functions should be explained in simple words, assuming the reader to be an entry-level user. The language in which the information of a function is displayed should be within the context in which that function appears. For example, the function of analysing utility should be explained by not just defining what data utility is but how it fits in the context of the data anonymization process. These explanations, depending on how extensive they have to be, can be made available on request either by hovering over the element or by providing a search feature.

Additionally, data presentation can also be done using visual metaphors to engage users instinctively with complex data [30]. For example, using bullet charts can help in category comparisons.

User Guidance

User guidance is much needed when the functionality of the software is too complex for general users to comprehend. Which is the case in ARX. The ARX UI does not provide any subtle clues

to its users to guide them through how its many functionalities can be used holistically. The first screen that comes up on opening the tool does not indicate to what needs to be done (see Figure 4.1). The user has to hover over the displayed buttons to see their description and then decide on an action.

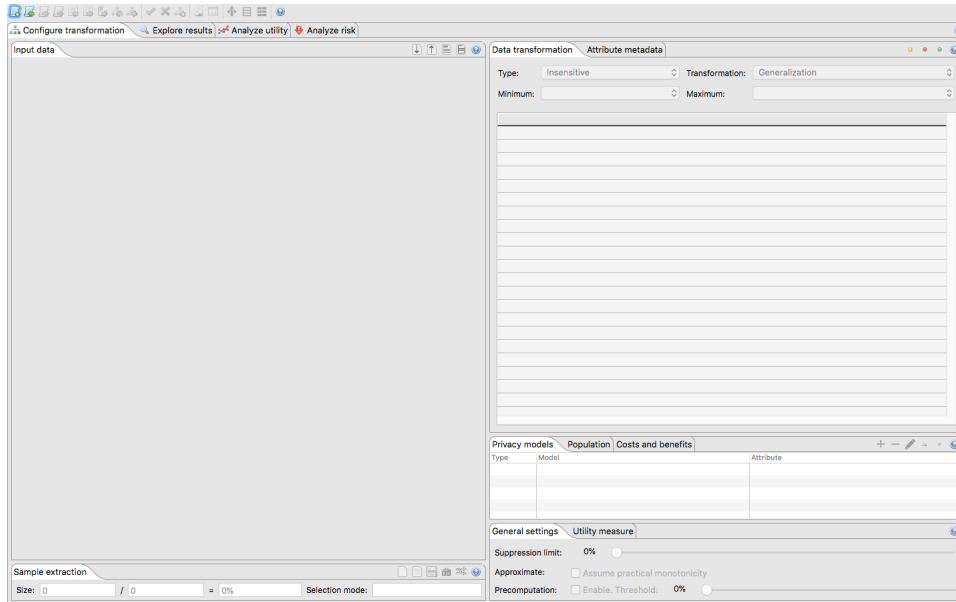


Figure 4.1: Screen on startup in ARX

Another characteristic of ARX that adds to its complexity is how its functionalities are displayed to the user. Almost immediately, without even uploading the data, different buttons and tabs are made available to the user. Whether these buttons/tabs need a prior set of action(s) to be completed in order to use them is not made explicit. They are enabled and thus, the users can click them. Such a display of functionalities can confuse a user or overwhelm them with choice. Logical steps required to perform a data transformation can be lost underneath an overload of functionalities of the tool, leading the user astray from its intended use.

Therefore, much like self-descriptiveness user guidance can be enhanced in a UI by providing clear instructions to users for guiding them to the next step. This can be done by using the principle of tunnelling [51]. It is a concept for guiding the user from point A to point B without any distractions from unnecessary elements on the screen [51]. This concept plays on the attribute of selective attention of the user and is especially helpful for designs where many steps or actions are needed to be taken. In the prototype, tunnelling can be implemented through the use of multi-step forms. These type of forms act as pathways for the user to complete one task from start to finish. In this case, the task of configuring all settings for data transformation can be done using this. Additionally, with the use of inline validation, the user can be notified of the completion of one step and can be prompted to proceed to the next one.

Navigability

Similar to user guidance, ARX's layout and structure impedes a seamless navigation experience. Its multiple tabs along with tabs within tabs layout (see Figure 4.2) leads to an increased number of user clicks, hand and eye movements. Easy navigation can eliminate some of the confusion users engage in when they open an application for the first time. It can also quickly make them accustomed to the layout so that they can anticipate where to click

next.

Navigation guidelines suggest defining clear and minimal hierarchical structures that embrace predictability [76]. Unconventional designs, albeit creative, can decrease the quality of the user experience. The guidelines suggest locating a primary navigation area in a noticeable place, preferably adjacent to the main body of the screen from where it can provide feedback on the user's location [69]. For the prototype, a left-hand side navigation menu is a fitting choice especially for desktop applications and also for displaying the location of the user without the need for additional clicking or hovering.

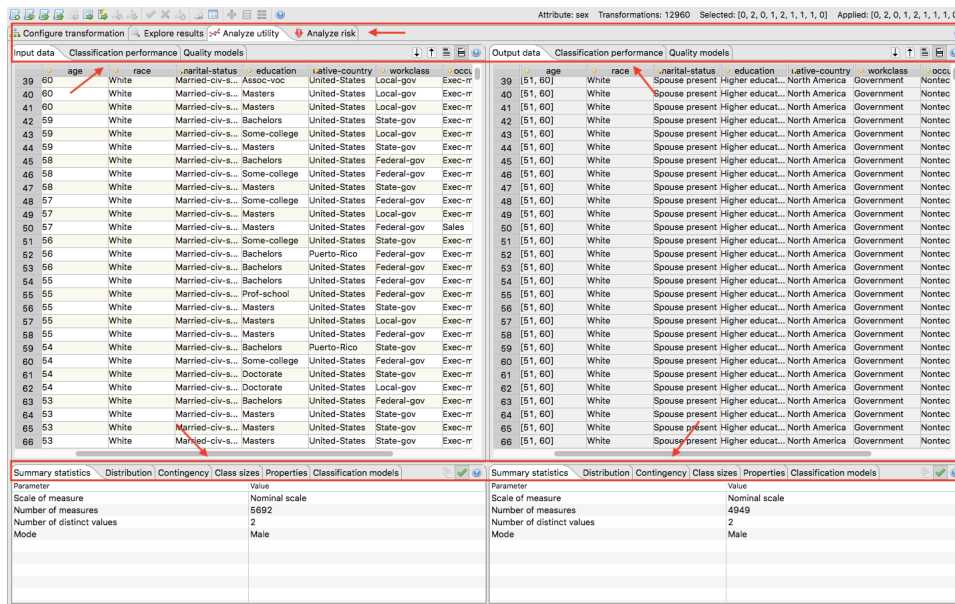


Figure 4.2: Multiple tabs in ARX

Minimal Action

Actions that require users to click, scroll or drag the mouse multiple times for doing a single task have to be minimised. As stated previously, ARX's has many tabs within tabs which increase user actions. In addition to this, attribute mapping in ARX which is a mandatory step for data transformation is designed quite inefficiently. The user has to first select a column from the data set and then navigate across the screen to configure its individual properties. This is highly inefficient if the data set has too many columns and the user has to horizontally scroll through them one by one each time. Moreover, there is no visual indication when a specific column is configured. In case a user fails to configure even one column then the only way to check this is by clicking on each column one by one or when an error is thrown by the application on proceeding to anonymize a data set.

Minimal actions can be done by streamlining and grouping similar task actions on one page/tab of the screen. For the prototype, the configuration for doing data anonymization can be grouped under one tab and the task for analysing the output data can be grouped under another page/tab.

Familiarity

As stated in the earlier chapter that familiarity and minimal action are interconnected. An unfamiliar design can lead to more user actions as the users undergo a hit-and-trial approach to understand the application. In the case of ARX, an unfamiliar design does not help its inherent

complexity. For example, users commented on the use of 'knob-like' buttons (see Figure 4.3) in ARX which had to be rotated in order to adjust a value. The use of such elements was not intuitive to them and hence, instead of a click and drag action to rotate the knob they proceeded to double click it. Thus, familiarity is another user requirement that will be incorporated into the prototype.

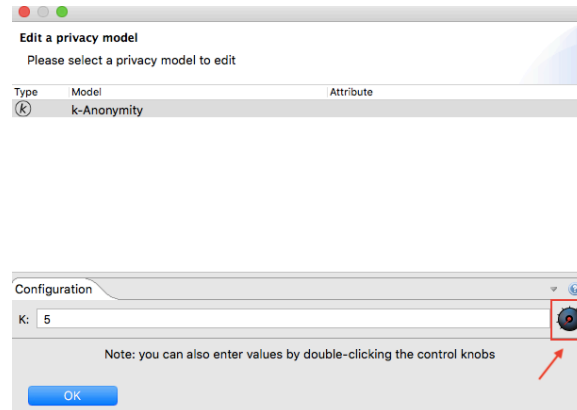


Figure 4.3: Knob button in ARX

Similar to minimal action, familiarity can be increased by a structured design. It can also be introduced by improving navigability by incorporating predictable design elements. The look and feel of a UI that is more in pace with the current trends in commonly used applications can enhance familiarity. As there is a possibility that the target users might also be users of these commonly used applications. Hence, a modern UI is a suitable design for the prototype.

4.1.2. Functional Requirements

Functional requirements play an important role in defining a product from the perspectives of both the customers and the developers to capture specific design requirements [96]. They are a set of domain-related functionalities of an application and are implemented by a subsystem or a group of components which are traceable in the architecture of the application [22].

During the investigation of the usability of ARX, it was surmised that the functionalities of ARX have to be simplified as an overload of options does not necessarily lead to better results. This can be achieved by reducing the domain-related functionalities. Providing users with fewer options can result in them making better-informed decisions without facing decision fatigue, as observed in the literature review.

Thus, by using relevant literature on SDC techniques, the functional requirements for the prototype are streamlined into three categories which are specified below.

Anonymization Approach

To define the functional requirements it is important to know the approaches in which data anonymization is done. There are two ways to anonymize data using SDC techniques. First, the traditional approach in which SDC method with a heuristic parameter choice and with suitable utility preservation properties is run on the data and, after that, the risk of disclosure is measured [60]. Second, an alternative approach which is also provided by ARX is based on the notion of a privacy model. A privacy model is a condition, dependent on a parameter, that guarantees an upper bound on the risk of re-identification disclosure and risk of attribute disclosure by an intruder [60]. Since ARX is the subject of analysis for this thesis the anonymization approach are privacy models.

For designing the prototype, an example of one privacy model is taken and functions are defined by tracing the steps needed to perform a data transformation. Also, the goodness of the data is assessed in terms of utility and risk applying only that privacy model. The privacy model selected to illustrate this is k-anonymity.

K-anonymity is a popular and much-developed privacy model used for anonymizing microdata sets. The main principle of this model is that the information attributes of each person contained in a data set cannot be distinguished from at least $k-1$ individuals whose information also appears in data set [92]. This is similar to hiding a person in a crowd. Here, the size of the crowd is determined by the value k . This model focuses on quasi-identifiers (QI), which are not direct identifiers (e.g., name, citizenship number), but are those attributes which in combination might be used to link the anonymized data with an external identified data source to re-identify the individual to whom an anonymized record corresponds to [60].

Subsequent research has shown that k-anonymity is susceptible to attribute disclosure risk where an attacker can discover the values of sensitive attributes (e.g., ethnicity, political affiliation), that are neither quasi-identifiers nor direct identifiers [58]. This has given rise to the development of extensions of k-anonymity that protect against attribute disclosure risk, such as l-diversity [58], t-closeness [56], etc. Hence along with k-anonymity, l-diversity will also be used as part of the example. The reason for doing this is to develop the prototype as close to a practical application in terms of providing an adequate solution for anonymizing a data set.

Additionally, other privacy models exist that are essentially different from the family of k-anonymity privacy models. Namely, ϵ -differential privacy that defines privacy standards differently than k-anonymity models [39]. This definition of privacy overcomes the limitations of k-anonymity models and also provides an unconditional privacy guarantee by not making any assumptions about the attacker's strategies [73]. However, this is a relatively new concept and requires addressing some practical challenges before it can be deployed in a real-world scenario.

It should be bought to the notice of the reader, that the academic community is divided in their support for one reigning privacy models. Each model has its limitations. And this thesis does not in any way tries to settle this debate by preferring one model over the other. It should be understood that as of now there is no near-perfect solution for anonymizing data sets and it is quite possible that in the future, a new way of anonymizing data sets may be developed that gives better results than existing models. Therefore the thesis focuses more on making privacy tools usable so that it can be adopted by a wider user base. Additionally, designing for scalability to accommodate different types of privacy method, making the prototype more flexible in design. Thus, for the sake of simplification, the thesis uses the process of k-anonymity with l-diversity as an example to illustrate the functional aspects of the prototype.

Data Utility and Risk Analyses

Next, the number of measures that a user can avail to analyse data utility and risk are defined. ARX provides several options for measuring data utility and risk of the transformed data.

Data utility measures are indicators for assessing the usefulness of the data. For microdata sets, there are two categories of data utility measures, the so-called special-purpose measures and general-purpose measures, depending on whether or not the usage of data is already known, respectively [13]. General-purpose metrics are selected for designing the prototype as they are more useful for open data initiatives where data publishers do not know how

data recipients are going to use and analyse the published data [13]. Average Equivalence Class Size, Non-Uniform Entropy, and Granularity are three general-purpose metrics that are incorporated in the functional requirements.

For analysing risk, the risk measures considered are three popular risk evaluation metrics which are based on the prosecutor, journalist, and marketer attacker models [74]. In ARX, these models are used to depict records at risk and the rate of success of the attacker as graphical representations. As a result, few participants had commented on the ease with which they could assess the risk using attacker models because of how the data was displayed. Hence, attacker models will be selected as another functional requirement that the prototype will possess.

General Anonymization Configurations

The functional requirements also consider providing an option for setting some general configurations that are applicable to all privacy models. Currently, ARX provides around 4-5 configurations of which some are set to the default recommended setting. Whereas, other data anonymization tools do not even provide this functionality. It is assumed that these configurations are set by default by the back-end system and are not part of the user process i.e the front-end.

The prototype design follows the ARX approach but provides only a few configurations such as 'suppression limit' to the user. On displaying the configurations, these setting will be set to the default (i.e 100% suppression) value but the user will have the ability to edit this. This functionality allows the user to set a limit on the maximum number of records that can be removed from the input data set [1]. It is a preferred configuration to have as it can significantly reduce loss of information [53].

Global Requirements

The prototype will also support some other common operations that can be defined as global requirements. A list of these requirements are given below:

1. The user should be able to open a new project or an existing project in the application.
2. The user should be able to save a project.
3. The user should be able to import the input data into the applications. The supporting formats for the data include CSV, Excel (XLS, XLSX) and Database (JDBC) files.
4. The user should be able to export the transformed data. The supporting formats for the data include CSV, Excel (XLS, XLSX) and Database (JDBC) files.
5. The user should be able to close, minimise or maximise the application.
6. The user should be able to filter raw data to remove direct identifiers from the data set through an edit function.
7. The user should be provided hints or information on SDC related terminologies on the screen through a pop-up while hovering on a particular term.
8. The user should be able to disable or enable the hint option from the settings.
9. The user should be able to use a search feature. This feature will search within the documentation of the application and related theories of the functionality provided by it.

Functional Workflow

In figure 4.4, the workflow involved in anonymizing a data set using k-anonymity is described.

The dotted box divides the process by differentiating between steps that will be common for all

other privacy models and steps that are specific only for k-anonymity. As mentioned earlier, this is done to make the prototype more flexible in design. Flexible arrangement of simplified operations can help in the extensibility of the software and integration with other operations with the software [79].

