A Usability Evaluation and Re-design of the password manager software KeePass2

Harris Flourentzos

Master of Science
School of Informatics
University of Edinburgh

2018

Abstract

Despite their security and usability flaws, passwords still remain the most widely adopted user authentication system used on the web today, mainly due to the high deployability benefits they offer. Due to the rapid and continuous increase of the numbers of online accounts an average user needs to manage however, memorability becomes a major weakness for vanilla passwords which is met by lowering their strength and uniqueness by the end-users. To help mitigate the problem, password managers were introduced to provide a partial solution. These software systems hope to maintain the deployability attributes of passwords while eliminating their memorability burden. However password managers do not come without their own problems.

This study has focused in evaluating the usability of a password manager application called KeePass2 through a series of usability inspection and lab-study methods. During the study several usability issues were identified and a re-design of a revised version of KeePass2 was developed to provide solutions to those issues. The developed solution was then further evaluated through a usability lab study to identify any remaining issues.

Acknowledgements

First and foremost I want to thank my parents for their constant emotional support through out the whole year and duration of the project.

Special thanks to my two brothers – I promise you won't have to go through any more of my mock-ups.

Starbucks probably wants to thank me for the numerous filter coffees I bought to get through the project.

Finally, I want to thank my supervisor Dr. Kami Vaniea for the support through out the project and wish her a happy married life.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Although many of us dont yet realize, our digital selves are already well established and alive in the world wide web. Scattered across several social media profiles, bank accounts, web browser histories and cookies, they are undoubtedly an ever-increasing part of our identities and everyday lives. Yet, the primary tool users are equipped with to ensure their security and privacy is the password, despite its numerous security and usability flaws.

To use any security system/tool correctly, users are asked to follow complex rules called security policies. These rules deplete users of mental effort [Herley, 2009, Schneier, 2008] ,of which they have limited reserves of, and so they are usually ignored or bypassed by users. This behaviour results in erroneous usage of security systems and so users are considered to be the weakest link in the security chain.

The same is true for password authentication systems. Users are asked to follow password creation policies which are cumbersome to abide to given the increasing numbers of online accounts an average user needs to manage. A proposed solution to this problem is to limit these policies and replace passwords with other authentication systems. These systems will be effortless to use and will depend on user's security knowledge as less as possible. However, this has been proven to be a troublesome task, since simply none of the suggested alternatives, be it password managers, dual device authentication, biometrics, visual cryptography to name a few, provide complete superiority to password usage from a usability and security perspective [Bonneau et al., 2012].

One technique that seems to at least mediate the burden placed on users are password managers (PMs). Although not perfect, it is believed that with the correct usability focused design of a password manager, a reasonable security standard could be achieved while keeping the effort invested by users to the minimum. In a recent study [Reeder et al., 2017] to outline the most important security advice to end-users, *"Use* *a password manager"* was one of the most common advice given by security experts. Therefore, the focus of this study is to evaluate and re-design an existing password manager software called KeePass2 [https://keepass.info/].

1.2 Aim

As briefly discussed in the motivation section, research [Adams and Sasse, 1999][Herley, 2009] suggests that the responsibility should be shifted from the users to the designers of security systems. Software engineers should design systems that abstract as many security policies as possible from the user to remember and understand. They should make sure that the mental model that their system imposes on the user is the correct one. And finally make the use of the system as effortless as possible.

Research Questions
In that, our study will aim to provide answers to the following research questions:

- Was KeePass2 designed and developed as suggested above?
- Is KeePass2 usable for the average user at its current state?
- If not, what are the key features or attributes that make it unusable for the average user?

Contribution
The contribution of this study is:

- It uncovers major usability issues that KeePass2 has in its current design and the security risks that those issues expose users to.
- It remedied some of the issues identified by designing a revised version of KeePass2.
- Provide guidance to future security software engineers as to what to consider while creating new security software.

1.3 Results

The main findings from this study are that the original version of KeePass2 password manager (PM) has basic usability issues (some of which are the use of bad visual metaphors, complex work flow and inaccurate mental model induction to the user) for the average user and so a complete re-design of a new version was required, which is called the "revised version" through out the study. The revised version was designed and tested with usability lab studies and was found to have remedied some of the issues identified in the original.

1.4 Outline

In the following chapters we expand on topics mentioned in the introduction and present the process of evaluating and developing a revised version of KeePass2. Specifically:

- *CHAPTER 2* will provide the reader with background knowledge he will need to comprehend this study
- *CHAPTER 3* and *CHAPTER 4* describe and analyze the 2 usability evaluation studies (a Cognitive Walkthrough study and a Think-Aloud study) performed on the original version of KeePass2 as part of the requirements gathering process.
- *CHAPTER 5* describes the design process and design decisions taken as well as the developments process to design the revised version of KeePass2.
- *CHAPTER 6* provides the results of the usability study (a Think-Aloud study) performed to evaluate the revised version.
- *CHAPTER 7* provides a discussion of the results and draws useful conclusions. It finally provides the reader with the limitations and future work related to this project.

Chapter 2

Background

2.1 Related Work

Usable Security is a multidisciplinary field and so this project has various sources of related work. Studies like [Whitten and Tygar, 1999] that evaluated the usability of a security program, PGP 5.0, have concluded that usability principles should be extended beyond the ones commonly used for generic user interface evaluation and design. They claim that security as a field has a series of properties that introduce extra difficulties in the design of secure and usable systems. A summary of those properties is outlined in the Appendix C.1

Another source of related work is studies like [Florencio and Herley, 2007], [Ur et al., 2012] that have focused to collect/uncover password user habits and behaviours.
[Florencio and Herley, 2007] aimed to collect password creation habits of average users while [Ur et al., 2012] focused on the effects of password meters. These are discussed further in the password habits subsection of this chapter.

More relevant to this study are usability evaluation studies that focused specifically on password manager (PM) software, like [McCarney et al., 2012, Chiasson et al., 2006, Karole et al., 2011]. [McCarney et al., 2012] designed a PM application, named Tapas, that works with dual device authentication instead of a master password to encrypt the database. Their study aimed to alleviate the users from the burden of memorability completely while keeping a good usability standard as other popular PM applications that they used to compare Tapas with. Tapas was evaluated with a usability lab study as well as with a framework developed by [Bonneau et al., 2012] which aims to assess security systems in terms of Security, Usability and Deployability. [Chiasson et al., 2006] performed a usability evaluation on two PM applications PwdHash and Password Multiplier. They discovered that while using those 2 PMs, the participants of the study failed to securely complete the different security related tasks they were assigned. They observed that the main reason for this failure was that users had inaccurate or incomplete mental models of the security system and that those inaccurate mental models were induced due to the current design of the respective systems. Furthermore they found that users were reluctant to trust a password manager with their passwords. The results of this study are very related to the results of our own study that has observed similar behaviours from the study participants.[Karole et al., 2011] performed a comparative usability evaluation study on 3 different types of password managers, an online-based (LastPass) PM, a smart-phone-based (KeePassMobile) PM and a USB-based (Roboform2Go) one. Their main findings included that non-tech users preferred the 2 portable solutions (Mobile,USB) rather than the more usable online solution.

Although not a usability user study, [Bonneau et al., 2012] followed the suggestion of [Biddle et al., 2012], to produce a systematic evaluation and design scheme to compare various types of authentication systems. They developed the UDS framework which included 25 Usability, Deployability and Security attributes the ideal system should possess. In the same study they used UDS to evaluate a collection of different user authentication systems including popular PMs. Their main finding was that there was no better alternative system to replace password and PM systems.

It should be noted that, to the knowledge of the writer, there has been no official usability studies on KeePass2. Nonetheless, [Gasti and Rasmussen, 2012] have compared and analyzed the security provided by the most popular password manager database formats. On of them was the .kdbx format used by KeePass2. Their general conclusion was that AES-CBC encryption was not enough to provide complete security to most of the database formats. Interestingly they claim that .kdbx is not secured when stored in a cloud based storage directory. Their finding are well beyond the scope of this project but they were non the less interesting since a lot of people seem to be using KeePass2 by uploading their database files in the cloud.

2.2 Usable Security

The field of Usable security (and Privacy) is a relatively young and multidisciplinary field, drawing from the fields of Security(and Privacy), Human Computer Interaction (HCI) and Psychology. The main reason for the existence of this field is due to the fact that security systems, however robust they are designed to be from a cryptography perspective, will eventually be used and applied by end-users. This fact becomes troublesome combined with a property of security called the weakest link property, which simply states that an attacker can compromise a security system by attacking its weakest aspect, and no matter how strong the rest of the design is, the attack will be successful due to that weakest point. The weakest point of security systems that will eventually be used by users, is no other but the user himself, and so if the user operates the security system in the wrong way then the system will fail.

What causes a user to operate a security system in an erroneous manner is his **1) lack of security knowledge** and **2) lack of motivation** when the system he is asked to use is complex and cumbersome. In that, usable security tries to find better strategies and designs that will improve security systems helping users overcome those 2 causes of erroneous behaviour. In the following sections we outline why users are unmotivated to use security systems and that the best strategy for success is to design systems that motivate users rather than trying to force security knowledge on them.

2.2.1 Psychology of Security

Crucial to designing usable security systems is to understand why users would ignore or fail to follow the intended usage of these systems. [Herley, 2009, Adams and Sasse, 1999, Schneier, 2008] are among the most influential studies that tried to answer this question.

[Schneier, 2008] argued that users behaviour and decisions is guided by their perception of risk and their behaviour in security is no exception. They state that users perceive security related tasks as secondary to the objectives they are really after, and try to minimize the cognitive efforts they invest in completing that secondary task. This cognitive effort is perceived as an immediate cost the users have to pay. On the contrary the benefit from staying secure is not at all visible most of the times and neither is the feedback (and reward) the user attains when he has completed a security task. So when the users weigh the cost/reward they receive by complying to security systems the real and immediate cost outweighs the hypothetical and implicit reward (or consequence).

Similarly, [Herley, 2009] suggests that users turn down and ignore security advice because the cost that burdens the user with is ongoing (users continually need to read and update their knowledge) where as the cost of an attack is a one time event or sometimes non-existent. They suggest that we should start minimizing the education of users and start increasing the education of security system designers to produce systems less cumbersome for users to use.

2.2.2 Psychology of Passwords

Password User Habits

[Florencio and Herley, 2007] conducted a large scale study that involved more than half a million users to uncover and document the password creating and management habits of an average web user. They discovered that an average web user has more than 25 online accounts on average. To protect those 25 accounts, users created and managed 6.5 unique passwords. These passwords were shared across 3.9 different websites on average and they were estimated to have an average quality (measured in bit-strength) of 40.53 bits. The study also found that most users tended to create longer, lower-case passwords rather than using upper-case letters and special charac-

ters to improve their password strength.
Their work also revealed that users would forget their passwords quite often since 4.28% of Yahoo users forgot their passwords over a three month period. Finally the study pointed out that some users are exposed to online phishing attacks which accounted for 0.4% of the total user population per year.

As mentioned briefly in the introduction, many online services and organizations try to guide and/or restrict users in selecting stronger passwords with providing password meters and restrictive policies(for example disallowing users to choose less than 8 character passwords or forcing them to choose special characters and Capital letters). It's not uncommon however to observe users ignoring or working around those strategies. [Ur et al., 2012] conducted a study with the aim to study the effects of such visual meters and restrictive policies on the average user. They discovered that, services that deployed visual meters to indicate password strength had significantly increased the strength of user chosen passwords. However their results revealed that such an increase was not enough to make a difference against offline brute-force attacks. During the study they tested various designs of such visual meters and found that their difference in design had little or no effect whatsoever. On the other hand, they discovered that stringent policies (for example use of all ASCII character types) did produce a significant increase in strength of user chosen passwords along with resistance to brute-forced attacks, but resulted in frustrating users.

Many more studies [Burr et al., 2004, Shay et al., 2010, Veras et al., 2014] have confirmed the frustration of users and its effects on password creation. They found that users would fulfill policy requirements in predictable ways such as use only a small fraction of the symbols on a keyboard, choose semantically meaningful passwords and password-phrases that follow grammatical rules.

Finally another study on password-creation policies [Shay et al., 2016] found that the usual comp8 is very susceptible to both online and offline attacks and should be replaced with more usable and secure alternatives, like the 2word16 or 2class12.

2.3 Passwords

2.3.1 Password Strength

Password strength is usually measured in terms of the number of guesses an attacker needs to iterate through in order to succeed in a brute force attack. A more formal measure is information entropy which is equal to the base-2 log of the number of guesses for a successful brute force attack and is its units are bits. Entropy of a password can be calculated by:

$$H = log_2 N^L \tag{2.1}$$

where N is the number of possible symbols in the password, N is the number of characters of the password.

For example, a 42 bit random password would require a maximum of 2^{42} (4,398,046,511,104) attempts to crack during an offline attack. Adding an extra bit (not character) to this password would require the attacker to invest double the attempts in order to succeed his offline attack. As the reader might be able to guess by now, this is under the assumption the password is chosen randomly, which is not the case with user chosen passwords.

[Florêncio et al., 2016] suggest that 10^6 is a reasonable estimate of the number of guesses an adversary can perform in order to gain access to password protected accounts in an online attack. In other words if he is unable to hack the account after he had exhausted 10^6 the account is safe. Similarly 10^{14} guesses is the estimate for an offline attack.

2.3.2 Password Storage

Hash Functions:
Storing user passwords and other sensitive credential information in online databases or in password managers, definitely requires encryption. This is done with the use of, a **cryptographic hash function** that takes as input data of arbitrary length and maps them to a fixed-length string. Any changes in the input message will result in a different output string. A hash function has also the following properties:

- It is a `one-way` function. This means that it is easy and straight forward to compute but extremely difficult/impossible to invert. In other words if an attacker is able to steal its output he will never be able to recover the original string.
- It is `collision resistant`, which implies that it is extremely rare for two different messages to map to the same hash function output.

So the credentials of a user are firstly hashed to produce the hashed value which is then stored in a database. The authentication is then accomplished by comparing the hashed value stored in the database with the calculated hashed value from the user's submitted password, which alleviates the need of storing the actual password string the user has chosen.

Salting:
To defend against dictionary and pre-computed rainbow table attacks (which is the hashed equivalent of a dictionary attack) plain text passwords are salted right before they are hashed. Salting, in cryptography, is the process of appending random data, called the salt, to a plain text password and then calculating its hash value.

The process of Salting involves 1) the random generation of a new salt for each password, 2) the concatenation of the salt and the password, 3) the hashing of the concatenated string by a cryptographic hash function and finally 4) the storage of both the salt and the resulting hash value into a database.

Using salts, massively increases the size of rainbow tables and allows the the usage of identical passwords to be used by different users.

2.3.3 Attacks on Passwords

There exist numerous strategies and types of attacks deployed against password authentication systems. In the following sections we outline the most commonly used attacks and the most relevant to our study. We can group these attacks into 2 types under the context of this study. Direct attacks and attacks through users. The former aim to gain access to sensitive information by guessing the password and the latter aim to manipulate the user to reveal the password to the attacker.

Direct attacks:
The most common types of direct attacks are:

- **Brute-force attacks** - Brute force attacks rely on trial and error and they are the most direst approach to password cracking. The attacker attractively submits all possible character combinations of arbitrary length until he guesses the correct one. As discussed in the password strength section, the amount of time needed for a brute-force attack increases exponentially in relation to the length of the password and is highly ineffective. Depending on whether the attack is online or offline (see more details in offline and online attacks sections), brute force attacks can be guarded against by choosing passwords of high entropy (offline attacks) and/or using rate limiting techniques (offline and online attacks) that limit the amount of available guesses an attacker can have or increase the time needed for authentication to take place.

- **Dictionary Attacks** - Instead of iterating through all possible combinations of characters randomly, attackers prioritize a subset of all the possible combinations to try first. This subset of possible combinations is essentially called a dictionary and it contains a series of leaked commonly used passwords that users have used in the past. It has to be noted here that due to the fact that users usually rely on natural language to produce their passwords and due to the development of natural language processing and machine learning technologies attackers are able to train language models on leaked passwords that are able to guess passwords even more effectively. However, dictionary attacks are relatively easy to defeat in theory just by choosing ransom passwords or pass-phrases or by creating uncommon passwords. Of course this is not the case when users come to choose their own passwords as discussed previously. Finally, salting is also a type of defence against these attacks.