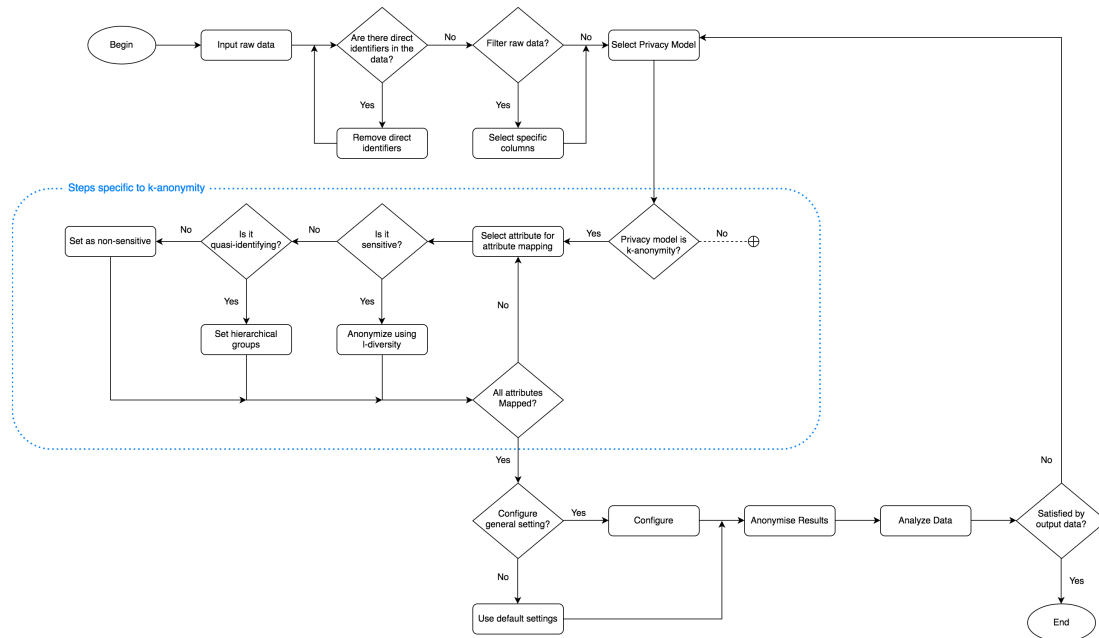


Figure 4.4: Functional steps involved in k-anonymity privacy model

4.2. Use Case

Once the requirements for the prototype are gathered, it is vital to identify and define the different processes of the prototype. Use-case modelling is the technique identified during the literature review to capture those scenarios in which a user interacts with a system [80].

A general use case is defined in Figure 4.5, in which a user first creates a project in the prototype application. Next, the user uploads the microdata file that is required to be anonymized. On uploading the microdata file the user can edit this file to remove certain columns such as direct identifiers from the data set. Then the user can configure settings required to anonymize the file. The user can select the desired privacy model. Additionally, the user can perform attribute mappings and apply general privacy settings which are applicable for all privacy models. On configuring the settings the user can now apply it to reveal the transformed data set. The transformed data set can then be analysed based on measures for assessing data utility and risk of disclosure. Lastly, the user can download the anonymized microdata file for distribution.

This is a general use case, however, in a real scenario the user will probably go back and forth between changing the settings till the desired level of anonymity is achieved.

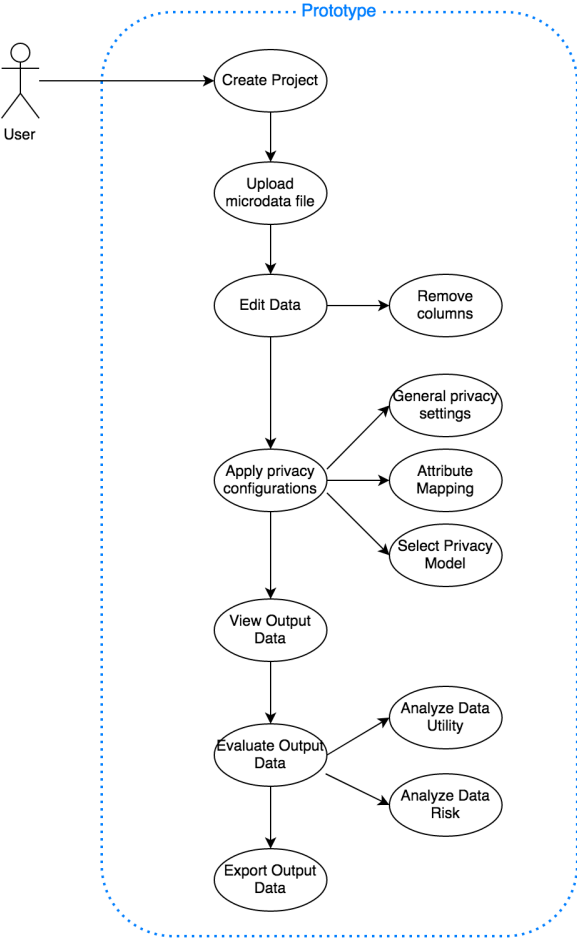


Figure 4.5: Use case for anonymizing a data set

4.3. Conclusion

In this chapter, software requirements were defined to address the complexity of ARX. The requirements were divided into two categories. First, user requirements were defined that directly tackled the problems users face while using ARX. Then, within the context of SDC guidelines domain-specific functional requirements were identified in the form of anonymization approach, data utility and risk analysis measures, and general configurations. These functionalities were simplified and streamlined to reduce choice overloading. Additionally, some global requirements were defined to support some user customary functions in desktop applications. Lastly, through use-case modelling system behaviour was delineated.

5

Prototype

In this chapter, based on the requirements identified earlier a prototype is developed and then evaluated. Thereby, addressing the last two research questions.

The fourth research question is answered in the first two sections. Section 5.1 provides a general description of the prototype followed by section 5.2 which goes into detail to explain the individual pages of the prototype and how the user interacts with it. Then, the evaluation is conducted in section 5.3 to answer the final research question. This section first explains the purpose, setup and method of the evaluation. Followed by the results and its implications. Finally, the chapter is concluded in section 5.4.

5.1. General Description

The prototype is given the name Danaamta which is a culmination of two words – anaam and data. *Anaam* is a Hindi word for 'one without a name' and is merged inside the word *data* to form *d-anaam-ta*, signifying anonymity in a crowd of data. The logo which is a masquerade mask also depicts this line of thought.

The prototype is given a modern outlook by taking examples of commonly used software to promote familiarity. A minimalist look with a dark colour scheme are some of the attributes of recent UI designs. Gradients and subtle drop shadows are used to draw user attention to interactive elements.

The UI is divided into a static and dynamic section. A left-hand side menu is the primary navigation area which remains static. And placed adjacent to this, is the dynamic main body which takes up most of the screen to display context-sensitive content based on the menu option selected.

Figure 5.1 offers a sitemap to indicate the structure of the prototype and all the pages contained within it, along with major functionalities. The pages represent context-sensitive content which dynamically load on the main body of the prototype depending on the menu option selected. The description of the individual pages is provided in the next section. The working prototype can be accessed through [this link](#).

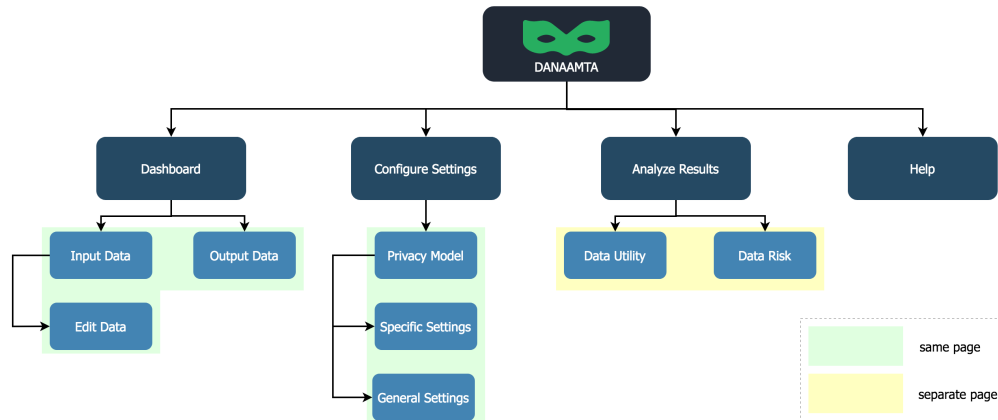


Figure 5.1: Sitemap of Danaamta

5.2. Page Description

5.2.1. Dashboard

The dashboard is the first page that is displayed to the user on opening the prototype. On initial startup, this page displays a dialogue box prompting the user to create a new project or open an existing one (refer Appendix E, Figure A.1). The user cannot proceed ahead without selecting one of the given options as the other menu options are disabled and there is no way to close this dialogue box. However, the user can click on the menu options 'help' to access the user manual or 'exit' to close the application. The user can also access these options from 'file'.

If the user selects the first option then the current dialogue box is replaced by a new one prompting the user to upload a microdata file (refer Appendix E, Figure A.2). If the user opts to open an existing project then the current settings or state of that project is loaded. This statement will hold more meaning later on when all the pages in the prototype are explained and the states of the prototype involved in the data transformation process are defined.

Upon uploading the data file it is displayed dynamically on the dashboard under the section input data (refer Appendix E, Figure A.3). The user can scroll through the data on this page. Additionally, an 'edit' option is provided by which the user can delete some columns in the data file. The purpose of this is to vet the data by removing direct identifiers or any other data which is not required in the final data set.

Additionally, the menu option 'configure settings' is enabled and is now accessible by the user. This is an example of tunnelling and is adopted throughout the workflow of the prototype.

5.2.2. Configure settings

This page provides configurations needed to be done to anonymize the data set. On clicking it, settings are displayed on the right-hand section of the dynamic body.

As per the functional requirements defined in the previous chapter, the anonymization approach taken here is of privacy model. Therefore, the user is first prompted to select a privacy model. This section is numbered and shown to be the first step in a series of steps that will result in enabling the 'apply settings' button. Refer Appendix E, Figure A.4.

On selecting the desired privacy model from the drop-down, additional configurations specific to that privacy model are then displayed (refer Appendix E, Figure A.5). K-anonymity is used as an example to simulate this.

Recall in ARX, attribute mapping is a seemingly tedious task for the user. In the prototype, this functionality is addressed to reduce minimal user action. Here, the columns are dynamically identified and displayed as drag-able elements. By default, all the columns are placed under the non-sensitive section, hence, reducing some user actions by partly mapping attributes as non-sensitive that would eventually have been set as non-sensitive attributes. This is not done in ARX.

In the prototype, the user can drag the column name under the appropriate group – 'sensitive' or 'quasi'. Such a view reduces some user actions as the list of columns needed to be configured is generated by the prototype itself without requiring the user to scroll through the data as is the case in ARX.

Through the use of coloured asterisk marks the user is also notified of additional configurations for the attributes. For sensitive attributes, the user has the option to further anonymize them by using other privacy models like I-diversity to control the risk of attribute disclosure. For quasi-identifiers, the row data can be optimised by creating hierarchies which further abstracts the column data.

It is taken into consideration to display additional settings within the context of the content to show the connection between different functionalities. For example, in the case of ARX creating hierarchies is an option placed on the top toolbar along with import and export data. A first time user would not be able to make the connection of optimizing the quasi-identifiers as this option is placed so far off from its context.

Additionally, the last section provides settings which are generic and can be applied for any privacy model. By default, these settings are configured to the recommended value but are still editable. This is done to reduce memory load by offloading some user tasks.

Therefore, through a list-like display, the user is prompted to configure at least the mandatory settings which will enable the 'apply settings' button resulting in an anonymized data set.

5.2.3. Analyse Results

Once the settings are applied, the user is taken back to the dashboard where the anonymized data is loaded under the 'output data' section. The user can perform a side by side comparison of the input and output data. Refer Appendix E, Figure A.6.

In addition to this, the 'analyse results' menu option is enabled. The user can click this to reveal a drop-down list with the sub-menu option of either analysing data utility or disclosure risk.

Both these pages (refer Appendix E, Figure A.7 and A.8) use terminologies which might be unfamiliar to the user. Using the hover action a description of the terms is made available to the user through a message box. The language used to describe the terms is simple, non-generic and unbiased to not influence the user's decision when analysing the output data.

Different visualization techniques are used to analyse the results. In addition, colours are used to convey meaning. These are visual clues to further enhance the self-descriptiveness property of the prototype.

Observe, that on 'data utility' page, the statistics are presented using a doughnut chart to show that a higher percentage is preferable as it relates to higher data quality. Whereas, in the 'data risk' page a higher percentage is not preferable as it corresponds to higher disclosure risk. Therefore, a different visualization technique, i.e a bullet chart, is used to convey this contrast to the user.

For the sake of consistency, the same sections are used from the dashboard to display the risk in input data versus output data. Thus, providing further assistance to the user when analysing results. In the case of 'data utility' page, quality models are only applicable to the anonymized data set. Therefore, the sections are arranged in such a way to show the utility measures alongside the output data.

Since the user completed all the steps needed to anonymize the data, all menu options are now accessible. The user can now keep tweaking the configurations until the desired level of anonymization is achieved. Once satisfied with the output data, the user can export it using the 'file' menu option.

5.3. Evaluation

5.3.1. Purpose

The purpose of the evaluation was to observe how the prototype is perceived by the general users of ARX. The prototype addressed the problem areas that translated into ARX's complexity. Through an evaluation it was examined if the prototype indeed increased the usability while providing a uniform, non-expert approach to anonymize a data set in minimal steps.

5.3.2. Setup

The evaluation was carried out with five participants who had earlier been interviewed on ARX. This time, the participants were only contacted via an email (See Appendix G, 2 and 3) and given brief instructions on the data collection process. To increase the response rate the participants were asked to circulate it among their peers who had some experience with ARX. However, no response was received and hence, the evaluation was carried out with only five participants.

The data collection process consisted of a survey that was based on 15 statements which analysed elements of an SDC application that were identified during the previous data collection method and then addressed in the prototype. For example, measurable criteria identified in the QUIM model [84] formed some of the statements that directly measured the improvement in the problem areas identified previously with ARX. The survey required the participants to rate their agreement with the statements on a 5 point Likert scale. The two surveys can be found in Appendix F.

The participants were first asked to fill the survey evaluating ARX. Then, they were given the link to the prototype (Danaamta) and asked to explore it. On completing this, they were asked to fill the same survey, but now evaluating the prototype. Besides, they were also asked to share feedback and their experience with the prototype.

It should be noted, that the participants were given basic written instructions on how to use the prototyping software tool in the browser and no explanation on the prototype itself. They were asked to explore it for multiple days. This was done to reduce some response biases and also, to emulate a real-world setting where a user of a new technology takes time to learn it on their own.

5.3.3. Method of Analysis

The data was analysed by first visualizing it through diverging stacked bar charts to compare the results. Then, a significance test was conducted on the two data sets. A non-parametric test like the Wilcoxon signed-rank test was applied [9]. This test is appropriate for evaluating two samples where the same subjects are evaluated under two different conditions or where

the population cannot be assumed to be normally distributed [82]. In this case, the same set of participants were asked to fill the survey under two different conditions. First, for ARX and then for Danaamta.

The null hypothesis (H_0) tested here asserts that the medians of the two samples are identical. In this case, the H_0 is tested for all the 15 items on the Likert scale by calculating the individual p values and rejecting the H_0 for $p < 0.05$.

However, they could have been response bias and other limitation of such a study. For instance, participants were only provided with a prototype and not a fully functional tool. Hence, the prototype only gave idealistic results as part of the simulation.

5.3.4. Results

On plotting the graphs for the two surveys it was evaluated that the level of agreement to the software usability measurement statements in the case of Danaamta was much higher than in the case of ARX. Refer to Figures 5.2 and 5.3.