- **Rainbow-Table Attacks** - The same way that dictionaries store common plain text passwords, a rainbow table is essentially a dictionary which stores a list of common passwords alongside their hashes for a given hashing algorithm. These tables take a considerable amount of storage space since they have to store the same string for each different hash function but they are well optimized to look up their entries with high speed and efficiency. Nonetheless high computer power is needed to run through these tables and also they can be rendered ineffective if salting is applied to the passwords before the hashing is conducted as discussed previously.

Direct attacks can be further classified into Offline and Online depending on whether the attacker has access to the encrypted database file or not respectively.

- **Offline attacks** are brute-force attacks that can be deployed in the event of a leaked or stolen hashed database file. The attacker uses his own computing resources to try to guess the master password that the database was encrypted with and the time taken until he succeeds is directly proportional to the power of his GPU.It can be understood that as technology progresses and more powerful GPUs are developed these attacks can become more effective. To defend against these attacks, master passwords should be chosen with high bit strengths and hash functions with slower hash iterations can be deployed as well.

- **Online attacks** take place in the case of the database not being leaked or stolen. Attackers submit passwords the same way a normal user would try to log in to his/her online account. Unlike the case of offline attacks, attackers are limited to the speed in which client and server communicate and to the resources and computing power of the server. Thus online attacks are inherently slower and so allow for weaker passwords to be chosen by users. To further defend against online attacks password blacklisting and throttling techniques are also deployed.

Attacks through Users:
These attacks are known as social engineering attacks which unlike brute force attacks that target the password authentication mechanism directly, they aim to deceive the users in some way so they would reveal their private information.

1. **Phishing**
   The malicious adversary in this scenario disguises himself as a trustworthy entity and communicates with an unsuspecting user usually through electronic means. Usually the attacker contacts the user through email (email spoofing) or instant messaging and provides a link leading to the a fake website of the service the attacker is pretending to represent. This fake site is designed to mimic the authentic website and usually is identical to the original with the only difference being in the URL. Once there the user is instructed to provide his/her credential information in order to log in to his online account and the adversary steals the information. Phishing is considered to be one of the most costly types of attacks and according to the 2013 Microsoft Computing Safety Index, released in February 2014, the annual worldwide impact of phishing could be as high as US $5 billion.

   [Dhamija et al., 2006] conducted a study that aimed to identify the main reasons that phishing was one of the most effective attacks on users. They discovered that: Good phishing websites tricked 90% of the study's participants and that existing anti-phishing browsing cues were ineffective. Security indicators as well as address and security bars of web browsers were ignored by 23% of participants. Furthermore 15/22 of the participants chose to visit websites with fraudulent certificates even when warned by the system.

Most of the participants were found to be susceptible to phishing attacks no matter their demographic background including age, sex, hours of computer usage.

They also found that phishing becomes effective due to 1) lack of knowledge 2) Visual deception and 3) bounded attention by analyzing common phishing strategies from 200 real life phishing examples.

2. **Keystroke logging**
Keystroke logging (also known as keylogging or keyboard capturing) is the strategy deployed by malicious adversaries to record keyboard input without the user realizing it. Keylogging tools (called keyloggers) vary between software and hardware forms. The former can be either malware programs or legit software installed on the users machine and they record directly digital input from keyboard devices. The latter vary much more in type and form between wireless sniffers, acoustic, optical, electromagnetic or firm ware based to name a few.

2.4 Problems with Regular Passwords

Passwords and Password authentication is by far the most dominant and highly adopted user authentication scheme and as suggested by [Bonneau et al., 2012] this is unlikely to change in the near future. This is because, as they conclude in their study, passwords have low costs when it comes to creation and management and also provide familiarity to use by end-users. Many alternative schemes have been proposed, like biometrics, multi-factor authentication, graphical authentication, One-time and token base authentication but as the study showed non has been superior so as to replace passwords.

Nonetheless, passwords are far from perfect from a security and usability standpoint. Following is a list of the most important points that make passwords problematic:

- **Password Entropy** - To be effective against the direct attacks (brute-force and dictionary attacks) we have mentioned above, the passwords chosen by users need to be increasingly lengthy and uncommon which makes it increasingly difficult to be created by the all users no matter demographic background.

- **Memory Demands** - As the number of accounts an average user possesses keeps increasing with the progress of technology so is the number of different passwords a user needs to remember and manage. Users of course try to overcome this issue by using the same passwords across multiple websites and services which is an insecure practise.

- **Social Engineering attacks** - As discussed previously, passwords by them selves do not provide any protection against any kind of social engineering attack, with phishing being a very serious threat.

2.5 Password Managers (PM)

2.5.1 Overview

Password managers (PM) are software systems that try to mitigate the problems identified for ”vanilla” password use, by assisting users create, store and manage passwords in a more secure and usable manner. PMs vary in types depending their implementation and functionality but non the less they aim to offer users:

- `Stronger Passwords` – this is achieved by visual password meters for users to choose their own passwords more securely or by automatic password generation.
- `Alleviates users from the Memorability burden` – Users can store their credentials into a secure database that can be decrypted with a Master Password/Key.
- `Eliminate password reuse across different profiles` – Since theoretically, memorability is no longer an issue with the user needing to remember only a master password he/she can choose different passwords for each service.
- `Faster authentication` – This is provided by auto-complete forms, copy&paste or drag&drop functionalities.
- `Some protection against social engineering attacks` – For example, phishing can be prevented by storing the correct URL of the webpage.
- `Save time and effort in the long run` – No forgetting passwords, No effort to come up with new passwords when asked to update an existing password

2.5.2 Taxonomy

Password Managers come in different types depending on the way they are deployed. We outline some of the most common types here and their most important features. A more detailed summary can be found in [McCarney, 2013].

Generative Password Managers

Mostly found in **academia**, they are typically used to produce cite-specific passwords. The way they achieve that is by asking the user to choose a single secret (usually referred as Master Secret or Master Password) which is then combined to the website’s

URL or Username to produce a unique secret. That unique secret is then hashed using a cryptographic hash function to produce the final output which will then be used as the service's password.

$$H(MasterSecret + URL) = password \tag{2.2}$$

where the H: cryptographic hash function, MasterSecret: chosen by user - always the same, URL: specific for each account - always different, password: always unique to the specific account.

Pros:

- Phishing protection since the PM uses a unique URL.
- Memorability Burden Free - user needs to remember only the Master Secret
- No poorly, low entropy site passwords
- No reuse of passwords across sites
- No stored state
- Allows users to access their accounts from different machines easily - all they need is the PM software and their Master Secret

Cons:

- Cumbersome initial transition - since the site passwords for each user account is generated by the PM system, this means all existing passwords for each account must be changed
- Cumbersome change of Master Secret - Every time the Master Secret is changed/updated, all site passwords need to be re-calculated and reset on by one.
- Users give full control to the choice of password to the PM - Users usually do not like to do that. Also if a site has a specific policy and the site password created by the PM does not match that policy would create problems.
- Single point of failure - Losing your Master Secret or exposing it to an adversary means loss or exposure of all of your accounts respectively.

Retrieval Password Managers

Retrieval PMs are the most common PM systems available for **commercial** use. These PMs ask the user to select a Master Password (or sometimes use the password of the Operating System they are used on) that is used to encrypt and decrypt a local database file. This file acts as a container to store the users site passwords along with associated metadata (like URL, Username/Email, etc) when used on the site for the first time. They come in different forms, some are desktop applications, some are browser plug-ins, etc.

Pros:

- Phishing protection - They store the correct URL as metadata and warn users when URL is different to the one stored.