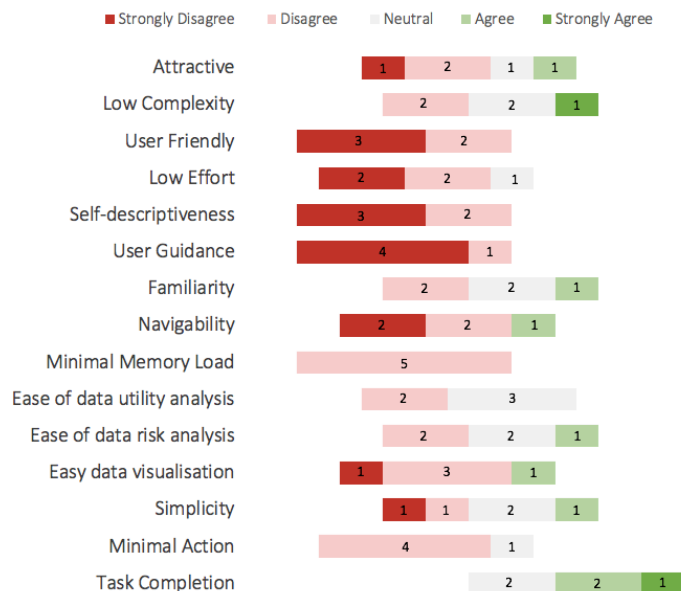


Figure 5.2: Graphical representation of the quantitative results for ARX

Table 5.1 provides the quantitative data collected from both the surveys and the test of significance i.e the p-value. Two out of fifteen items from the Likert scale survey had p-values greater than 0.05 and thus, failed to reject the H_0 . An explanation of these results is provided as follows.

First, the survey item that measured whether participants felt that the application reduces some of the complexity of the data anonymization theories and practices was not comparable. There was no significant evidence to support that one application ranked higher than the other in terms of reducing the complexity. This is interesting because as previously assumed that the complexity of the SDC theories is not well masked in ARX's functionalities is false. This can be explained by the way the participants would have interpreted the survey statement. The literature on the different anonymization models is difficult to comprehend as it deals with high-level statistics, so, instead of calculating these statistics manually ARX/Danaamta provides a button that does just this. Which could have been how the participants interpreted

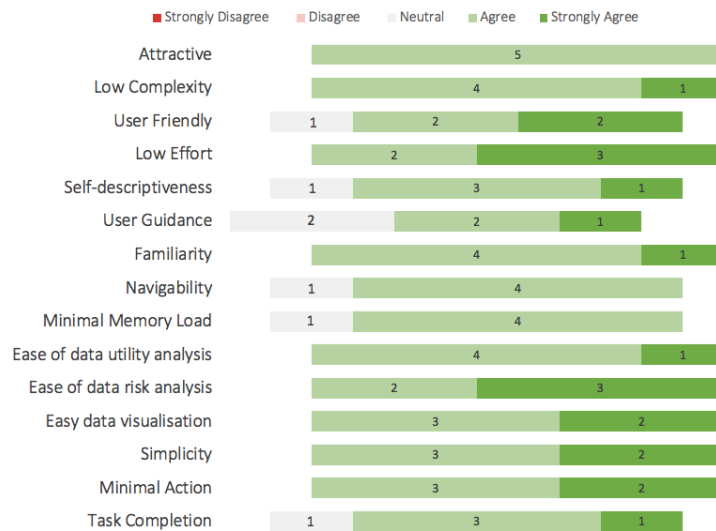


Figure 5.3: Graphical representation of the quantitative results for Danaamta

this survey item and not necessarily how the two compare with each other in abstracting complex concepts or explaining them.

Second, H_0 is not rejected for the Likert item which measures whether users can complete a specified task using the application. The reason for this could be that despite the complexities of ARX, users are still able to anonymize a data set. However, since the survey does not measure whether the completed task results in satisfactory results nor does it measure the cost of completion in terms of time and effort put in, this finding cannot be entirely significant. On the other hand, one Likert item came close to measuring how much effort needs to be put in to operate the application. Here, ARX ranked considerably low in comparison to Danaamta.

Independent feedback received from the participants revealed that they preferred the prototype over ARX especially if they were first-time users of SDC tools. To quote one such feedback *"I really like the prototype. It already made it way simpler to do data anonymization"*. They described the layout as *clear* and the anonymization process *sequential* in design. However, it is good to note that majority of the participants felt that such a design might not fair well with complicated data sets or highly skilled experts. This insight is not surprising as the intention of the prototype was never to directly replace sophisticated tools like ARX that are better equipped to handle complex data sets.

The prototype was made with the intention of guiding entry-level users i.e general users through the anonymization process. As a consequence, with the use of concepts like tunnelling and selective attention, the prototype could be interpreted to be influential in determining the path of the privacy analysis. This could make the prototype persuasive in design. Persuasive design use principles of psychology to design effective and engaging interfaces [71]. While this works for game designs or social networking websites, this might not be acceptable for an SDC tool. Here, the process of privacy analysis of sensitive data could seem biased. This is an insight that can be revisited in future research.

Overall Danaamta received favourable responses in contrast to ARX. It overcame the six problems that were identified in the investigation of ARX.

Strongly Disagree = 1, Disagree = 2, Neutral = 3, Agree = 4, Strongly Agree = 5 sample size = 4, decimals up to 3 places						
Measure	Statement	ARX	SD _{ARX}	Danaamta	SD _{Danaamta}	p
Attractive	The desktop application is attractive to the user, in terms of its colour or graphical user interface	1	1.019	4	0	0.025
		2		4		
		2		4		
		3		4		
Low complexity	The application reduces some complexity of the data anonymization theories	4	1.095	4	0.4	0.103
		2		4		
		2		4		
		3		4		
		3		4		
User friendly	The application can be used by entry-level users	5	0.489	3	0.748	0.01
		1		4		
		1		4		
		2		5		
		2		5		
Low effort	The amount of effort to operate the desktop application is low	1	0.748	4	0.489	0.01
		1		4		
		2		5		
		2		5		
Self-descriptiveness	The desktop application conveys its purpose and give clear user assistance in its operation	3	0.489	3	0.632	0.01
		1		4		
		1		4		
		2		4		
		2		5		
User guidance	The user interface provides context-sensitive help and meaningful feedback	1	0.4	3	0.748	0.009
		1		3		
		1		4		
		1		4		
		2		5		
Familiarity	The user interface has recognizable elements and interactions that can be understood by the user	2	0.748	4	0.4	0.025
		2		4		
		3		4		
		3		4		
		4		5		
Navigability	It is easy to navigate the desktop application in an efficient way	1	1.095	3	0.4	0.043
		1		4		
		2		4		
		2		4		
		4		4		
Minimal memory load	The user is required to keep minimal amount of information in mind in order to achieve a specified task	2	0	3	0.4	0.006
		2		4		
		2		4		
		2		4		
		2		4		
Ease of data utility analysis	It is easy to analyse the utility of the data	2	0.489	4	0.4	0.009
		2		4		
		3		4		
		3		4		
		3		5		
Ease of data risk analysis	It is easy to analyse the data risk	2	0.748	4	0.489	0.018
		2		4		
		3		5		
		3		5		
		4		5		
Easy data visualisation	It is easy to understand the visual content of the desktop application	1	0.979	4	0.489	0.021
		2		4		
		2		4		
		2		5		
		4		5		
Simplicity	The user interface is simplistic in design without using irrelevant elements	1	1.019	4	0.489	0.023
		2		4		
		3		4		
		3		5		
		4		5		
Minimal Action	The desktop application helps the user achieve their task in minimal number of steps	2	0.4	4	0.489	0.009
		2		4		
		2		4		
		2		5		
		3		5		
Task completion	A user can complete a specified task using the desktop application	3	0.748	3	0.632	0.734
		3		4		
		4		4		
		4		4		
		5		5		

Table 5.1: Results of quantitative data analysis

5.4. Conclusion

The prototype encompassed the requirements identified in the previous chapter. A minimalist design with clear segregation between the steps needed to anonymize a data set was developed. The design avoids visual clutter and does not strive to overwhelm the user by revealing too many functionalities at each step. Rather, the functionalities are introduced sequentially to some extent, playing on the selective attention of the user.

Danaamta was then evaluated by conducting quantitative interviews to measure its usability contrast with ARX. The results revealed that the user group preferred Danaamta over ARX. Findings suggested that Danaamta overcame the problems of ARX which were identified in previous chapters. But, there was no significance in the results that measured whether Danaamta actually reduced the complexity that is associated with the theories of data anonymization process. It was perceived that the user could complete the specific task using either of the applications. This observation could have been better justified if the interviews could measure the accuracy and cost of completing the task using the two applications given the diversity in the skill level of the participants with a larger sample size.

Lastly, the design of the prototype was reflected upon to be persuasive in nature which might not be preferred in practice. Danaamta tried to reduce the complexity of ARX by providing a simplified, user-friendly approach to data anonymization. As stated earlier, it cannot be perceived as a direct replacement of ARX but an alternative approach for introducing the process of anonymization to non-experts.

6

Discussion

In this chapter, important aspects of the thesis are discussed. First, the thesis is reflected upon in section 6.1. This is followed by indicating the contributions of such a research in section 6.2. Then, its limitations are discussed in section 6.3. Section 6.4 and 6.5 provide recommendations and areas of future research respectively. Lastly, the chapter concludes by emphasizing the relevance of the thesis to the master's programme.

6.1. Reflection

ARX packs a lot of functionalities in its design to provide its users with an arsenal of anonymization techniques in hopes of resulting in the best anonymized data. However, it does not take into consideration the needs of the users and a practical scenario in which its users might not be experts. This implicit assumption that its users will be well versed in the theories of SDC is its limitation of being adopted at a large scale. A reason for this approach can be the infancy and instability of the SDC field. Therefore, the developers of ARX are not aiming at mass adoption but rather at a stable software solution that incorporates existing research.

During the research, it was found that the complexity of a software tool is deep-rooted in the functionality it provides. A messaging application is a mere communication platform, nothing more, nothing less. On the other hand, a data anonymization tool holds rather a much more complex definition. The fact that there is no one-shot operation of eliminating the risk of data disclosure but only measures to reduce it is in itself quite tricky to explain.

Thus, ARX provides multiple functionalities to give its users the capabilities to anonymize a data set to the best of their abilities. But this, overwhelming choice of options mostly leads to a higher cognitive effort by the user. Subsequently, leading the user to make rather mediocre decisions which is posited by the literature on the paradox of choice.

Moreover, given the low user base of ARX, it is quite understood that there aren't many supporting materials to explain its functionalities in detail or tutorials to see how it can be used at its maximum potential. For cases like these, software needs to be self-contained. Meaning that they do not depend on third-party materials to explain its functionalities but are self-descriptive through its design. This is partially explained by technology adoption models where facilitating conditions like technical infrastructure can influence user behaviour.

ARX's design is not developed to please an average user with fancy or in-trend UI elements nor

is it designed for easy access for entry-level users. It is designed to be used by experts which is an assumption that stems from the fact that handling data for the process of anonymization is a task best left to experts.

However, there has been an oversight in this assumption. As Helen Hayes rightly put "*every expert was once a beginner*" [11]. Everyone has to start somewhere in order to become an expert and hence beginning from a complex but stable solution like ARX might not be the right approach. There is a need to start small, to master the basics before moving onto more sophisticated concepts to build that expertise. And this is the gap that the prototype might be able to bridge.

Thus, the thesis explores the usability problems of ARX from the perspective of non-experts. Six problem areas are identified, they are minimal memory load, self-descriptiveness, user guidance, navigability, minimal action and familiarity. The prototype tackles these problems by adhering to specific software guidelines. Additionally, the prototype is simplified in its functionalities to address the issue of choice overloading.

On evaluating the prototype it was found that the prototype indeed is preferred over ARX by entry-level users, however, it failed to abstract the complex concepts of SDC theories. This could be explained by how the prototype was evaluated. In conclusion, the prototype was able to bridge the application gap for non-experts in the field of SDC to some extent.

6.2. Contributions

6.2.1. Theoretical Contribution

Despite the advancements in the field of anonymization approaches of data sets and the development of enabling tools there hasn't been any focus on analysing the usability of such tools or making them accessible for practical use. Though it is understood that the field of SDC is slightly volatile, there are still some robust SDC techniques that can be applied. Individual organizations like statistical bureaus have developed standalone software solutions. However, these are not open to the general public. Therefore, such a research makes direct contributions to the usage of SDC tools by focusing on making them accessible to anyone.

Privacy is considered to be an important feature and there are stringent laws that protect it. There is a general expectation that compromises cannot be made when it comes to protecting private data. Therefore, tools like ARX are developed from the perspective of experts as they are believed to be the ones who should ideally handle private data. The thesis contributes to the fact that though we have such expectations we do not provide a learning platform for these "to-be" experts. The prototype proposed in this research provides such a platform that can facilitate learning.

Lastly, the research contributes to a valuable learning about the reduction of complexity especially in software tools that have complex functionalities. Intricate concepts like SDC can be abstracted in software tools through design guidelines.

6.2.2. Practical Contribution

The prototype tried to address the complexities of ARX by streamlining functionalities that could result in better decisions without overloading the user with choices. This is attributed to referring to the literature on choice overloading. This cognitive process is usually observed in online and offline markets where a customer has to choose between different products. However, the research showed that this concept can also be applied to complex applications

such as ARX where there are multiple functionalities to achieve a singular purpose. Therefore, when designing such applications, elimination of choice overloading should be taken into consideration as a software guideline.

Additionally, the theory suggests a number of software guidelines that have to be considered when designing a software. However, this necessarily does not mean that the resulting software would be a successful design. Often, these guidelines have to be filtered and tailored to the needs of the user as well as the context in which it will be used. Thus, the software guidelines cannot be followed blindly and have to take into account the perceived use-fulness of their application as a whole. Hence, the prototype developed provided a simplified universal approach to data anonymization using only specific software guidelines which helped in increasing its usability to some extent.

Technology acceptance models can explain to some extent why these tools are not adopted. However, in this case, they could not be used to improve their adoption rate but only partly understand the problem.

Lastly, organisations who are looking to use these tools can benefit from this thesis greatly. They can incorporate the use of this tool to shorten the learning curve associated with the anonymization process. Developers of SDC tools can also learn from this thesis as they can gain valuable insights on how to design these tools for the purpose of better usability and adoption.

6.3. Limitations

The thesis has some limitations which can be observed in the research methodology, literature and prototype design. They are:

- The research methodology had to be altered in light of the COVID-19 situation. Instead of a traditional usability study, the approach had to be modified to a virtual one-on-one interview with the participants. This is one reason why the interview process was structured as a mixed-methods approach with a heuristic evaluation in the qualitative part of the interview process. While there could be some drawbacks in the significance of the results as compared to a fully fledged usability study, the research was effective in finding certain problem areas in ARX. While these findings might not be exhaustive they are still noteworthy.
- The research was also restricted due to a small sample size. This might not be significant for the exploratory study conducted on the problems users face with ARX as a large sample size is not needed for problem finding research. However, it can be a drawback during the evaluation of the prototype where the reliability of the results is affected. Conducting a non-parametric test on Likert data with a sample size < 10 and with 15 Likert items could result in low reliability of the survey results.

Moreover, a small sample size also restricts diversity in the participants. In this case, all participants were from an IT background and the same set of participants took part in both the interview processes. This could have impacted the evaluation results as the participants were aware of what kind of problems ARX had and would have an inclination of what had been improved in the prototype.

- The QUIM model provided some Likert data items that could measure the usability of the applications. However, the model cannot be applied to measure the time taken to complete a task and judge the goodness of the results. This was also restricted by the prototype design as it is only a simulation and not a fully functional tool.

- The prototype was persuasive by design as it was designed to guide entry-level users. There is a possibility that it nudged users in a specific direction during the anonymization process. While this maybe acceptable and needed for entry-level users, it might not be preferred in practical applications where the process might seem biased and not preferred by expert-level users who might like to apply their own reasoning to arrive at acceptable results.
- While the prototype might be a good approach to address the problem statement, there could be other approaches to solving the same problem such as increasing the usability of ARX by directly suggesting some design changes or creating a learning framework to understand ARX.

6.4. Future Research

The thesis is finally concluded by providing some suggestions for future research and recommendations for further development:

- Integrating with the APIs of ARX to provide a fully functional prototype. This can help in evaluating the prototype much more extensively and effectively. For example, interview participants can be asked to anonymize the same data set first with ARX and then with the prototype to compare the results side-by-side. In such a case, factors such as time taken to complete the task, goodness of the results in terms of resulting data quality and reduced risk of disclosure can be compared.
- To evaluate the prototype with a larger sample size or within the context of the organisation such as participants who could be the potential data processors. Their valuable input can be incorporated into the design. The research can go one step forward by going beyond the organisation to generalise the results to the population.
- To conduct a similar study with experts. Recall the stakeholder analysis, where SDC experts are identified as context setters. They have far better knowledge of the working of SDC practices and can therefore, provide valuable insights that can influence the software requirements for developing a prototype.
- Incorporating other approaches to data anonymization such as differential privacy as it eliminates the need for attack modelling [40] which is not the case in k-anonymity model and its extensions. Thereby, other measures for analysing privacy risk can be considered.

6.5. Recommendations

- It is possible that in practical applications data controllers might be apprehensive to use a simplified design. The anonymized data they release will be assumed to be subjected under attack from data intruders and therefore, they cannot make any compromises in its anonymization process. They might perceive a complex tool as ARX to give better anonymization results just by the extent of the functionalities it provides. In such a case, the prototype can be a tool that is used to expose the employees of an organization to the field of SDC without overwhelming them with its complexities. Thus, through the concept of micro-learning [24], the employees can be managed to move on to much more advance tools.
- Data anonymization is only a small part of data pre-processing phase in the OGD lifecycle [17]. The success of open-data systems requires the government to take a broader perspective than just the simple provision of access to data [49]. Designing a

mechanism that follows an onion layer principle where data is protected in a series of multiple layers as there is no single method to protect data sets.

The steps can include pre-processing methods such as collecting and storing that data which is absolutely necessary, vetting the data that needs to be opened by experts before it is approved for anonymization. Thus, the data can be protected much before it is fed into the anonymization tool and hence reduce some more of the privacy risk associated with it.

After the data has been anonymized, it can undergo post-processing methods like vigorous testing by ethical data intruders or sharing certain anonymized data sets with other organizations through selective disclosure where they are given access to the data for a short amount of time on contractual basis.

6.6. Relevance to the MOT programme

The thesis research was completed as part of the Management of Technology (MOT) MSc. programme offered by TU Delft. This course focuses on learning to "*explore and understand technology as a corporate resource*". The thesis refers to the technology that is offered as SDC software tools, identifies problems to adoption and focuses on one problem to improve their adoption.

The thesis weighs in on the significance of managing SDC tools as their usage lags behind when they can be used to open data sets and realise important initiatives like OGD goals. Thereby, making these tools important resources for public as well as private organisations.

The research study identifies a gap in the adoption of these tools and develops a prototype to address that gap by utilizing the knowledge gained from the MOT courses such as Technology Dynamics, Business Process Management and Technology, Emerging and Breakthrough Technologies, I and C Architecture Design, I and C Service Design, etc. Therefore, such a research study is highly relevant to the MOT programme as it directly relates to one of the main objectives of this course.

7

Conclusion

The goal of this thesis was to provide a solution for addressing the usability of anonymization software tools. The use of these tools is important as they provide methods for anonymizing data that can be released to the public. A practical application of these tools can be in OGD initiatives where opening data sets have many advantages. However, it is not easy to use these tools. This can be attributed to how these tools are designed and the knowledge needed to operate them.

Most of these tools are designed from the perspective of experts or for the purpose of demonstration. Moreover, ongoing research in the field of SDC impedes the development of a single software tool that can address all privacy issues related to opening data sets. Rise of new anonymization methods or debunking old methods has led to a slow progress in not only the development of these tools but also their adoption. Resulting in limited support material and even smaller user base. As a consequence, individuals or organizations looking to adopt these tools to satisfy their data privacy objectives cannot use them.

Thus, the thesis addressed this problem by structuring the solution around five research questions. The first research question "*What are the typical challenges of using SDC tools?*" was answered in chapter 2 where a literature review was conducted. The literature on technology adoption models showed how different factors aid in the adoption of technology and that it is not an outcome of the success of a single factor but rather a culmination of many interdependent factors. The literature on how a software interface is designed plays a significant role in easing some of the challenges users face. Especially, if the underlining functionality of the software is complex then the way it is presented to the user is important. A lot of the complexity of a software can be abstracted by the way it is designed. Moreover, its design can also directly translates into its usability. In summary, the typical challenges of using SDC tools can be as simple as user knowledge (from the UTAUT model) or the lack of design for usability.

The second research question "*What are the usability problems with ARX in practice?*" was addressed in Chapter 3. A mixed-methods exploratory research approach was undertaken to identify the problems with ARX. The scope was defined to focus only on those problems that could be tackled in a new design of SDC tool.

Chapter 3 also laid the foundation for answering the third research question "*What are the requirements to overcome these problems?*". The problems identified during the investigation of ARX were addressed in Chapter 4. In this chapter, software requirements were defined. The

requirements were divided into two parts. First, user requirements were determined to tackle the problems that users faced with ARX. Second, functional requirements were reduced and streamlined from existing SDC guidelines and methods found in the literature. The functional requirements were kept similar to ARX to some extent.

Then, in Chapter 5 the requirements were translated into a prototype design called Danaamta to answer the fourth research question "*How are the requirements translated into a design?*". Followed by the final research question "*How is the resulting design perceived by users of ARX?*" where the prototype was evaluated through quantitative research methods. Subsequently receiving favourable responses in comparison to ARX.

Lastly, the thesis was reflected upon in Chapter 6 by highlighting the contributions, limitations and future research.

The thesis was instrumental in addressing the major usability problems of ARX and improving upon its design through Danaamta. However, this research is all but a drop in the ocean as there are further barriers to the actual adoption of these tools in practice. For now, we can only hope to bridge the gap for all those people who want to delve into the application of SDC methods by providing them a platform to build their expertise.

Bibliography

- [1] ARX - data anonymization tool. [Online]. Available: <https://arx.deidentifier.org/>.
- [2] Cornell Anonymization Toolkit. [Online]. Available: <http://sourceforge.net/p/anony-toolkit/>.
- [3] UTD Anonymization Toolbox. [Online]. Available: <http://cs.utdallas.edu/dspl/cgi-bin/toolbox/>.
- [4] Misdaad in Kaart. [Online]. Available: <https://www.politie.nl/themas/misdaad-in-kaart.html>.
- [5] Organisation for Economic Co-operation and Development. [Online]. Available: <https://stats.oecd.org/>.
- [6] Open Government Data. [Online]. Available: <https://opengovernmentdata.org/>.
- [7] About PARAT De-Identification Software. [Online]. Available: <http://www.privacynalytics.ca/software/parat/>.
- [8] Software Engineering Chapter 2: Software Requirements. [Online]. Available: <http://www.inf.ed.ac.uk/teaching/courses/cs2/LectureNotes/CS2Ah/SoftEng/se02.pdf>.
- [9] Wilcoxon Signed-Rank Test Calculator. [Online]. Available: <https://www.aatbio.com/tools/mann-whitney-wilcoxon-signed-rank-test-calculator>.
- [10] Design guidance - displaying adverse drug reaction risks. *Microsoft*, 2009.
- [11] "Every expert was once a beginner." Are there other famous quotes that mean the same thing?, 2015. [Online]. Available: <https://www.enotes.com/homework-help/every-expert-was-once-beginner-other-quotes-mean-559169>.
- [12] P. Ajibade. Technology acceptance model limitations and criticisms: Exploring the practical applications and use in technology-related studies, mixed-method, and qualitative researches. *Library Philosophy & Practice*, 2018.
- [13] A Amighi, M. S. Bargh, S. Choenni, A. Latenko, and R. Meijer. On Protecting Microdata in Open Data Settings from a Data Utility Perspective. In *Proceedings of The Fourteenth International Conference on Digital Society (ICDS)*, pages 38–47, 2020.
- [14] B. Anda, H. Dreiem, D. I. K. Sjøberg, and M. Jørgensen. Estimating Software Development Effort Based on Use Cases — Experiences from Industry. In M. Gogolla and C. Kobryn, editors, *The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, pages 487–502. Springer Berlin Heidelberg, 2001.
- [15] A. I. Anton. Goal-based requirements analysis. In *Proceedings of the Second International Conference on Requirements Engineering*, pages 136–144, 1996.

-
- [16] C. Arthur. Tech giants may be huge, but nothing matches big data, 2013. [Online]. Available: <https://www.theguardian.com/technology/2013/aug/23/tech-giants-data>.
- [17] J. Attard, F. Orlandi, S. Scerri, and S. Auer. A systematic review of open government data initiatives. *Government Information Quarterly*, 32(4):399–418, 2015.
- [18] N. Babich. Golden rules of user interface design, 2016. [Online]. Available: <https://uxplanet.org/golden-rules-of-user-interface-design-19282aeb06b>.
- [19] M. S. Bargh, R. Meijer, and M. Vink. On statistical disclosure control technologies: For enabling personal data protection in open data settings. *Research and Documentation Center WODC, The Hague, The Netherlands, Tech. Rep.*, Cahier 2018-20.
- [20] T. Benschop, C. Machingauta, and M. Welch. Statistical Disclosure Control: A Practice Guide. 2019.
- [21] H. N. Boone and D. A. Boone. Analyzing likert data. *Journal of extension*, 50(2):1–5, 2012.
- [22] J. Bosch and P. Molin. Software architecture design: evaluation and transformation. In *Proceedings ECBS'99. IEEE Conference and Workshop on Engineering of Computer-Based Systems*, pages 4–10, 1999.
- [23] L. Bourne and D. Walker. Visualising and mapping stakeholder influence. *Management Decision*, 43:649–660, 2005.
- [24] I. Buchem and H. Hamelmann. Microlearning: a strategy for ongoing professional development. *eLearning Papers*, 21(7):1–15, 2010.
- [25] B. G. Cameron, E. F. Crawley, G. Loureiro, and E. S. Rebentisch. Value flow mapping: Using networks to inform stakeholder analysis. *Acta Astronautica*, 62:324–333, 2008. ISSN 0094-5765.
- [26] L. J. Camp. Designing for trust. In *Workshop on Deception, Fraud and Trust in Agent Societies*, pages 15–29. Springer, 2002.
- [27] T. Carmichael and N. Cunningham. Theoretical Data Collection and Data Analysis with Gerunds in a Constructivist Grounded Theory Study. *Electronic Journal of Business Research Methods*, 15(2), 2017.
- [28] L. L. Constantine and L. A. D. Lockwood. Structure and style in use cases for user interface design. *Object modeling and user interface design*, pages 245–280, 2001.
- [29] J. Creswell, V. Clark, M. Gutmann, and W. Hanson. *Advance Mixed methods Research Designs*, pages 209–240. 01 2003.
- [30] J. L. Cybulski, S. Keller, and D. Saundage. Interactive exploration of data with visual metaphors. *International Journal of Software Engineering and Knowledge Engineering*, 25(02):231–252, 2015.
- [31] F. D. Davis. *A technology acceptance model for empirically testing new end-user information systems: Theory and results*. PhD thesis, Massachusetts Institute of Technology, 1985.

-
- [32] F. D. Davis and V. Venkatesh. Toward pre-prototype user acceptance testing of new information systems: implications for software project management. *IEEE Transactions on Engineering Management*, 51(1):31–46, 2004. ISSN 1558-0040.
- [33] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw. User acceptance of computer technology: a comparison of two theoretical models. *Management science*, 35(8):982–1003, 1989.
- [34] S. S. Dawes. Stewardship and usefulness: Policy principles for information-based transparency. *Government Information Quarterly*, 27(4):377–383, 2010.
- [35] J. Domingo-Ferrer and J. M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. *IEEE Transactions on Knowledge and data Engineering*, 14(1):189–201, 2002.
- [36] J. Domingo-Ferrer and V. Torra. Disclosure risk assessment in statistical data protection. *Journal of Computational and Applied Mathematics*, 164:285–293, 2004.
- [37] J. Domingo-Ferrer, D. Sánchez, and J. Soria-Comas. Database anonymization: privacy models, data utility, and microaggregation-based inter-model connections. *Synthesis Lectures on Information Security, Privacy, & Trust*, 8(1):1–136, 2016.
- [38] P. Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87, 2012.
- [39] C. Dwork. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pages 1–19. Springer, 2008.
- [40] C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.
- [41] X. Ferré, N. Juristo, H. Windl, and L. Constantine. Usability basics for software developers. *IEEE software*, 18(1):22–29, 2001.
- [42] G. Fischer. User modeling in human–computer interaction. *User modeling and user-adapted interaction*, 11(1-2):65–86, 2001.
- [43] A. Frey, S. Dupuy-Chessa, and G. Calvary. Model based self-explanatory user interfaces. *Ingenierie des Systemes d’Information*, 22:129–157, 01 2017.
- [44] J. J. Garrett. *The Elements of User Experience: User-Centered Design for the Web and Beyond*. Pearson Education, 2010.
- [45] J. D. Gould and C. Lewis. Designing for usability-key principles and what designers think. In *Proceedings of the Conference on Human Factors in Computing Systems*, volume 28, pages 50–53, 1983.
- [46] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, R. Lenz, J. Naylor, and P.P. De Wolf. Handbook on Statistical Disclosure Control. 2010.
- [47] S. S. Iyengar and M. R. Lepper. When choice is demotivating: Can one desire too much of a good thing? *Journal of personality and social psychology*, 79(6):995, 2000.
- [48] K. Janssen. The influence of the PSI directive on open government data: An overview of recent developments. *Government Information Quarterly*, 28(4):446–456, 2011.

-
- [49] M. Janssen, Y. Charalabidis, and A. Zuiderwijk. Benefits, Adoption Barriers and Myths of Open Data and Open Government. *Information Systems Management*, 29(4):258–268, 2012.
- [50] R. B. Johnson, A. J. Onwuegbuzie, and L. A. Turner. Toward a definition of mixed methods research. *Journal of mixed methods research*, 1(2):112–133, 2007.
- [51] E. Kerti. Designing effective step-by-step process, 2018. [Online]. Available: <https://blog.prototypr.io/designing-effective-step-by-step-process-29303c93da08>.
- [52] A. C. Klassen, J. Creswell, V. L. P. Clark, K. C. Smith, and H. I. Meissner. Best practices in mixed methods for quality of life research. *Quality of Life Research*, 21(3):377–380, 2012.
- [53] F. Kohlmayer, F. Prasser, and K. A. Kuhn. The Cost of Quality: Implementing generalization and suppression for anonymizing biomedical data with minimal information loss. *Journal of Biomedical Informatics*, 58:37–48, 2015.
- [54] D. Lambert. Measures of disclosure risk and harm. *Journal of Official Statistics (JOS)*, 9:313–313, 1993.
- [55] J. Li. Blockchain Technology Adoption: Examining the Fundamental Drivers. In *Proceedings of the 2020 2nd International Conference on Management Science and Industrial Engineering*, MSIE 2020, page 253–260, New York, NY, USA, 2020. Association for Computing Machinery.
- [56] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *2007 IEEE 23rd International Conference on Data Engineering*, pages 106–115. IEEE, 2007.
- [57] N. Li, W. Qardaji, D. Su, Y. Wu, and W. Yang. Membership privacy: a unifying framework for privacy definitions. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 889–900, 2013.
- [58] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkatasubramanian. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3–es, 2007.
- [59] T. J. Madden, P. S. Ellen, and I. Ajzen. A comparison of the theory of planned behavior and the theory of reasoned action. *Personality and social psychology Bulletin*, 18(1): 3–9, 1992.
- [60] E. Mark and J. Domingo-Ferrer. The future of statistical disclosure control. *National Statistician’s Quality Review - Government Statistical Service, UK*, abs/1812.09204, 2018.
- [61] L. M. Maruping, H. Bala, V. Venkatesh, and S. A. Brown. Going beyond intention: Integrating behavioral expectation into the unified theory of acceptance and use of technology. *Journal of the Association for Information Science and Technology*, 68(3): 623–637, 2017.
- [62] J. J. McDermott and T. J. Shimeall. Software Security in an Internet World: An Executive Summary. *IEEE Software*, 18(04):58–61, 1999. ISSN 1937-4194.