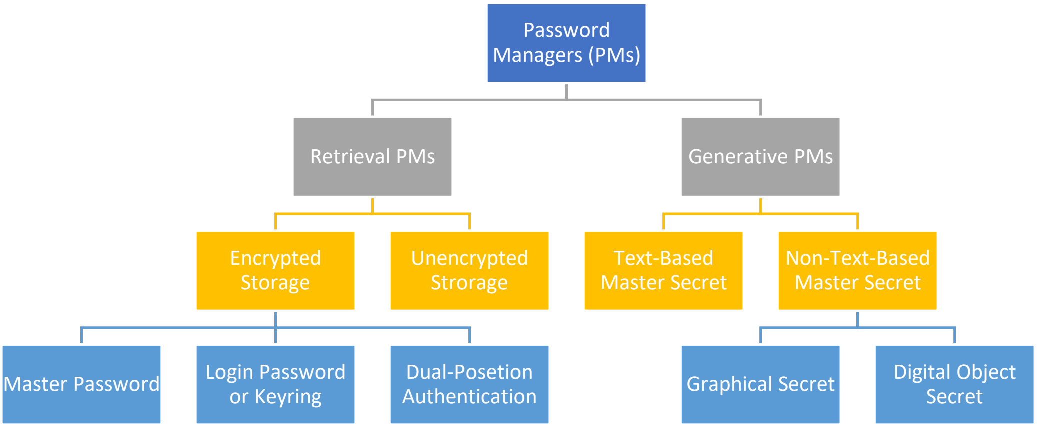


Figure 2.1: Further Classification of Password Managers

- Memorability Burden Free - user needs to remember only the Master Password
- Easy deployability - Users have the control to store passwords as they create them
- Easy initial transition
- Not a single point of failure - Adversary needs both the Master Password and the actual Database file to retrieve user credentials.

Cons:

- User-chosen passwords - low entropy
- Password reuse is allowed
- Session Model - Once Master Password is provided, the database remains decrypted for the remainder of the session.
- statefull
- Lack portability - This can be remedied by using a cloud based service, but then that service becomes the target of attacks.

What we have described above are the minimum functionality the 2 categories of PMs provide. Additional features, more frequently present in the Retrieval type, are the following: 1) **Backups** - the encrypted database file can be exported and saved locally, but loss of the master password ultimately prohibits the recovery of the content. 2) **Synchronization** - commercial PMs usually allow access to the encrypted database through different devices by storing the encrypted database file to either manually (any 3rd party cloud service like Dropbox or OneDrive or Google drive etc.) or automatically (some PMs like LastPass or Dashlane have their own cloud service) maintained cloud services. 3) **Random Password Generation** and **Password Meters** - these 2 features usually go together and they are self explanatory. The former generates random passwords which provide higher entropy passwords for the users to use if they choose to do so. The latter provides the users feedback about the strength of the password they are creating. 4) **Secure Notes** - some PMs allow users to encrypt notes the same way they encrypt their passwords. 5) **Web Access** - some PMs provide their services through web browser powered applications which implies that users can sign in to a PMs website and use the service from any device that can run a web browser.

Finally `Generative` and `Retrieval` PMs are sub-devided into further subcategories. These are summarized in figure [2.1] (re-drawn from [McCarney, 2013]).

2.6 KeePass2

Dashlane, 1Password, LastPass are some of the most popular and widely adopted commercial PMs that have managed to survive over the years. They all belong under the Retrieval PM class, as most commercial PMs do. Unfortunately these PMs are not entirely free to the public and do not have their source code and implementation open sourced. In that, we decided to focus on KeePass2 PM which abides under these two requirements. Firstly, since security is considered as a secondary task, it seemed counter intuitive to ask users to pay in order to accomplish such a task. Secondly, being an open source system, would have allowed us to modify and re-design it according to the results of our user studies.

KeePass [`https://keepass.info/`] is a **free**, **open source** password manager, which helps users to create, store and manage passwords and other sensitive credential information in a secure way. It belongs to the Retrieval PM class and so asks the user to create a **database file**, locked with a **composite master key** of his choice. Once the encrypted database file is created, the user can then populate it with multiple entries. An **Entry** represents the collection of a users credential information that are associated with one single account. **Entries** can be stored either directly into the database file itself and/or into **categories** which appear to be folders located inside the database file. The database can then be stored either locally or in the cloud. More specifically the main functionality of the software assists users to:

- Create an encrypted database file - this file will be encrypted by the Master Composite Key chosen by the user, shown in figure D.4.
- Populate the database with several password entries, shown in figure D.2. This can be done either by migrating password credentials for an existing online profile or creating a new one. A single entry contains the following fields | **Tittle** | **User Name** | **Password** | **URL** | **Notes** | and it will be fully encrypted; Not just the password field.
- Use the Composite Master Key to unlock the database and access any of the stored password entries, shown in figure D.2.
- Copy&Paste or Drag&Drop the appropriate field from the password entry to the appropriate website fields.
- Generate strong random passwords with various settings, and check the strength of both the automatically generated and the user chosen passwords as shown in

figure D.3.

KeePass2 uses Advanced Encryption Standards (AES) and the Twofish algorithm to encrypt the database, which are considered very secured at the time of writing. SHA-256 is used to hash the master key components. SHA-256 is a 256-bit cryptographically secure one-way hash function. No attacks are known yet against SHA-256.

KeePass 1 and 2 was developed by Dominic Reichl, and the official software can run on Windows OS only. Being an open source project it allowed the development of many unofficial ports over the years that support all main operating systems including MacOS, Linux, Android and IOS. Its functionality can also be extended by the numerous plug-ins developed by 3rd party developers that can be installed in the Original Version.

For this study we will only focus on the official software which provides the basic functionality, mainly because we want to avoid any trust related issues that users may experience due to the reference to the 3rd party software as unofficial. Also we will not consider any plug-in functionality either, since it appears too complicated and cumbersome to install by the average user.

Chapter 3

Cognitive Walkthrough(CW) Study - Original Version

3.1 Methodology

A cognitive walkthrough is a task-specific usability inspection method. It is based on the belief that people learn systems by trying to accomplish tasks with them, rather than first reading through instructions. It is ideal for systems that are meant to be walk-up and use.

To construct a cognitive walkthrough study the researcher needs to firstly identify a clear goal that the user wants to achieve using the software under inspection. This goal is the primary/main task. To achieve this task a typical user should go through a specific set of subtasks. The researcher conducting the walkthrough presents each subtask to a group of HCI experts through a series of screens. When each screen is presented, everyone is asked to write down the answers to the following four questions:

1. Will users want to produce whatever effect the action has?
2. Will users see the control (button, menu, label, etc.) for the action?
3. Once users find the control, will they recognize that it will produce the effect they want?
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?

This process is performed for each one of the main tasks identified. Once this is completed the participants can discuss the results both among themselves and with the researcher. Feedback and results from each HCI expert is recorded by the researcher.

To complement the above vanilla implementation of the cognitive walkthrough additional elements were introduced:

- **Persona**: This is a description/profile of the average user of KeePass2. It aims to provide the HCI experts with a better understanding of the target population.
- **Scenarios**: The scenario is Main Task specific and aims to provide the HCI experts with a better understanding of the goal the user is trying to achieve by using the KeePass2 software.

3.2 Study Design

The persona outlined below represents our definition of the average user and will be used throughout the study to define the average user and his/her attributes and qualities.

3.2.1 Persona: Alice

Alice is a 38-year-old real estate agent working for the real estate company, House of Cards. Alice regularly uses a Windows Personal Computer both for her professional and personal life.

Professionally, she uses her PC to manage information about houses as well as sensitive information about her clients private data. Her day to day usage of her PC involves using Microsoft Office Products, image and video editing software along with a basic usage of the Microsoft file system. Due to the companys security policy, Alice needs to use strong passwords for her accounts, which she must provide every day at the office computer.

At home, she uses her PC to manage her email, online banking, social, dating, and Netflix accounts. As well as the sites she uses rarely.

3.2.2 Main Tasks

As mentioned previously the researcher needs to identify the main tasks/goals an average user would want to achieve using the security software. Based on the description of the average user introduced with the persona above a list of the following main tasks was identified.

- **Main Task 1:** Creation of a new encrypted database
- **Main Task 2:** Populating the new database with password entries. These would be either new entries or existing online accounts that the user will have to migrate to KeePass2

- **Main Task 3:** Logging into KeePass2 - Unlocking the encrypted database - Using saved entries to log in to online accounts
- **Main Task 4:** Changing or Updating entries

3.2.3 Study Purpose

Define Usability

As suggested by [Whitten and Tygar, 1999] and extended by [Chiasson et al., 2006], for a security software to be usable, users must:

1. be reliably made aware of the security tasks they must perform.
2. be able to figure out how to successfully perform those tasks.
3. not make dangerous errors.
4. be sufficiently comfortable with the interface to continue using it.
5. be able to tell when their task has been completed.
6. have sufficient feedback to accurately determine the current state of the system.

The same way that the Persona will be used to define the average user for the rest of the study, the above definition of Usability will be used throughout the study to describe what we mean by usability.