-
- [63] M. B. Miles and A. M. Huberman. *Qualitative data analysis: An expanded sourcebook*. sage, 1994.
- [64] L. A. Miller and J. C. Thomas Jr. Behavioral issues in the use of interactive systems. *International journal of human-computer studies*, 51(2):169–196, 1999.
- [65] M. G. Morris and A. Dillon. How user perceptions influence software use. *IEEE Software*, 14(4):58–65, 1997. ISSN 1937-4194.
- [66] R. Newcombe. From client to project stakeholders: A stakeholder mapping approach. *Construction management and economics*, 21(8):841–848, 2003.
- [67] J. Nielsen. Finding Usability Problems through Heuristic Evaluation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, page 373–380, New York, NY, USA, 1992. Association for Computing Machinery.
- [68] J. Nielsen. Usability inspection methods. In *Conference companion on Human factors in computing systems*, pages 413–414, 1994.
- [69] J. Nielsen. 113 Design Guidelines for Homepage Usability, 2001. [Online]. Available: <https://www.nngroup.com/articles/113-design-guidelines-homepage-usability/>.
- [70] J. Nielsen and R. Molich. Heuristic Evaluation of User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, page 249–256, New York, NY, USA, 1990. Association for Computing Machinery.
- [71] M. Oduor, T. Alahäivälä, and H. Oinas-Kukkonen. Persuasive software design patterns for social influence. *Personal and Ubiquitous Computing*, 58:1689–1704, 2014.
- [72] U. Östlund, L. Kidd, Y. Wengström, and N. Rowa-Dewar. Combining qualitative and quantitative research within mixed method research designs: a methodological review. *International journal of nursing studies*, 48(3):369–383, 2011.
- [73] H. Page, C. Cabot, and K. Nissim. Differential privacy an introduction for statistical agencies. *NSQR. Government Statistical Service*, 2018.
- [74] N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, and Y. Kim. Data Synthesis Based on Generative Adversarial Networks. *Proc. VLDB Endow.*, 11(10):1071–1083, June 2018. ISSN 2150-8097.
- [75] N. Patel. The Paradox of Choice: Why Less is More in UX Design, 2017. [Online]. Available: <https://usabilla.com/blog/paradox-choice-less-ux-design/>.
- [76] P. Pierce. 10 Guidelines For Navigation Usability, 2019. [Online]. Available: <https://usabilitygeek.com/10-guidelines-for-navigation-usability/>.
- [77] F. Prasser, F. Kohlmayer, R. Lautenschläger, and K. A. Kuhn. ARX—A Comprehensive Tool for Anonymizing Biomedical Data. In *Proceedings of the Annual Symposium / AMIA Symposium*, pages 984–993, 2014.
- [78] F. Prasser, J. Eicher, H. Spengler, R. Bild, and K. A. Kuhn. Flexible data anonymization using ARX—Current status and challenges ahead. *Software: Practice and Experience*, 2020.

-
- [79] G. Reitmayr and D. Schmalstieg. OpenTracker: A flexible software design for three-dimensional interaction. *Virtual reality*, 9(1):79–92, 2005.
- [80] D. Rosenberg and K. Scott. *Use case driven object modeling with UML*. Springer, 1999.
- [81] J. Saldaña. *The coding manual for qualitative researchers*. Sage, 2015.
- [82] S. W. Scheff. *Fundamental statistical principles for the neurobiologist: A survival guide*. Academic Press, 2016.
- [83] B. Scheibehenne, R. Greifeneder, and P. M. Todd. Can there ever be too many options? a meta-analytic review of choice overload. *Journal of consumer research*, 37(3):409–425, 2010.
- [84] A. Seffah, M. Donyaee, R. B. Kline, and H. K. Padda. Usability measurement and metrics: A consolidated model. *Software quality journal*, 14(2):159–178, 2006.
- [85] U. Sekaran and R. Bougie. *Research methods for business: A skill building approach*. John Wiley & Sons, 2016.
- [86] B. Shneiderman and M. Leavitt. Research-based web design & usability guidelines. *Washington DC, Department of Health and Human Services*, 2006.
- [87] J. M. Six and R. Macefield. How to Determine the Right Number of Participants for Usability Studies, 2016. [Online]. Available: <https://www.uxmatters.com/mt/archives/2016/01/how-to-determine-the-right-number-of-participants-for-usability-studies.php>.
- [88] C. J. Skinner and D. J. Holmes. Estimating the re-identification risk per record in microdata. *Journal of Official Statistics*, 14(4):361, 1998.
- [89] S. L. Smith and J. N. Mosier. Guidelines for designing user interface software. 1986.
- [90] J. Soria-Comas and J. Domingo-Ferrer. Big data privacy: challenges to privacy principles and models. *Data Science and Engineering*, 1(1):21–28, 2016.
- [91] A. Strauss and J. Corbin. *Basics of qualitative research techniques*. Sage publications Thousand Oaks, CA, 1998.
- [92] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.
- [93] H. Taherdoost. A review of technology acceptance and adoption models and theories. *Procedia manufacturing*, 22:960–967, 2018.
- [94] M. Templ. Statistical Disclosure Control for Microdata Using the R-Package SdcMicro. *Transactions on Data Privacy*, 1(2):67–85, 2008. ISSN 1888-5063.
- [95] J. Y.L. Thong, W. Hong, and K.Y. Tam. Understanding user acceptance of digital libraries: what are the roles of interface characteristics, organizational context, and individual differences? *International journal of human-computer studies*, 57(3): 215–242, 2002.
- [96] M. M. Tseng and J. Jiao. A variant approach to product definition by recognizing functional requirement patterns. *Journal of Engineering Design*, 8(4):329–340, 1997.

- [97] B. Ubaldi. Open Government Data. 2013.
- [98] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis. User acceptance of information technology: Toward a unified view. *MIS quarterly*, pages 425–478, 2003.
- [99] K. Whintont. Minimize cognitive load to maximize usability, 2013. [Online]. Available: <https://www.nngroup.com/articles/minimize-cognitive-load/>.
- [100] L. E. Wood. *User interface design: Bridging the gap from user requirements to design*. CRC Press, 1997.

Appendices

Appendix A: Interview Transcripts

Interview 1

Q.Can you state your age?

A. I am 22 years old.

Q.Can you tell me a bit about your academic background?

A. I am in my last year of my Computer Science Bachelor.

Q.How did you come in contact with ARX?

A. Through a minor course.

Q.How did you learn to use it?

A. Mostly by trial and error. Looking up online tutorials and asking for help from other students.

Q.How were the online tutorials?

A. They were mostly subjective, pretty much what you searched for you got to see only that. It made it quite hard to experiment with other stuff.

Q.How much time did it take to learn it?

A. A span of multiple days. There were certain moments it was tempting to switch over to the programming for it. And that is what I did with my project team.

Q.When you say programming, do you mean using the ARX APIs instead of the GUI?

A. Yes.

Q.Why did you switch?

A. The short answer is that it was helpful. It was first hard to find my way through the program but once you get that working it is way faster than doing it in the GUI. But the most important part for me was just how painstaking the UI user experience was.

Q.How would you rate your proficiency in ARX?

A. With the APIs, it is very good because the examples in the documentation are very good. So, you can read a lot about it. I didn't even look for online tutorials for the APIs as the documentation was enough. But for the GUI I know somethings about it, but I don't feel quite proficient with it so to say.

Q.Did you try exploring the GUI by yourself?

A. I did not really experiment with the GUI because as far as I know whatever I could do in the APIs is also in the GUI.

Q.When you anonymized the data set did you feel that you could estimate whether your results were acceptable?

A. Yes, I could because of trial and error. You could see the suppression. You get pretty much the same picture in the GUI and the APIs.

Q.Would you recommend this application (GUI) to a non-programmer?

A. That is quite hard to answer. I had issues with the GUI, not just the user experience part but also that you already need to have some knowledge about the which principles there

are and the settings you can use. Basing my answer on that and also that I am a computer science student so I should be able to run the application and do something with it. So I find it hard to recommend just the GUI to someone else because the GUI is already quite hard but you have to know the theory anyhow. So, I would rather like to say quite APIs for it but that can be hard too if you don't have a background in programming. I would recommend it in the sense that it is really good for all privacy measure and such, but it is quite hard to get it working initially. It would require some helping hand.

Q.Can you explain what you mean by 'the GUI is hard'?

A. It boils down to the user experience. Multiple times you press a button, or you expect something to happen and maybe it does happen and maybe it doesn't. But It is not always why that happens. For example, there was a text input field for k-anonymity function. I felt I could edit it as I would do for a normal form. But that is not allowed and rather you have a knob-like button that you have to click on and drag your mouse in some kind of way and then it scrolls through the value of K. I thought this was not handy and somehow clicked it twice. I got a pop-up with the same text input field which was editable now. It was really weird. This is my primary example but all the structure of the application. It is not like reading a book – you go from left to right or up or down. You have some kind of order you like to look at in an application, but this was not the case in the GUI. You are constantly moving between different parts of the applications. There was a bit of disconnect when you edit in one tab then the tab next to it would change.

Q.Have you used any other anonymization tool?

A. ARX is the first and only tool.

Q.If you were hired by a public organisation to anonymize real confidential data would you use ARX to do it?

A. First, explore a bit. I am not sure if there are other tools, but It is good to have that knowledge. I know that the theory and utility/risk measures will remain the same. The only thing that will change will be the GUI and how you interact with it. Maybe other tools also have APIs. Having already worked with ARX I am confident I will be able to anonymize the data. The only thing I am not confident about is how well the data set is anonymized and that the data set has a good privacy standard. I also don't have any experience in measuring how well privacy is. I know there are statistics and I can read them like I know what they mean but I don't know what the impact is.

Q.Can you compare ARX with other desktop applications that you have used, for example, MATLAB or Adobe Photoshop?

A. I haven't used MATLAB but I have used Adobe Photoshop. The first thing that I can think of is the knob-like button in ARX which reminds me of old music player applications in which you have plug-ins. As for Adobe Photoshop, I know it is a powerful tool and still quite hard to use initially. Just like ARX you need to know the theory for any application for that matter. The hard part is how you achieve your goal in an application. It is the same with ARX. The ARX GUI is a bit more complicated in the sense that it doesn't always do what you expect it to do.

Usability Testing: The first time when you open the application you get this empty screen and by that time, I was already confused about how I could add my data to it. These buttons are on the top left corner for the screen. If I had to design it I would make it much simpler

taking examples from other desktop applications. You have multiple tabs on which you can do different things. It is kind of isolated. Making changes in one tab for sure has some kind of effect in other tabs but you shouldn't have to go between different tabs and different views within the applications to do something. It reminds me of a spaghetti that you can write in code which in this case is the GUI. The GUI works but you have to find all the features. The user experience is hard. The anonymization process has to be mapped out in such a way that you have to take a minimal road to do at least the basic steps. And within the minimal road, you can have different branches where you can apply different privacy models.

Interview 2

Q.Can you state your age?

A. I am 22 years old.

Q.Can you tell me a bit about your academic background?

A. I am in my final year of bachelors. I am studying Business IT and Management. I first did Information Science, but it was too much programming so now I am doing a little bit of the business side of IT.

Q.How did you come in contact with ARX?

A. Through a Data Science minor in an Ethics course in which we had to use ARX.

Q.How did you learn to use it?

A. The teacher tried to explain through slides in one lesson, but it wasn't really good to follow through. So, then I explored it by myself but that was pretty hard to do as there wasn't much information for me to find.

Q.Can you explain what you mean by 'there wasn't much information'?

A. There weren't any tutorials. There was the manual but that too was a bit complex. Because of the lack of information, we just tried exploring it by ourselves.

Q.When you say 'complex' do you mean the manual was complex or the application?

A. Both. The manual had some difficult words in it which you had to know before to use ARX. And then in ARX, you had to find where everything was, but it was difficult to find. It was also difficult to know all the measures (utility and risk) and what they were. It was not explained well in the manual.

Q.In the end were you able to understand the application?

A. I think a bit. Not as much as the teacher would have liked.

Q.Are you proficient in using ARX?

A. No, I am just a basic user. I knew what the input was and what the output should have been. So, if you know the output, you can work towards it. If you ask me something else about ARX I would probably say 'I don't know how to do this'.

Q.How much time did it take to learn it?

A. Since I knew what the output was and I was working with 3 other people, I would say it took me 3 days to learn it.

Q.Did you choose to work with other people because of the lack of information?

A. Not necessarily because we were in a project group. But I think if I was on my own, I would have reached out to other people for help.

Q.Can you tell me about your first impression of the applications?

A. Outdated.

Q.Can you elaborate on that?

A. The GUI does not seem attractive to me but then I understand that it is an anonymization tool and that it does not need to be fancy. The colours are grey. There is not much information about the buttons. My first impression was also that it was difficult.

Q.Can you explain what do you mean by 'difficult'?

A. On seeing it for the first time I see a lot of things and texts when I import the data set. I saw a lot of privacy models/utility measures and I can look them up online but there wasn't much information about it. So, it was difficult because of a lack of information. Maybe if there was good manual or tutorial then we could have figured it out better.

Q.Why did you say that you 'understand that it is an anonymization tool and that it does not need to be fancy'?

A. If I work with any software which looks attractive, I immediately feel that it is fine or a nice tool to work with. But my first impression with ARX was that this is going to be difficult and a little bit outdated. If it was a bit more attractive with a modern UI I would feel that it was fun to work with.

Q.You said that you knew what the output looked like but let's say if you were given a new data and you did not know the anonymized output for that. Would you be able to anonymize it?

A. Maybe. Because the teacher gave us a walk-through and told us some steps to be done. We only used I-diversity privacy model, but I do not know about the other models.

Q.Do you trust the application for anonymizing real confidential data?

A. Yes, I trust the tool not because I know the privacy models behind it, but I can analyse the utility and risk. I can see the percentages of how much of the data is anonymized and how much of the data is at risk. For me, that kind of information is precise, or it gives me a lot of information to think that it is reliable. But since I do not know the math behind it, I am not 100% sure.

Q.Who do you think this application is more suited to?

A. I think this application is more for experts because it has a lot of functionalities. You really have to know what you have to do good data anonymization.

Q.Can you compare ARX with other desktop applications that you have used, for example, MATLAB or Adobe Photoshop?

A. I have used Adobe Lightroom, Illustrator, Photoshop and all the Office applications, of course. Taking the example of Adobe Illustrator, I think it also made more for experts. But Illustrator is easier to go through because there are more tutorials about it and many people to help you. As for the user interface, it is clear where everything is if you know the Adobe suite. If you know photoshop then it easy to work with Illustrator because a lot of the things are the same. But if it is your first Adobe application then, in the beginning, it might easier to work because of the tutorials available. You don't even have to open the manual.

Usability Testing: When I load my data with the hierarchies, I don't know most of the elements on the screen. There is no function to click on them and get a text message that explains them. In the top section, you can click file, edit, view and help. There is nothing else. Maybe there could have been an information section below it. The information on the tabs speaks for themselves. But if you want to use a different privacy model, you don't know what they are because there is no information about it. I have to look it up on google or the manual. For me, this is lacking in the software that there is no quick lookup function. I don't know what the green ovals in the 'Explore Results' mean. I know a few measures in the 'Analyse Utility and Risk' tab. I think there is a nice overview with percentages ('Analyse Utility and Risk') given and it doesn't have too much information on the screen. I think it is good because if you have too much information then it could have an impact on much you could pick-up. In general, the GUI doesn't seem modern to me. And because of my lack of knowledge, I don't know what exactly it will function if I change something. But if you are an expert then you can do everything and then ARX will work fine. It is not fun to play with. If you want to reach a wider market, then I think the developers should work on the GUI.

Interview 3

Q.Can you state your age?

A. I am 23 years old.

Q.Can you tell me a bit about your academic background?

A. Sure, I am in my final year of doing a bachelor's in computer science.

Q.How did you come in contact with ARX?

A. We did a security class for a Data Science Minor in which we had to use ARX to anonymize a data set that was given to us by our teacher.

Q.How did you learn to use it?

A. I think I used a tutorial to see how the ARX application works and then just figuring it out myself by playing with it.

Q.And how was your learning experience?

A. It was quite difficult. I think the tutorial was somewhat comprehensible. But if I wanted to do some more advanced steps then I had to figure it out by myself through trial and error. It could have been a smooth experience if there was some sort of in-system tutorial.

Q.How much time did it take to learn it?

A. I think around 3 to 4 hours to get my way around the system.

Q.Are you proficient in using ARX?

A. I think I could find my way around the system. I knew what to do, where to find certain functions and carry out some actions.

Q.Can you explain what you mean by your learning experience being 'difficult'?

A. Sure. The steps that I wanted to take weren't clear from the beginning. So, there were a lot of buttons that do not have an exact description when you start the application. So, you have to search through all the stuff to find the right things you want to know.

Q.Can you tell me about your first impression of the applications?

A. It looks kind of old. It's pretty minimal. It could be more self-explanatory. I don't think it's really bad, but it could be better in terms of giving information to the user.

Q.Did you explore all the functionalities?

A. I used multiple privacy models to analyse certain risks.

Q.While using all these privacy models were you able to correctly judge what you were doing was proper?

A. That is a good question. For a big proportion, I had no idea if it was correct or not. I was just winging it hoping that it was correct. I could use some utility measures to sort of get an indication of what I was doing was correct, but I wasn't really sure.

Q.Sometimes when you know the theory, you know what is correct and sometimes the application itself guides or lets you know what is correct. In terms of ARX, what pointed you in the right direction?

A. I think mostly the theory pointed me in the right direction. The user interface was a little confusing. For example, if you take baseline accuracy, I used multiple configurations of k-anonymity and the accuracy would either jump up or go down that didn't give me any indication of what exactly was happening. So, I had to look up the theory of how the different values of 'k' were impacting the accuracy.

Q.Who do you think this application is more suited to?

A. It is definitely for the experts because if you know very little about anonymization and generalisation then I think it will be very very difficult to get into the application. For a lot of the definitions and explanation about the system itself, I had to refer the documentation of ARX. And even after reading the documentation, it was still difficult to comprehend some of the definitions and functions. That's why I think ARX is designed more for the experts instead of entry-level users.

Q.Can you compare ARX with other desktop applications that you have used, for example, MATLAB or Adobe Photoshop?

A. I think it is pretty similar in how difficult it is to use. if you use photoshop without any knowledge then it will be difficult to use and I felt the same way about ARX. I don't think it was very easy to get into without any knowledge.

Usability Testing: The first thing I notice immediately is that on the top left there are 20 items that are only icons and I have to hover over them to see what exactly they do. There is a lot of space below that is unused which could have the explanation or description of the icons. As for the data transformation screen on the right side is big and could be used for something better maybe. On 'Explore Results,' tab there is an enormous graph with green dots. In my opinion, this image that takes up a lot of space does not give me any information. This also applies to other tabs. The properties in the 'Analyse Utility and Risk' tabs are crammed together. I have to scroll through a lot of information that I might find interesting, but it all so small on the bottom of the screen. There is a lot of space that is lost that could be used for something. I also noticed that whenever I changed the value of K-anonymity or some configuration the graph for population unique always stayed empty. Some attributes are unclear what they are used for. I have I looked them up but I only got general definitions about them which were not helpful. I think one of the most important things is that there

should be a clear explanation for a lot of the variables and definitions in the system itself. For example, in analyse utility you have some properties with percentages, but the application does tell you what it is and what it's for. Sometimes I have to dive into the documentation to find that out and sometimes I can't even find out what that is. There are many scores and numbers, but it is not very well explained in the system. What I think is okay about the system is the way the data is shown in the configuration transformation. I have a clear view of what data I am seeing. I also think in 'analyse risk' the attacker models and the percentages look pretty good. I can clearly see the attacker success rate and risk.

Interview 4

Q.Can you state your age?

A. I am 42 years old.

Q.Can you tell me a bit about your academic background?

A. I have a PhD in computer science and I did my bachelors and masters in computer science

Q.How did you come in contact with ARX?

A. I came into contact with ARX because of my research.

Q.How did you learn to use it?

A. I learnt to use the tool by referring the documentation and scientific papers and then exploring it by hit and trial. Scientific papers on ARX helped me to understand the different functionalities.

Q.Do you trust the application forgetting desired results?

A. The application itself is robust and has enough supporting literature to get the desired results.

Q.How much time did it take for you to learn ARX?

A. It can be said that it took me 2-3 days to understand how basic functions work in the application. But it is difficult to quantify this as first you need to know what exactly is your end goal regarding the usage of the application. I experimented both with the GUI and APIs. The GUI has many parameters which are not the case in the APIs.

Q.If you had to recommend this application to new users, what advice would you give him/her?

A. New users should know what their goal and aim regarding the application is. My feedback is to first understand the target users and categorise them as basic/normal users and expert users before advising them on how to use the tool. The tool is complex and is definitely made for experts. I also interacted with the developers of ARX and they admitted the tool is made for experts. This is the reason why most of the information is not given on the GUI as it is assumed the user knows about the parameters and their definitions.

Q.Are you aware of other SDC tools?

A. I am aware of other SDC tools and have read reviews about them. There might be simpler tools out there, other than ARX but they might be limited in their functionality.

Usability testing: The starting screen has a lot of parameters. For a beginner, it might be difficult to understand the first step like choosing the privacy model – what they are

and what they do. In the 'Explore Results' tab, the lattice is for experts, new users might understand what the numbers or colours denote. In the subsequent tabs (Analyse Utility and Risk) there are a lot of parameters but not much information about them is given for which good documentation is needed especially for new users. Some graphs/numbers are not clear and do not know what they are for. 60% of the parameters or graphs can be removed for non-experts. For many new users, the user interface could be simplified but you cannot use this tool without any prior knowledge in statistics or some theory. Some basic assumptions of the new users should be made before designing a tool.

Interview 5

Q.Can you state your age?

A. I am 28 years old

Q.Can you tell me a bit about your academic background?

A. I am currently studying in Breda, in my final year of Business IT and Management. The school is at an HBO level.

Q.How did you come in contact with ARX?

A. Though a minor course in data science. I had to do the re-sit.

Q.How did you learn to use it?

A. Mostly on my own. The teacher did explain about the program in general, sometimes with questionnaires at the end of the class. The people who attended those questionnaires said That it did not add much to the explanation of the application. There is some helpful function in the application with a short explanation about certain functions and others we had to search on the internet.

Q.How was your experience of learning the software by yourself?

A. For the purpose of the interview, I will use nice language and say that it wasn't nice. It wasn't a great experience. There was a lot of frustration in figuring it out. The software isn't that self-explanatory, at least to me. The other students had the advantage of coming from a computer science background where they have experience in programming. For them, it is just another computer language. It was quite difficult to learn it and work on the other courses. Usually, there are numerous online tutorials for such applications which you can simply refer to or copy just like that, you barely put in that much effort. But for ARX that was not the case. I had to search a lot of the functions like utility and risk measures, and what they meant. Also, there wasn't much of an explanation on what most of the buttons did. Most of the explanation in the application was not extensive. I had to search online a lot but did not receive many results on google. I ended up asking the other students and the teacher for help. Eventually, I somehow did it on my own. It cost me a lot of extra time just to do a lot of basic functions.

Q.Did you refer the user manual?

A. Yes, some bits helped. The explanation for the taxonomy tree for numbers helped. Also, there was a video on that particular taxonomy tree on the internet. But for other taxonomy tree, there was no explanation. If you use programming applications you know what most of the buttons do but this is not the case with ARX for example, the anonymization button. Most functions, buttons and measures are not explained extensively. It would have helped if some sort of steps or tutorial was shown.

Q.How much time did it take for you to learn ARX?

A. A lot. I have at least spent 30 to 40 hours on it just to get it working. Even having spent so much time learning it I don't think I am at a satisfactory level. I spent double the time learning ARX than scripting languages like R (programming language statistical computing) but my user proficiency is better in R as compared to ARX. I can probably copy the same steps I took for one data set and apply to another but if the utility or risk measure results in strange numbers then I wouldn't know what it was for.

Q.Can you tell me about your first impression of the application?

A. Old. I am used to working with application academically and recreationally. I also have an interest in technology. So, I am fairly well known with interfaces and applications. And when I look at ARX it looks like something from the 90s and that it has not been updated in a very long time. Maybe that's the reason it seems that it is not used often as there is not a lot of stuff to find on the internet about it. For example, I found some old videos and pictures of it and it doesn't seem that it changed much. Purely visually it looks like an old java programme.

Q.And how did that make you feel?

A. Discouragement. Usually, my experience with older looking software is that they are clunky to work with or that they have many bugs as they are not frequently updated. Usually when some software has a modern look to it means that it is updated and used frequently. ARX is not something most people use, it mostly for programmers or computer engineers. You can tell by looking at an application that it will be hard to use if you have a lot of buttons or pop-ups. ARX does not have a lot of pop-ups but it does have a lot of buttons. The application doesn't speak for itself or you don't get a pop-up with a small tutorial on how it works.

Q.After spending so much time learning it, did you feel you could complete your task?

A. Yes. The first time I didn't as I had to take a retest but that was because I did not understand the utility measures and what the numbers meant in the data set and also not understanding what I did. But eventually, I did, after getting an extensive explanation from the teacher. But I did have 2-3 meetings with other students to be completely sure of what I did was correct. But despite all of the effort I put into it I still got a low grade. I was not satisfied with this outcome even though I believed that the anonymized data was decent. Maybe the anonymized data could have been better, but I have no idea how to do that mainly because I still do not know what all the other utility and risk measures mean.

Q.If you had to recommend this application to someone, what advice would you give him/her?

A. Find an expert on ARX as you may not find all the information you need on the internet. An expert can explain to you what everything means and how it works step by step.

Usability Testing: On opening the application, you can see that there is a lot of clustering of functions on the top left, top right and bottom right. And overall a blank view. In other applications, the first screen directs you to the first step but ARX doesn't explain anything. I can see the help function, but I cannot use it without first loading the input data set. The application also uses a lot of power on my laptop. The help function only provides a short sentence and does not explain further which I would have liked. As a first-time user, I wouldn't know what to do. I think the developers of ARX assumed that the user would be well versed with the theory and the different terminologies in the application. I would have liked a simple tutorial. Sometimes they help function explains what the button is but not what it does. There

a lot of tabs but I have no idea what they mean or what they do. There is nothing pointing me to click on a certain button to even open a project or importing a data as a first step. You get a lot of data and options/measures to choose from the 'Analyse Utility and Risk' tabs which are explained very briefly. I feel like if I had known how to use these options, I could have anonymized my data better. I think these options are nice to have if you understand them. There is a chance of having an overload of options and information that could confuse you a bit. Maybe hiding the options at first and then expanding them when you want to use them but currently, you see them immediately. Also, the result is just a bunch of numbers. I don't know what the figure in the 'Explore Results' tab. The wording of 'explore results' seems like it is important, but the teacher told me that we will be looking at mostly utility and the risk measures. If I anonymize my data I will be tempted to look more into the 'explore results' tab just because the name implies that it is important.

The desktop application has a degree of uniformity among elements of its user interface	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application conveys its purpose and give clear user assistance in its operation	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application is responsive to user inputs or events in a meaningful way	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application provides correct results or effects	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A user can complete a specified task using the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to understand the visual content of the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user feels in control of the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to navigate the desktop application in an efficient way	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface is simplistic in design without using irrelevant elements	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface has recognizable elements and interactions that can be understood by the user	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The visual metaphors (an image used to make a comparison, for example: a trash can for the function 'delete') in the user interface are meaningful	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Appendix C: Qualitative Data Analysis

Interview 1

Data	Code
“They were mostly subjective, pretty much what you searched for you got to see only that. It made it quite hard to experiment with other stuff.”	Limited information
“There were certain moments it was tempting to switch over to the programming for it. And that is what I did with my project team. ”	Discouraged
“It was first hard to find my way through the program but once you get that working it is way faster than doing it in the GUI. But the most important part for me was just how painstaking the UI user experience was.”	Effortful
“Yes, I could because of trial and error. You could see the suppression.”	Easy visualisation
“I had issues with the GUI, not just the user experience part but also that you already need to have some knowledge about the which principles there are and the settings you can use. Basing my answer on that and also that I am a computer science student so I should be able to run the application and do something with it. So I find it hard to recommend just the GUI to someone else because the GUI is already quite hard but you have to know the theory anyhow.”	Prior knowledge needed
“I would recommend it in the sense that it is really good for all privacy measure and such, but it is quite hard to get it working initially. It would require some helping hand.”	Guidance required
“It is not like reading a book – you go from left to right or up or down. You have some kind of order you like to look at in an application, but this was not the case in the GUI. You are constantly moving between different parts of the applications. There was a bit of disconnect when you edit in one tab then the tab next to it would change.”	Non-uniform structure
“I also don’t have any experience in measuring how well privacy is. I know there are statistics and I can read them like I know what they mean but I don’t know what the impact is.”	Cannot interpret data
“The first thing that I can think of is the knob-like button in ARX which reminds me of old music player applications in which you have plug-ins.”	Outdated
“The ARX GUI is a bit more complicated in the sense that it doesn’t always do what you expect it to do.”	Inconsistent
“The first time when you open the application you get this empty screen and by that time, I was already confused about how I could add my data to it.”	Not instructive
“Making changes in one tab for sure has some kind of effect in other tabs but you shouldn’t have to go between different tabs and different views within the applications to do something.”	Switching between tabs
“It reminds me of a spaghetti that you can write in code which in this case is the GUI. The GUI works but you have to find all the features.”	Cluttered