Research Question

If an average user that fits into the above persona decides to use KeePass2, will KeePass2 current design allow that person to realize what needs to be done, figure out how to do it, and avoid dangerous errors, without becoming so frustrated that he or she decides to give up on using KeePass2 after all?

3.3 Study Setup

To answer the research question as accurately as possible while accounting time and resource constraints, the cognitive walkthrough was performed in 2 separate settings; A grouped setting and an individual setting. The Group study was run first with the help of several HCI experts. During the study the experts were asked to evaluate only one of the main tasks(2nd task) identified in the section above, since that was deemed the most important from the list. The Individual study was run second only by the researcher while taking into account the results and feedback of the Group study. During the study the rest of the main tasks were evaluated and analyzed. Both studies are described in the following sections in more detail.

The full documentation used during both studies which includes Ethics and Consent forms, Questionnaires, Cognitive Walkthrough Sheet for the HCI experts as well as a power point presentation to brief the HCI experts is too large to include. In the Appendices section we provide the 2 most important main tasks, one that was used during the grouped study and one that was used during the individual study. Here, we outline the main tasks along with their corresponding scenario and sub-tasks.

3.3.1 Group Study

The group study was run during one of the weekly TULIPS workshops organized by the project supervisor. This setting was deemed ideal to run such a study since the workshop participants consisted mainly by HCI(Human Computer Interaction) and/or security experts. The general organization of the workshop was the following:

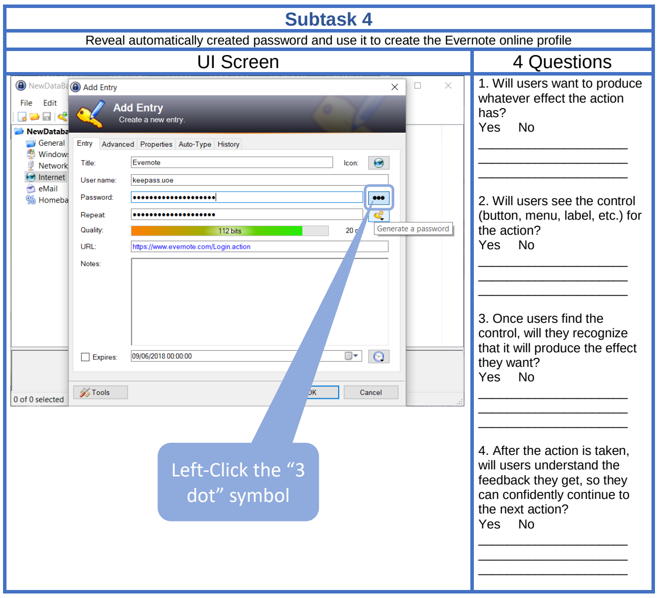
- *Briefing of Participants* - The participants were briefed about the general premise of the MSc project, introduced to KeePass2 and password managers in general and instructed on how to perform a cognitive walkthrough evaluation trough a Power Point presentation.

- *Signing of concent forms* - These informed the participants about their privacy rights if they chose to participate in the study along with an explanation about the data usage for this study

- *Answering a short questionnaire* - This involved basic demographic and basic password authentication related questions

- *Performing the cognitive walkthrough* - Each participant was handed a booklet that depicted each one of the subtasks for a particular main task along with the 4 questions that they needed to answer for the particular subtask. An example of one of the subtasks is depicted in 3.1. The booklet also included the Persona description of the average user and a scenario that explained the premise of the Main task. Along with the booklet, the participants had the option to look at the same screenshots on a large projector screen as part of a PowerPoint presentation. For even more clarification the actual running KeePass2 desktop application was available for display to all participants.

- *Discussion and Comments* - Finally the participants were able to discuss both among themselves and with the researcher about the subtasks they have gone through as well as provide any other comments for the KeePass2 UI(User Interface)



Subtask 4

Reveal automatically created password and use it to create the Evernote online profile

UI Screen

Title: Evernote

User name: keepass.uoe

Password: •••••••••••••••••••

Repeat: •••••••••••••••••••

Quality: 112 bits 20

URL: https://www.evernote.com/Login.action

Notes:

□ Expires: 09/06/2018 00:00:00

Left-Click the “3 dot” symbol

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Figure 3.1: Cognitive Walkthrough subtask example as presented to the study participants

3.3.2 Individual study

Taking into account the results from the Group study, an Individual study was performed by the researcher in order to assess the remaining main tasks previously identified. Running the Individual study allowed for a more detailed analysis since background in usability of password authentication as well as user password habits were taken into account. This background knowledge was encapsulated into a series of additional questions used along side the 4 original Cognitive Walkthrough questions. These additional questions are listed below and basically expand the research question previously stated.

Additional Questions:

1. Will the user understand that he needs to create an encrypted database which will act as the secure container to accommodate all his credential entries?
2. Will the user realize that the Master Password he is asked to choose will allow him to unlock the specific encrypted database he was asked to create. In other words will he/she understand the link between the encrypted database and the master password.
3. Will the user understand how to choose a secure master password?
4. Will the user realize that losing the master password will make it impossible to recover the database file?
5. Will the user realize that the database file he has created has to be maintained manually and once he has realized this will he be able to do so securely?
6. Will the user be able to successfully and securely populate the database with entries?
7. Will the user understand that adding a new entry in the database will not result

in the creation of the corresponding online profile?

8. Will the user be able to use the various elements of an entry to access his online profile without running any security risks while doing so?

Furthermore, during the individual study the validity of the 10 Nielsen's heuristics [Nielsen, 1995] was also taken into account, through a method called Heuristic Evaluation.

Heuristic Evaluation

This method was firstly introduced by [Nielsen and Molich, 1990] and was later refined to a 10 Heuristics list in [Nielsen, 1995]. Three to five UX(user experience) experts (or novices trained on the heuristics) individually assess a product by walking through a core set of tasks and noting any places where heuristics are violated. The evaluators then come together to combine their findings into a single report of issues that should be addressed. Note that, products that adhere to all 10 heuristics are not guaranteed to meet users needs, but it is significantly less likely that they will face the barriers of poor design. The list of Nielsens 10 Heuristics can be found in the appendix.

3.4 Study Results

3.4.1 Participant Demographics

Analysis of the pre-questionnaire revealed that 37.5% of participants were male and 50% of participants were female. 62.5% of participants belonged to the 21-30 years old age group where as 37.5% belonged to the 31-40 years old age group. Nearly all participants (7/8) claimed to have had either a university level education in HCI or have worked in HCI and 6/8 of them claimed to have had either a university level education in Computer Security or have worked in Computer Security. Furthermore, all participants said to have used at least one of the 3 main OS(Operating System) for the last 10 years. Among those, 87.5% claimed to have been using Windows. 62.5% of the participants said to have never used a password manager, 25% have used only LastPass and 12.5% percent has used both LastPass and Keepass.

3.4.2 Group Cognitive Walkthrough results

As previously mentioned the Group study was run first focusing on main task 2. What follows, is the outline of the subtasks along with the main results/conclusions drawn from each. We suggest the reader to refer to the depicted results of the study which are in the appendix [E.1,E.2] section due to space limitations.

Main Task 2:
Create a new entry for your newly created Evernote account. The new entry should be created in the database file called NewDatabase under the Online subfolder.

Scenario:
Alice needs an application to take notes during work. A colleague of hers suggests Evernote, but to use it she needs to first create an online Evernote account. She navigates to the Evernote official website and clicks the sign-up button. Alice has already installed KeePass2 in her personal computer and created an encrypted database by choosing a master password. She now plans to use KeePass to create and save the Evernote password.

Subtasks:

- **Subtask 1:** Navigate to the Internet subfolder of the current database, named NewDatabase
  Result/Comment: File system not very clear. Confusing that you can add entries under the database file directly.

- **Subtask 2:** Add an entry to the ”Internet” subfolder
  Result/Comment: The ”new entry” symbol not clear. Might be needing a ”+” symbol along with the key symbol. There is a more promising symbol which is the one that creates a new database and not a new entry.

- **Subtask 3:** Fill in the details of the new entry
  Result/Comment: Although filling in the text fields is a straight forward task participants noted that it might be confusing to the user about what the correct content of the fields might be. For example he might fill in the wrong URL or be confused about what he should enter under User Name since a lot of web services usually require email instead.