Interview 2

Data	Code
“There weren’t any tutorials. There was the manual but that too was a bit complex. Because of the lack of information, we just tried exploring it by ourselves.”	Limited information
“The manual had some difficult words in it which you had to know before to use ARX.”	Advanced language
“And then in ARX, you had to find where everything was, but it was difficult to find.”	Confusing
“I knew what the input was and what the output should have been. So, if you know the output, you can work towards it. If you ask me something else about ARX I would probably say ‘I don’t know how to do this’.”	Easy visualisation
“It was also difficult to know all the measures (utility and risk) and what they were.”	Prior knowledge needed
“The GUI does not seem attractive to me but then I understand that it is an anonymization tool and that it does not need to be fancy. The colours are grey.”	Unattractive
“There is not much information about the buttons.”	Not self-explanatory
“My first impression was also that it was difficult.”	Difficult to use
“If I work with any software which looks attractive, I immediately feel that it is fine or a nice tool to work with. But my first impression with ARX was that this is going to be difficult and a little bit outdated. If it was a bit more attractive with a modern UI I would feel that it was fun to work with.”	Discouragement
“I saw a lot of privacy models/utility measures and I can look them up online but there wasn’t much information about it. So, it was difficult because of a lack of information.”	Limited information
“Yes, I trust the tool not because I know the privacy models behind it, but I can analyse the utility and risk. I can see the percentages of how much of the data is anonymized and how much of the data is at risk.”	Easy visualisation
“When I load my data with the hierarchies, I don’t know most of the elements on the screen. There is no function to click on them and get a text message that explains them.”	Not instructive
“I think there is a nice overview with percentages (‘Analyse Utility and Risk’) given and it doesn’t have too much information on the screen. I think it is good because if you have too much information then it could have an impact on much you could pick-up. ”	Easy visualisation

Interview 3

Data	Code
<p>“It was quite difficult. I think the tutorial was somewhat comprehensible. But if I wanted to do some more advanced steps then I had to figure it out by myself through trial and error. It could have been a smooth experience if there was some sort of in-system tutorial.”</p>	<p>Limited information</p>
<p>“The steps that I wanted to take weren’t clear from the beginning. So, there were a lot of buttons that do not have an exact description when you start the application. So, you have to search through all the stuff to find the right things you want to know.”</p>	<p>Not instructive</p>
<p>“It looks kind of old. It’s pretty minimal.”</p>	<p>Outdated</p>
<p>“For a big proportion, I had no idea if it was correct or not. I was just winging it hoping that it was correct. I could use some utility measures to sort of get an indication of what I was doing was correct, but I wasn’t really sure.”</p>	<p>Cannot interpret data</p>
<p>“I think mostly the theory pointed me in the right direction. The user interface was a little confusing.”</p>	<p>Prior knowledge needed</p>
<p>“For a lot of the definitions and explanation about the system itself, I had to refer the documentation of ARX. And even after reading the documentation, it was still difficult to comprehend some of the definitions and functions.”</p>	<p>Language not helpful</p>
<p>“The first thing I notice immediately is that on the top left there are 20 items that are only icons and I have to hover over them to see what exactly they do. There is a lot of space below that is unused which could have the explanation or description of the icons.”</p>	<p>Cluttered</p>
<p>“On ‘Explore Results,’ tab there is an enormous graph with green dots. In my opinion, this image that takes up a lot of space does not give me any information. This also applies to other tabs. The properties in the ‘Analyse Utility and Risk’ tabs are crammed together. I have to scroll through a lot of information that I might find interesting, but it all so small on the bottom of the screen.”</p>	<p>Not self-explanatory</p>
<p>“in analyse utility you have some properties with percentages, but the application does tell you what it is and what it’s for. Sometimes I have to dive into the documentation to find that out and sometimes I can’t even find out what that is. There are many scores and numbers, but it is not very well explained in the system.”</p>	<p>Not self-explanatory</p>
<p>“I can clearly see the attacker success rate and risk.”</p>	<p>Easy visualisation</p>

Interview 4

Data	Code
“Scientific papers on ARX helped him to understand the different functionalities.”	Not self-explanatory
“The application itself is robust and has enough supporting literature to get the desired results.”	Robustness
“is the reason why most of the information is not given on the GUI it is assumed the user knows about the parameters and their definitions.”	Prior knowledge needed
“The starting screen has a lot of parameters. For a beginner, it might be difficult to understand the first step like choosing the privacy model – what they are and what they do.”	Not instructive
“In the subsequent tabs (Analyse Utility and Risk) there are a lot of parameters but not much information about them is given for which good documentation is needed especially for new users. Some graphs/numbers are not clear and do not know what they are for. 60% of the parameters or graphs can be removed for non-experts.”	Limited information

Interview 5

Data	Code
“For the purpose of the interview, I will use nice language and say that it wasn’t nice. It wasn’t a great experience. There was a lot of frustration in figuring it out”	Effortful
“The software isn’t that self-explanatory, at least to me.” Not	self-explanatory
“It was quite difficult to learn it and work on the other courses.”	Effortful
“there wasn’t much of an explanation on what most of the buttons did.”	Limited information
“Most of the explanation in the application was not extensive.”	Minimal explanation
“If you use programming applications you know what most of the buttons do but this is not the case with ARX for example, the anonymization button”	Not self-explanatory
“It would have helped if some sort of steps or tutorial was shown.”	Not instructive
“Even having spent so much time learning it I don’t think I am at a satisfactory level. I spent double the time learning ARX than scripting languages like R (programming language statistical computing) but my user proficiency is better in R as compared to ARX.”	Effortful
“when I look at ARX it looks like something from the 90s and that it has not been updated in a very long time.”	Outdated
“Usually, my experience with older looking software is that they are clunky to work with or that they have many bugs as they are not frequently updated. Usually when some software has a modern look to it means that it is updated and used frequently.”	Discouraged
“An expert can explain to you what everything means and how it works step by step.”	Guidance required
“On opening the application, you can see that there is a lot of clustering of functions on the top left, top right and bottom right. And overall a blank view. In other applications, the first screen directs you to the first step but ARX doesn’t explain anything”	Cluttered
“I can see the help function, but I cannot use it without first loading the input data set.”	Inconsistent
“Sometimes they help function explains what the button is but not what it does. ”	Language not helpful
“There is nothing pointing me to click on a certain button to even open a project or importing a data as a first step.”	Not instructive
“I feel like if I had known how to use these options, I could have anonymized my data better. I think these options are nice to have if you understand them.”	Cannot interpret data

Appendix D: Quantitative Data Analysis

*to 3 decimal places, sample size = 7									
S.No	Question	Strongly Disagree = 1	Disagree = 2	Neutral = 3	Agree = 4	Strongly Agree = 5	Mean*	Variance*	SD*
1	The desktop application is attractive to the user, in terms of its colour or graphical user interface	1	3	2		1	18/7 = 2.571	1.387	1.178
2	The user likes the desktop application	1	3	2	1		17/7 = 2.428	0.816	0.903
3	The user interface can be customised to the user's personal preferences	4	2	1			11/7 = 1.571	0.530	0.728
4	The desktop application helps the user achieve their task in minimal number of steps	1	4	2			15/7 = 2.14	0.408	0.638
5	The amount of effort to operate the desktop application is low	3	3		1		13/7 = 1.857	0.979	0.989
6	The user is required to keep minimal amount of information in mind in order to achieve a specified task	1	6				13/7 = 1.857	0.122	0.349
7	The user interface provides context-sensitive help and meaningful feedback when errors occur	3	4				11/7 = 1.571	0.244	0.494
8	The desktop application has a degree of uniformity among elements of its user interface	1	2	1	2	1	21/7 = 3	1.714	1.309
9	The desktop application conveys its purpose and gives clear user assistance in its operation	2	4	1			13/7 = 1.857	0.408	0.638
10	The desktop application is responsive to user inputs or events in a meaningful way	1	3	1	1	1	19/7 = 2.714	1.632	1.277
11	The desktop application provides correct results or effects		1	1	2	3	28/7 = 4	1.142	1.069
12	A user can complete a specified task using the desktop application		1		5	1	27/7 = 3.857	0.693	0.832
13	It is easy to understand the visual content of the desktop application	2	3	1	1		15/7 = 2.142	0.979	0.989
14	The user feels in control of the desktop application	1	3	1	2		18/7 = 2.571	1.102	1.049
15	It is easy to navigate the desktop application in an efficient way	2	2	1	2		17/7 = 2.428	1.387	1.178
16	The user interface is simplistic in design without using irrelevant elements	1	3	1	2		18/7 = 2.571	1.102	1.049
17	The user interface has recognizable elements and interactions that can be understood by the user		5	2			16/7 = 2.285	0.204	0.451
18	The visual metaphors (an image used to make a comparison, for example: a trash can for the function 'delete') in the user interface are meaningful	1	2	2	2		19/7 = 2.714	1.061	1.030

Appendix E: Prototype Screens

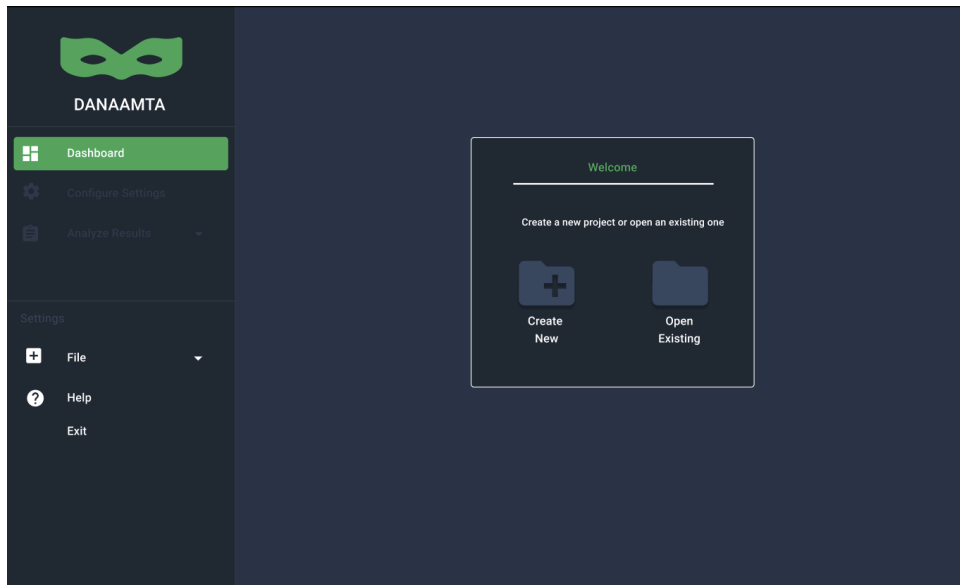


Figure A.1: Screen on startup, part 1

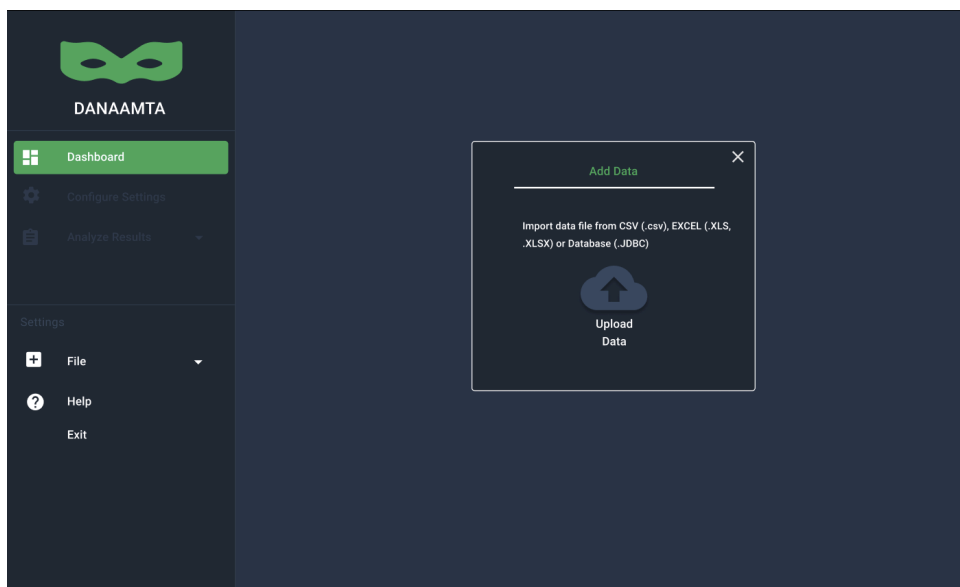
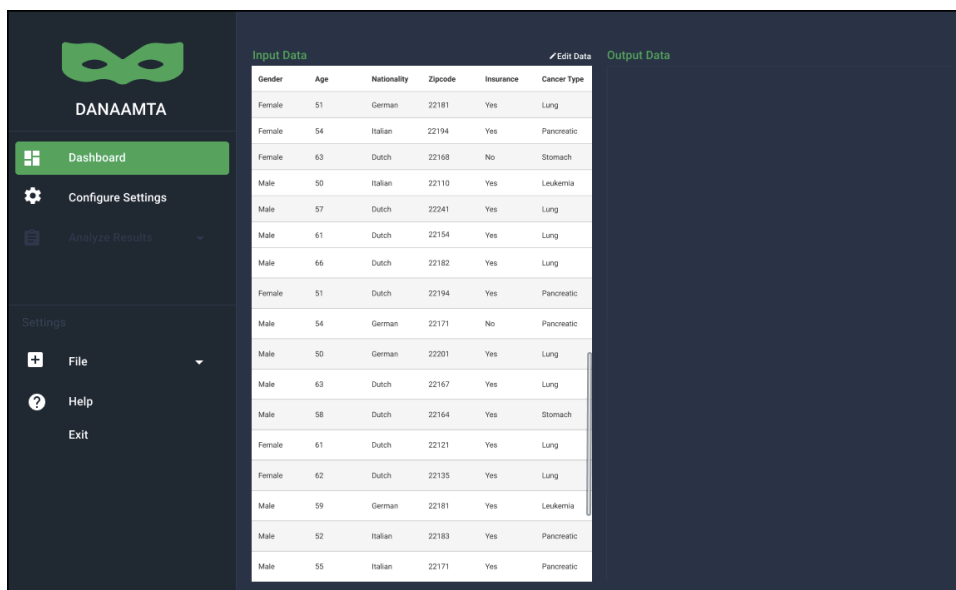


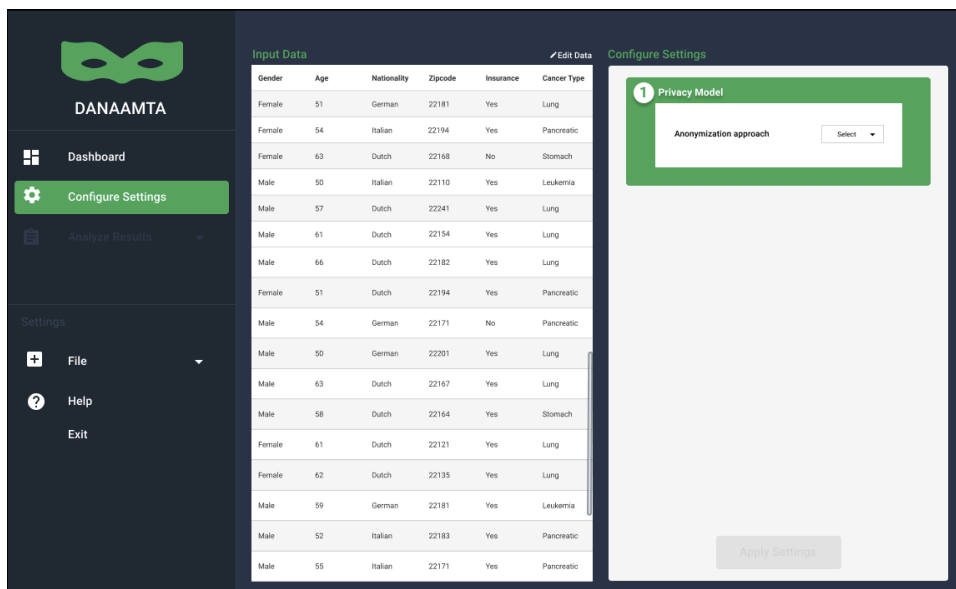
Figure A.2: Screen on startup, part 2



The screenshot shows the DANAAMTA dashboard interface. On the left is a dark sidebar with the DANAAMTA logo and navigation options: Dashboard (highlighted), Configure Settings, Analyze Results, and Settings (with sub-options: File, Help, Exit). The main area is split into two panels. The left panel, titled 'Input Data', contains a table with 20 rows of patient data. The right panel, titled 'Output Data', is currently empty.

Gender	Age	Nationality	Zipcode	Insurance	Cancer Type
Female	51	German	22181	Yes	Lung
Female	54	Italian	22194	Yes	Pancreatic
Female	63	Dutch	22168	No	Stomach
Male	50	Italian	22110	Yes	Leukemia
Male	57	Dutch	22241	Yes	Lung
Male	61	Dutch	22154	Yes	Lung
Male	66	Dutch	22182	Yes	Lung
Female	51	Dutch	22194	Yes	Pancreatic
Male	54	German	22171	No	Pancreatic
Male	50	German	22201	Yes	Lung
Male	63	Dutch	22167	Yes	Lung
Male	58	Dutch	22164	Yes	Stomach
Female	61	Dutch	22121	Yes	Lung
Female	62	Dutch	22135	Yes	Lung
Male	59	German	22181	Yes	Leukemia
Male	52	Italian	22183	Yes	Pancreatic
Male	55	Italian	22171	Yes	Pancreatic

Figure A.3: Raw data is uploaded on the dashboard



The screenshot shows the DANAAMTA dashboard with the 'Configure Settings' page active. The sidebar is the same as in Figure A.3. The 'Input Data' table is visible on the left. The 'Configure Settings' panel on the right features a 'Privacy Model' section with a sub-section 'Anonymization approach' containing a 'Select' dropdown menu. An 'Apply Settings' button is located at the bottom of the settings panel.

Figure A.4: Configure settings page

Input Data

Gender	Age	Nationality	Zipcode	Insurance	Cancer Type
Female	51	German	22181	Yes	Lung
Female	54	Italian	22194	Yes	Pancreatic
Female	63	Dutch	22168	No	Stomach
Male	50	Italian	22110	Yes	Leukemia
Male	57	Dutch	22241	Yes	Lung
Male	61	Dutch	22154	Yes	Lung
Male	66	Dutch	22182	Yes	Lung
Female	51	Dutch	22194	Yes	Pancreatic
Male	54	German	22171	No	Pancreatic
Male	50	German	22201	Yes	Lung
Male	63	Dutch	22167	Yes	Lung
Male	58	Dutch	22164	Yes	Stomach
Female	61	Dutch	22121	Yes	Lung
Female	62	Dutch	22135	Yes	Lung
Male	59	German	22181	Yes	Leukemia
Male	52	Italian	22183	Yes	Pancreatic
Male	55	Italian	22171	Yes	Pancreatic

Configure Settings

1 Privacy Model
Anonymization approach: k-anonymity
Value of k: 4

2 Attribute Mapping
Non-sensitive: (empty)
Sensitive*: Insurance, Cancer Type
Quasi-identifying: Gender, Age, Nationality, Zipcode

3 General Settings
Suppression limit: 100%

Apply Settings

Figure A.5: Specific configurations are loaded for k-anonymity

Input Data

Gender	Age	Nationality	Zipcode	Insurance	Cancer Type
Female	51	German	22181	Yes	Lung
Female	54	Italian	22194	Yes	Pancreatic
Female	63	Dutch	22168	No	Stomach
Male	50	Italian	22110	Yes	Leukemia
Male	57	Dutch	22241	Yes	Lung
Male	61	Dutch	22154	Yes	Lung
Male	66	Dutch	22182	Yes	Lung
Female	51	Dutch	22194	Yes	Pancreatic
Male	54	German	22171	No	Pancreatic
Male	50	German	22201	Yes	Lung
Male	63	Dutch	22167	Yes	Lung
Male	58	Dutch	22164	Yes	Stomach
Female	61	Dutch	22121	Yes	Lung
Female	62	Dutch	22135	Yes	Lung
Male	59	German	22181	Yes	Leukemia
Male	52	Italian	22183	Yes	Pancreatic
Male	55	Italian	22171	Yes	Pancreatic

Output Data

Gender	Age	Nationality	Zipcode	Insurance	Cancer Type
Female	[50, 55]	German	22***	Yes	Lung
Female	[50, 55]	Italian	22***	Yes	Pancreatic
Female	[61, 65]	Dutch	22***	Yes	Lung
Female	[50, 55]	Italian	22***	Yes	Leukemia
Female	[56, 60]	Dutch	22***	Yes	Lung
Female	[61, 65]	Dutch	22***	No	Stomach
Male	[66, 70]	Dutch	22***	Yes	Leukemia
Male	[50, 55]	Dutch	22***	Yes	Lung
Male	[50, 54]	German	22***	Yes	Lung
Male	[50, 55]	German	22***	Yes	Lung
Male	[61, 65]	Dutch	22***	Yes	Pancreatic
Male	[50, 55]	Dutch	22***	Yes	Lung
Male	[61, 65]	Dutch	22***	No	Lung
Male	[61, 65]	Dutch	22***	Yes	Stomach
Male	[56, 60]	German	22***	Yes	Leukemia
Male	[50, 55]	Italian	22***	Yes	Pancreatic
Male	[56, 60]	Italian	22***	Yes	Pancreatic

Figure A.6: Anonymized data is displayed on the dashboard

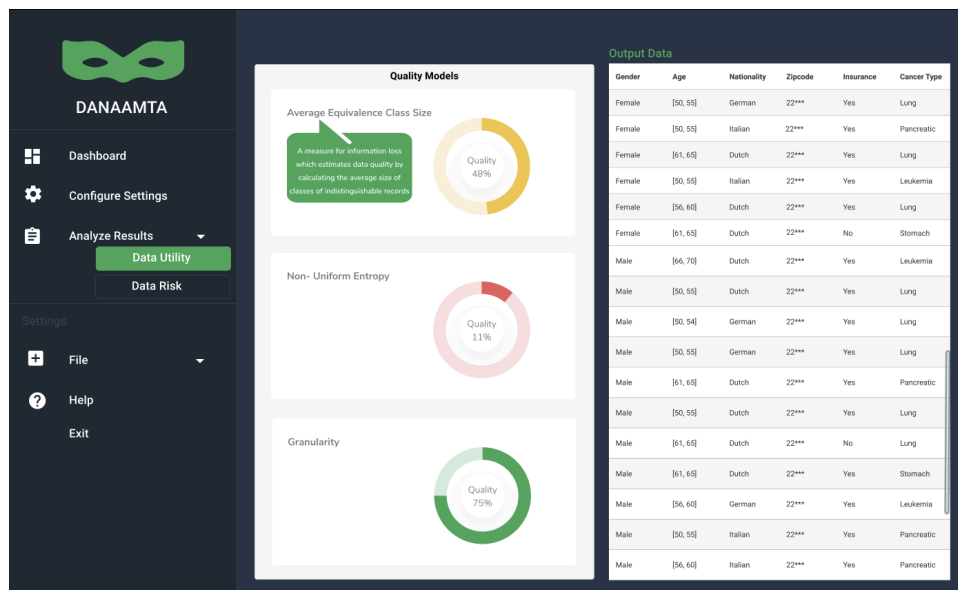


Figure A.7: Analyse data utility page

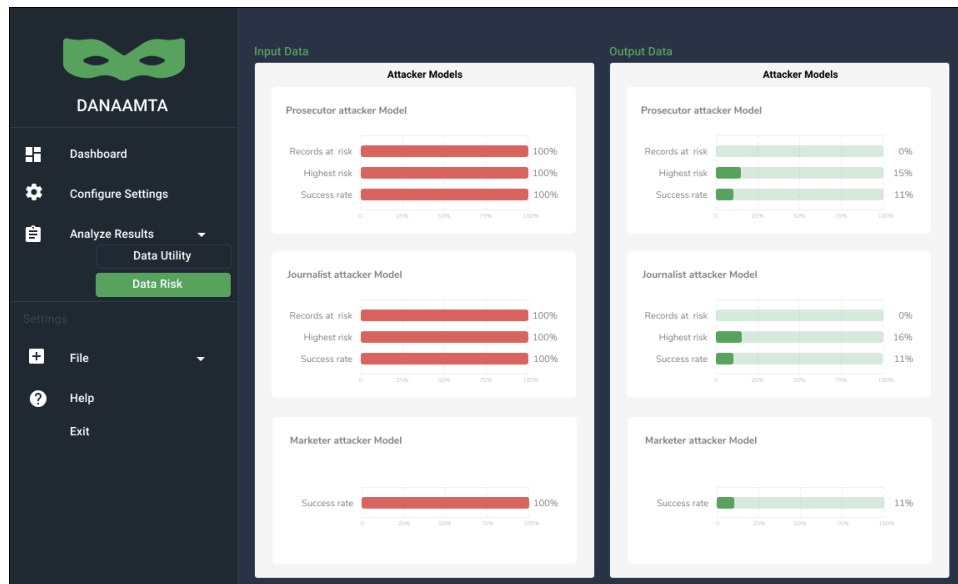


Figure A.8: Analyse data risk page

Appendix F: Evaluation

Part 1 of evaluation

Part 1: Survey to analyze the usability of ARX

This is an evaluation study conducted as part of a Master Thesis by a student of TU Delft. The survey is motivated by the Quality in Use Integrated Map for analysing software applications. This survey is anonymous. All questions are mandatory. Please answer them as honestly as possible.

*Vereist

Please indicate how much you agree or disagree with the following statements regarding the ARX desktop application. *

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The desktop application is attractive to the user, in terms of its color or graphical user interface	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The application reduces some complexity of the data anonymization theories	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The application can be used by entry-level users	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The amount of effort to operate the desktop application is low	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application conveys its purpose and give clear user assistance in its operation	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface provides context-sensitive help and meaningful feedback	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface has recognizable elements and interactions that can be understood by the user	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to navigate the desktop application in an efficient way	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

The user is required to keep minimal amount of information in mind in order to achieve a specified task	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to analyse the utility of the data	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to analyse the data risk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to understand the visual content of the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface is simplistic in design without using irrelevant elements	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application helps the user achieve their task in minimal number of steps	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A user can complete a specified task using the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Submit

Part 2 of evaluation

Part 2: Survey to analyze the usability of Danaamta

This is an evaluation study conducted as part of a Master Thesis by a student of TU Delft. The survey is motivated by the Quality in Use Integrated Map for analysing software applications. This survey is anonymous. All questions are mandatory. Please answer them as honestly as possible.

* Required

Please indicate how much you agree or disagree with the following statements regarding the DANAAMTA desktop application (prototype). *

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
The desktop application is attractive to the user, in terms of its color or graphical user interface	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The application reduces some complexity of the data anonymization theories	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The application can be used by entry-level users	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The amount of effort to operate the desktop application is low	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application conveys its purpose and give clear user assistance in its operation	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface provides context-sensitive help and meaningful feedback	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface has recognizable elements and interactions that can be understood by the user	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to navigate the desktop application in an efficient way	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

The user is required to keep minimal amount of information in mind in order to achieve a specified task	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to analyse the utility of the data	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to analyse the data risk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
It is easy to understand the visual content of the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The user interface is simplistic in design without using irrelevant elements	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The desktop application helps the user achieve their task in minimal number of steps	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A user can complete a specified task using the desktop application	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Share your opinions on the prototype. Any feedback/comments would be appreciated. *

Your answer

Appendix G: Email for contacting research participants

1. For the investigation of the usability of ARX

Hi xxxxx,

Hope this email finds you in good health!

I was informed by xxxxx that you would be willing to provide your insights for my master thesis research. I hereby thank you for helping out a fellow student and extend this invitation for informing you about the interview process.

The purpose of the interviews is to understand the usability of ARX from the perspective of first-time users and their experiences with this tool. The interview will be held virtually through a video call (Skype) meeting. The interview questions will range from open-ended questions to more structured questions and the whole process should not last more than 30 to 45 minutes. Please be rest assured the interviews are completely anonymous and you are not obligated to answer all questions.

This interview is NOT a test of your knowledge on SDC or ARX. So absolutely no preparation is needed for this interview! I would only ask you to relate your interactions/experiences with the ARX tool.

Lastly, as we are all practicing social distancing, I cannot ask someone to take down notes as I conduct the interviews. For this reason, during the interview I will ask your permission to audio record some of your answers to the questions. I want to keep this process as transparent and comfortable for everyone as possible.

If you agree with the above process, let us set a date for the interview! Please let me know out of the below options which time and day suits you best:

XXX XXX XXX

XXX XXX XXX

XXX XXX XXX

If you have any questions for me, please do not hesitate to send me an email. Wishing you a good day ahead!

Warm regards,
Anshika Rawat
Management of Technology
TU Delft

2. For the evaluation of Danaamta

Hi xxxx,

Hope you are doing well.

Last month you helped me investigate the data anonymization tool ARX by relating your experiences and the challenges that you faced while using it. As part of this master thesis, I have tried to address some of these challenges in a prototype of a data anonymization tool called Danaamta and now require your assistance to evaluate it.

Please know that Danaamta is not to be taken as a replacement for ARX but as a simplified alternative to do some necessary data anonymisation techniques. The aim of Danaamta is to reduce the complexity associated with ARX by providing a uniform, non-expert approach to anonymize a data set in minimal steps.

Getting that out of the way, let's begin with how you can proceed to help in the final phase of my thesis.

1. Please begin by first filling out this survey on ARX. Why another survey? Well, the previous interview was more of an exploratory study. This survey focuses only on those aspects that were found to be common in ARX across majority of the participants.
2. Next, please refer the two slides (Attached pdf). The first slide provides a general description of the functioning of the prototype. To not induce any biases, the prototype design decisions are not explained in detail and only information concerning on how to navigate the prototype is provided. For the purpose of simulation, an example of k-anonymity and l-diversity is used to illustrate the functioning of the prototype. Follow the link of the prototype on slide 2.
3. Lastly, please fill out this survey evaluating the prototype.

I would greatly appreciate it if you could complete the evaluation before 22nd June (Monday). In case, you have any doubts or need help during the evaluations you can email me here or ping me on skype to chat (skype username: xxxx).

Once again, I thank you for your help in bringing me one step closer to completing my thesis.

Warm regards,
Anshika Rawat
Management of Technology
TU Delft

3. Attached Instructions

Prototype Description

1. A basic welcome screen greets the user and points to the first step needed to start the anonymization process
2. After uploading the data, it appears on the dashboard screen where an edit option gives the choice to the user to remove certain columns in the data set. These columns can be direct identifiers.
3. Notice, the option configure settings is enabled
4. Configure setting is selective action based display. Meaning depending on the selection of the privacy model, context-related settings will be displayed.
5. In attribute mapping, the option 'optimize' refers to creating hierarchies (not simulated)
6. After configuring the setting, the analyse results option will be enabled.
7. You can hover on some items to get a description of their functionality (only some examples have been simulated)
8. Options like 'help' is designed to provide the user with a manual or general settings of the application where the hover function description can be disabled, etc.
9. If you do not know what is clickable or simulated in the prototype per screen, just click outside the application window to see prompts of what is clickable/simulated
10. The prototype follows one basic flow to get to all the screens and ends at analysing the results
11. If you wish to re-start the simulation, you will find a 'restart' option at the bottom right corner of the screen.
12. Lastly, please zoom in on the application screen to read some of the text as the prototype is configured to a screen dimension of 1152x700

Links

- **Prototype:**

<https://www.figma.com/proto/NFga12tiWWW0XzSQ3S3uKv/DANAAMTA?node-id=2%3A418&viewport=430%2C255%2C0.13771803677082062&scaling=min-zoom>