- **Subtask 4:** Reveal automatically created password and use it to create the Evernote online profile
  Result/Comment: A lot of confusion about the ”3 dot” symbol; Also some confusion about the ”generate password” symbol; user might confuse the 2 - User might need more guidance in what is the series of steps he has to complete in order to successfully complete this step/interface.

- **Subtask 5:** Copy and paste the revealed password to the Evernote sign up webpage and create your Evernote account
  Result/Comment: Copy and Pasting the generated password comes only from system knowledge. Nothing in the UI to suggest that this is the action that needs to take place. Suggestion of a ”copy password” button.

- **Subtask 6:** Finalize the creation of the Evernote Entry
  Result/Comment: Clear that an entry is added. OK button might be clearer if it said save or done.

- **Subtask 7:** Save Changes
  Result/Comment: Users will think that adding an entry is equivalent to saving it also. This is not the case. When users do save, there is no well visible cue to indicate that they have saved their entry.

3.4.3 Individual Cognitive Walkthrough results

As previously mentioned the Individual study was run 2nd focusing on the rest of the main tasks identified. What follows, is the outline of the subtasks for each main task, along with the main results/conclusions drawn from each. The comments marked in orange indicate which of the 10 Nielsen's heuristic is violated by the particular UI screen. We suggest the reader to refer to the depicted results of the study which are in the appendix [E.3,E.4,E.5,E.6,E.7] section due to space limitations.

Main Task 1:
Create a new encrypted database.

Scenario:
Alice has decided she has too many passwords that she has to remember, so she decides to start using KeePass2 to manage them. Her computer at work already has KeePass2 installed and a co-worker recommended she use it. Because she has never used it before, when she starts she has to setup a new password database before she can enter any new passwords.

Subtasks:

- **Subtask 1:** Click the File menu item.
  Result/Comment: The user needs to understand that a database file needs to be created, before he/she can save any of his online account credentials using KeePass2. The UI the user encounters when KeePass2 is launched for the first time gives no indication for that action.

- **Subtask 2:** Click New...
  Result/Comment: The user needs to choose either the "New..." option from the File drop down menu or choose directly the "New" symbol in the main UI screen. Neither option seems intuitive for the user to want to invoke. The user has no idea what he wants to create a new of in the first place. Furthermore, the visual metaphor of the "New" symbol is quite poor.

- **Subtask 3:** Read the notification and Click Ok
  Result/Comment: Once the user tries the "New..." button, only then he is provided with an explanation. Although "Database" is the correct domain name for what KeePass2 is creating, this term might be alien to the average user. (Match between system and the real world)

- **Subtask 4:** Choose the Directory where your new database will be saved, and

press save
Result/Comment: This step should be straight forward for users familiar with the windows file system. The only issue here, is that the user has no choice to go back a step in order to read the instructions of the previous screen. His/Her only choice is to cancel and start the process from the beginning. (User control and freedom) (Recognition rather than recall)

- **Subtask 5-8:** Create the Composite Master Key - Note: Subtasks 5-8 are condenced in this section due to space limitations. See them in full in the Appendices.
  Result/Comment: In this section we present comments for both alternative paths a user can take to complete subtasks 5-8. If a user chooses to read the fine print (see text in the orange square) he will understand that in this step he is creating a Composite Master Key which he will use to unlock the database. This Composite Master Key can be created by combining a Master Password, and/or the 2 expert options hidden behind the ”Show expert options” check box. We think that, both the technical language used in the fine print and the hidden expert options will result in the user being confused. We believe that the user will probably not even read the fine print and assume that he is only asked to create a master password, while forgetting all about the Composite Master Key nomenclature. This is likely to cause inconsistency in the system. For example once the user has created the database file and then he/she wishes to change the master password, he will only encounter a ” Change Master Key” option and no reference about a master password anywhere in the system (Consistency and Standards). Also the ”3 dot” symbol is again a bad visual metaphor (Match between system and the real world). It has also a double hidden functionality which is a bit confusing - it reveals the password and when pressed a second time it hides it and copies it to the repeat password field.

- **Subtask 9:** Fill in the database details
  Result/Comment: Users will think that adding an entry is equivalent to saving it also. This is not the case. When users do save, there is no well visible que to indicate that they have saved their entry.

- **Subtask 10:** Accept the changes Database settings
  Result/Comment: Users will think that adding an entry is equivalent to saving it also. This is not the case. When users do save, there is no well visible que to indicate that they have saved their entry.

- **Subtask 10:** Skip printing the Emergency Sheet
  Result/Comment: Users will think that adding an entry is equivalent to saving it also. This is not the case. When users do save, there is no well visible que to indicate that they have saved their entry.

- **Subtask 10:** Save the newly created database
  Result/Comment: Users will think that adding an entry is equivalent to saving it also. This is not the case. When users do save, there is no well visible cue to

indicate that they have saved their entry.

Main Task 3:
(Unlock your database and) Use an existing entry to log in to your online account.

Scenario:
Alice has been using KeePass2 for a while now. She has created an encrypted database file and has populated it with multiple entries. One of these entries is her Evernote account credentials. She wants to use Evernote and to do so, she needs to sign in to her online account.

Result/Comment: To unlock his/her database file, a user is asked to enter a ”Master Key”, where as while creating a new database, the user was asked to create a ”Composite Master Key” [see figure E.5]. The terminology used is inconsistent and thus it might confuse users (Consistency and Standards). The icons used in creating and unlocking the encrypted database do not match either. Once the database is unlocked the user needs to right click the Evernote entry and use the first 3 options from the drop down menu [see figure E.6].

Main Task 4:
Change the Title, URL and Password of the Evernote entry.

Scenario:
Alice has been using KeePass2 for a while now. She has created an encrypted database file and has populated it with multiple entries. One of these entries is her Evernote account credentials. She realizes that she has made a few mistakes while creating that entry.

Result/Comment: For this task, the user will need to once again right click the Evernote entry and select the ”Edit/View entry...” from the drop down menu. This will take him/her to the Edit Entry screen as shown in figure[E.7]. This screen has the same symbol, options and controls as the ”Add Entry” screen. Although this can introduce familiarity to the user, it could be also confusing to some extent.

3.5 Summary

3.5.1 Study Organization

As discussed in this chapter, the CW was performed in two settings, the grouped and the individual. During the grouped setting, 8 HCI experts were asked to assess the usability of Main Task 2 using screen shots of the original KeePass2 version by answering the classical 4 CW questions. During the individual setting, the researcher performed the same process on the remaining of the identified Main tasks, taking into account additional usability aspects like the Nielsen’s 10 Heuristics and background literature discussed in the Background Chapter.

3.5.2 Study Results

The conclusions drawn from this study were more specific and subtle than the ones discovered during the TA (Think-Aloud). Nonetheless experts suggested that in general the application would be unusable for the average user. For *creating a new Database*(**Main Task 1**), it was identified that users have no indication that they need to perform the task to begin with. For *adding a new Entry* (**Main Task 2**) the experts pointed out that again the control to initiate the process was hardly noticeable and that the work flow of the process itself gave no guidance/support to the user. For *using an existing Entry* (**Main Task 3**), it was identified that there might be some mix-up between the intended controls of performing this task and the controls for Editing/Viewing an existing entry by the user. Furthermore, **Main Task 4** along with the rest of the tasks identified in general inconsistencies in the usage of icons, poor visual metaphor choices and inconsistencies to the work flow among other issues.

We advise the reader of course to take some time and inspect the figures related to each of the 4 Main Tasks in order to understand and familiarize themselves with the UI of the original version of KeePass2.

Chapter 4

Think-Aloud(TA) Study - Original Version

4.1 Overview

As mentioned in the introduction, the Think Aloud protocol is used to evaluate both the Original version and the Revised version of KeePass2. Evaluation of the Original, provided further insight on the usability flaws of KeePass2 and suggested possible improvements for the development of the Revised version. The results of the Think Aloud evaluation of both versions is used as a direct comparison of the two versions in subsequent chapters.

In this chapter we introduce the general methodology and study set up of the Think Aloud protocol and finally present the results of the study on the Original KeePass2 version. The results of the revised version will be presented and analyzed in chapter 6.

4.2 Methodology

The Think Aloud protocol[Nielsen, 1995, Nielsen, 2012] is considered one of the most effective and commonly used evaluative methods and falls into the usability lab study category. During the study, participants are asked to complete a set of tasks using the product/interface under investigation. While doing so, they are asked to verbalize their thoughts, actions and feelings. This allows the researcher to observe the process of task completion unfold and so reveal the aspects of the interface that delight, confuse and frustrate the user. Being a lab study method, the Think Aloud protocol is performed in a controlled environment and the session is usually recorded with a multimedia recording device. The recordings can be then referred back to as both testimony and as data to be analyzed by the researcher.

It has to be noted that participants perform the experiment during individual sessions and are all asked to perform the same set of tasks. Before doing so, the participants are explained and trained on how to think aloud correctly.

4.3 Study Design

4.3.1 Recruiting Participants

Since the target of this study was to develop a password manager software that will be usable for the average user (as defined in 3.2.1), recruiting participants from the Informatics department of the University was not deemed a good choice. Thus recruiting teacher stuff from Cypriot primary schools was thought to be a closer match to the target population of this study. Short emails were sent to 30 primary school teachers explaining the purpose of the study. The first 10 to respond were then chosen to conduct the Think Aloud studies. 5 were randomly chosen to test the Original version and the remaining 5 to test the Revised version.

It should be noted that it was thought best if we tested the participants in their native language (Greek) so the most important documents used during the study were translated to Greek by the researcher. The participants were instructed to speak in any language (either Greek or English) they felt more comfortable with during the TA.

There is a lot of debate about the number of participants needed for usability evaluation (see [Borsci et al., 2013] for an academic evaluation). [Mathematical, 1910] argue that 5 participants are enough to uncover the most important of usability issues, as long as more than one evaluation is performed.

4.3.2 Choosing Tasks

The complete documentation of the Task sheet that was handed to the participant during the study can be found in appendix B.4. In this section we present the 5 tasks chosen for the Think Aloud study and explain the rationale of choosing those particular tasks.

Each task was introduced to the participant with a scenario. This was a general description/premise of the task the participant was asked to complete. Its purpose was 2 fold. It acted as a more detailed explanation of the situation and also it hoped to encourage the participant to treat the session as a real life situation and not as an experiment.

It has to be emphasized that some of the tasks required the participant to use real credentials. To avoid any security issues and any additional biases to the experiment, an additional **Credential Sheet** (see appendix B.5) was handed to each participant with already created dummy accounts to use during the study. This included email addresses and passwords where appropriate along with any other information needed to complete the tasks.

Task 1:
Scenario: Pretend that you have been having a hard time remembering all the passwords of your online accounts lately and a friend at work has suggested that you use Keepass2 password manager to help with this problem. You decide to start using Keepass2 and you download and install it in your computer.

1. **Save your Gmail account credentials into Keepass2. When you are done, exit Keepass2.**

2. **Launch Keepass2 again and find your Gmail account credentials.**

Task 1 Purpose: As explained in chapter 2, KeePass2 requires the user to create an encrypted, local database file in the process of which he is asked to choose a Composite Master key in order to encrypt this file. Only then he can proceed to add into this file the credentials of his online accounts. This was identified during the cognitive walkthrough as one of the most troublesome aspects of KeePass2. We wanted to test this and so the first task was designed specifically for this purpose. Instead of asking the participant directly to create a new database file, we asked him/her to add a Gmail credential. This aimed to reveal if the user would identify the need of completing the preliminary task of creating a database file before proceeding to add his Gmail credentials. In the case the participant was not able to complete this preliminary step, a brief explanation was given to him/her during the study as a **hint** and was allowed to proceed with the task.

Task 2:
Scenario: Pretend that you have been using Keepass2 for a while now. You have added several of your online account credentials into Keepass2. One of them is your Evernote account.

1. **Sign in to your Evernote account using Keepass2.**

Task 2 Purpose: The 2nd task aimed to investigate whether a participant would be able to unlock an existing database file, find an already added entry and use it correctly to access an online account.

Task 3:
Scenario: A friend at work has suggested that you start using Facebook. To do so though, you need to create an online account. You navigate to the official website of Facebook and you press the sign-up button in order to create your new online account.

1. **Create your new Facebook account using Keepass2 to help you. Remember to store your final password in Keepass2 so that you can remember it later. When you are done, exit Keepass2.**

2. **Launch Keepass2 again and find your Facebook account credentials.**

Task 3 Purpose: Task 3 aimed to reveal whether a participant would successfully remember how to find the controls to create a new entry. More interesting was to see whether the user would first create his new Facebook account without the help of KeePass2 (i.e. constract a password by himself) or choose to first consult KeePass2 for a secure random password and then use that password to create the account.

Task 4:
Scenario: You have realized that the password of your Evernote account is not strong enough. You decide that you need to update it to a strong one.

1. **Update (change) the password of your Evernote account to a strong one. When you are done, exit Keepass2.**

2. **Sign in to your Evernote account with the new password.**

Task 4 Purpose: Task 4 purpose was 3 fold. Firstly, we wanted to check whether the participant would find the controls to edit an entry. Secondly, we wanted to check whether the password meter will have been used successfully to produce a stronger password. Thirdly, we wanted to capture whether the participant understood that there was no connection between the KeePass2 entry and the corresponding online account. This was crucial as to understand whether the mental models induced to the participant from the UI were the intended and correct ones.

Task 5:
Scenario: You realize that you have been using the same master password for a while now, so you decide to change it just to make sure it hasnt been compromised.

1. **Change (update) your master password.**

Task 5 Purpose: Finally, task 5 aimed to test whether the user would find the controls to update the master password and whether he would realize the difference between master password and master key identified in figure E.5

4.3.3 Pre/Post Questionnaire

Pre-Questionnaire: This was handed to the participant to complete right after he signed the consent form and aimed to collect basic demographic information about the participant along with some basic habits concerning passwords.

Post-Questionnaire: This was handed to the participant after the completion of the think aloud session. It aimed to record the participants impressions and feelings about the software using the System Usability Scale (SUS)[Brooke et al., 1996] which uses a series of 10 Likert scale like questions. Finally it aimed to capture information about the mental model the participant developed after interacting with the software.

The full documentation of the questionnaire is shown in appendix B.3.

4.4 Study Setup

4.4.1 Overview

The general organization of the study was the following:

- *Acquaintance with Participant* - Brief conversation with participant to relax.
- *Signing of consent form* - This informed the participants about their privacy rights if they chose to participate in the study along with an explanation about the data usage for this study (see appendix B.1)
- *Pre-Questionnaire* - see details above
- *Briefing and Training* - see details in subsequent section
- *Performing the Think Aloud* - Each participant was handed the Tasks Sheet and Credential Sheet and allowed to complete each task using a Personal Computer with the KeePass2 software installed.
- *Post-Questionnaire* - see details above
- *Discussion and Comments* - Participants were finally encouraged to give any feedback about KeePass2 and were answered any questions concerning password security and password managers they might have developed during the study.

4.4.2 Controlled Environment

Both Think Aloud studies were carried out in a private office. The venue for the study was chosen specifically so as to avoid any distractions to the participants and to minimize any biases. A multimedia recording device was placed opposite the office desk to record the session. The participant was provided with a Windows 10 Personal Computer with the appropriate version of KeePass2 installed on the desktop. The PC was also running a screen recording software that captured the screen during the session. Due to the nature of the tasks, the participant was asked to use a web browser along with the KeePass2 software to complete some of the tasks. A Guest-mode Google Chrome browser was provided to all participants.

4.4.3 Participant Briefing and Training

The participants were briefed using the researcher script (see appendix B.2) to minimize any biases. Here we outline the key points mentioned during the briefing.

- Participants were explained in general what password manager applications are, and what they are used for.
- Specifically we explained that KeePass2 is a desktop application that is downloaded and installed onto a Windows operated device.
- Participants were explained how to perform a Think Aloud study and shown a video developed by the researcher as an example.
- Finally, we made sure to emphasize again that:

  1. KeePass2 helps people to:
     Create passwords - Keepasss2 can help generate strong, secure password for you to use instead of using your own.
     Store - KeePass2 can help store passwords and other credentials for your online accounts, so that you don't have to remember them by heart.
     Manage - KeePass2 can help you use those stored credentials to sign in to your online accounts.
  2. Treat this study as if you would behave in real life. Imagine that the accounts used and created through this study are your own real accounts.

4.5 Study Results - Original KeePass2

4.5.1 Pre-Questionnaire Results

Demographics

4/5 participants reported to be female and 1/5 participant reported to male. 2/5 were between the age of 25-30, 2/5 were between 36-40 and 1/5 between 41-45. All the participants reported to have had higher education in Teaching (2/5 a MSc degree and 3/5 a BSc). All participants reported Greek as a native language and fluency in English.

Computer and Online Experience

4/5 participants claimed to sign in to their online banking accounts weekly and 1/5 monthly. 4/5 of participants said that they use their work computer to sign in to their accounts weekly and 1/5 participants monthly. When asked what operating system do they use, all participants reported to use Windows regularly. 3/5 participants reported to use Chrome and Safari as their primary web browser and 2/5 said they use Chrome and Firefox.

Password Habits

2/5 participants said to have 0-10 online accounts that require password authentication, 3/5 reported 11-20. Respectively the participants that reported having less online accounts reported to have 1-3 unique passwords across those accounts and the ones that reported more said to have 4-6. It should be noted that participants were told to discount the passwords with slight permutations as unique. All participants said that they change their passwords only when prompted by the respective service. 4/5 of the participants reported that they both use memory and writing in physical format to remember their passwords and they marked this method medium in convenience since they don't always carry around the physical format. Surprisingly those participants felt that this method was not secure. 1/5 of the participants said that he/she remembers passwords by writing them down in digital format ("I write them down on paper, then take a picture of them and dispose the paper") and from memory. She marked this method as convenient and surprisingly thought that it was secure since he/she noted that "A computer cannot recognize hand writing". Finally, none of the participant reported to be using/used before any password manager application.

4.5.2 Think Aloud Results

As discussed in the beginning of this chapter, most of the data gathered from the think aloud protocol were **qualitative** in nature (video recordings and screen captures of each participant). To interpret and analyze the results as accurately as possible and to draw useful conclusions for the design and development of the revised version of KeePass2, it was decided to present and analyze the results in two ways. 1) Calculate a representative rate of success/failure of task completion, along with a measure of how much participants had strafed from the ideal path of competing each task (Deviation from path). 2) Identify the major themes by transcribing and coding the participants verbalization during the think aloud sessions by analyzing each one of the video recordings. The former are presented in table 4.1 and 4.2 and the latter in the following subsection.

To mark the success/failure of each task, the following labels were used:

- **Completed** (comp.) - The participant completed the task successfully without any erroneous actions.
- **Failed** (failed) - The participant failed to complete the task by giving up.
- **False Completion** (F. comp.) - The participant completed the task he was asked

to complete but with making mistakes in the process.

- **Dangerous Completion** (**D. comp.**) - The participant completed the task he was asked to do, but while doing so he committed actions that compromised security.

Please note that the exact meaning of each of the 4 labels depends on the context of the 5 different tasks the participants were asked to complete and not all of the labels apply to all situations(tasks). In the following section we explain their meaning according to the context of the task were applicable.

Think Aloud Task Completion						
	Goal	P1	P2	P3	P4	P5
Task 1	Create New DB	F. comp.	F. comp.	comp.	failed	F. comp.
	Add Existing Account	failed	failed	failed	F. comp.	comp.
Task 2	Use Existing Entry	failed	failed	**D. comp.**	comp.	failed
Task 3	Add New Account	F. comp.	failed	comp.	comp.	failed
Task 4	Edit Entry	F. comp.	F. comp.	F. comp.	**D. comp.**	**D. comp.**
Task 5	Change MP	comp.	failed	comp.	comp.	comp.

Table 4.1: Task completion results of the 5 participants on the Original KeePass2 version. DB = Database, MP = Master Password

Think Aloud results			
	Goal	Success Rate	Deviation from Path
Task 1	Create New DB	20%	3.6
	Add Existing Account	20%	
Task 2	Use Existing Entry	20%	2.8
Task 3	Add New Account	40%	1.4
Task 4	Edit Entry	0%	0
Task 5	Change MP	80%	3.6
Mean		30%	2.28

Table 4.2: Task statistics. DB = Database, MP = Master Password

Task Specific Themes

Task 1

- **Create New DB** - All participants used the new... control but all with doubt. Most of them complained about the ”Encrypted Database” terminology. Most participants provided the Gmail account credentials while creating the encrypted database without realizing that creating an encrypted Database was a preliminary step. The only participant that did realize it, complained about needing to cancel the process several times in order to be able to read the instructions in one of the wizard pages. All participants were confused about naming the database file through the file explorer step and also being asked again to name the DB during

DB settings. None of the participants bothered to look through the DB security settings.
- **False Completion**(F. comp.) meant that the participant had created a new Encrypted Database but used the credentials for the Gmail Account provided thinking that he was adding the Gmail Account right from the start.

- **Add Existing Account** - 3/5 participants had trouble locating Add Entry... control. Most of them, hovered above the control, read the tool-tip, but still thought that this was the wrong control. Grid-Box was extremely confusing to participants. All participants tried to press on Grid-Box column titles in order to add entries.
  - **False Completion**(F. comp.) meant that the participant added the User name and/ Password information in the wrong fields e.g under the notes section.

Task 2

- **Use Existing Entry** - While unlocking the encrypted DB file, some of the participants became confused by the 2 additional options/methods available for providing the Composite Master Key. Most of the participants tried to double click the Evernote entry. The result depended on the specific column of the Grid-Box they double clicked. This multiple functionality seemed very confusing to participants and it was the main reason for giving up. Most participants failed to notice that by selecting the Evernote Entry, some of the top toolbar controls became available/activated.
  - **Dangerous Completion**(**D. comp.**) meant that the participant used the Edit/View Entry control and erroneously revealed the password field in order to copy/paste it to the appropriate field.

Task 3

- **Create New Account** - Some of the participants seem to believe that the different groups give different functionality to the entry. All of the participants firstly created their new Facebook account using the browser, and then added the Facebook entry into KeePass2. Consequently, they did not use KeePass2 to create the password and instead chose their own. None of the participants filled in the URL field.
  - **False Completion**(F. comp.) meant that the participant would populate the fields of the new entry with the wrong information. e.g. Type their name in the ”User name” field.

Task 4

- **Edit Entry** - Most participants just added a few extra digits at the end of the already existing Evernote password. They stated that ”To make the password strong, I just need to add more characters...”. 3/5 participants erroneously changed the password in KeePass2 before updating their Evernote password from the browser.

- **False Completion**(F. comp.) meant that the participant would update the information using KeePass2 before he updated the information of the Evernote account.
- **Dangerous Completion**(**D. comp.**) meant that the participant did manage to change the password but not to a strong one.

Task 5

- **Change MP** - Although it caused confusion to all of the participants, most of them found the Change Master Key... control, ignored the difference between the "Key" and "Password" terms and changed the Master Password.

General Themes

Save Functionality
None of the participants noticed that changes to the DB of any kind where not saved automatically. Once they tried to exit KeePass2, they were prompted with a warning message which made them aware of the situation and resulted in all of them saving the changes successfully.

4.5.3 Post-Questionnaire Results

Participants' Satisfaction

Participants satisfaction was captured as mentioned earlier using the System Usability Scale (SUS) [Brooke et al., 1996]. The results are summarized in table 4.3.

SUS Results								
	P1	P2	P3	P4	P5	Mean	Sd	SE
Score	25/100	25/100	60/100	65/100	32.5/100	41.5	19.5	8.7

Table 4.3: SUS scores for the Original KeePass2 version - Sd: Standard deviation, SE: Standard Error

Participants' Mental Model

When asked about the location keepass2 saved the database file it created, 4/5 participants had formed the wrong impression.(3/5 participants reported that they thought KeePass2 saved the database file it created online, 1/5 said he didn't know and 1/5 said on the computer.) When asked about losing their master password, all 5 participants again erroneously thought that they could recover their master password by contacting the KeePass2 service. Surprisingly when asked about a hacker stealing their database file, 4/5 reported that their database will be hacked where as the remaining 1 said he/she wouldn't know. Finally when asked about the strength of the passwords used during the study (the ones the researcher had pre-chosen), 4/5 had formed again the wrong impression (2/5 said medium strength, 2/5 said they did not know and the remaining 1 said low).

4.6 Summary

In this chapter we have extensively described the methodology and setup of the Think-Aloud(TA) study. We have presented the results of the TA study performed on the Original version of KeePass2 in this chapter. These results are used in two ways. 1) To identify and improve usability of the new version and 2) to act as a control, in order to evaluate the performance of the revised version.

4.6.1 Think-Aloud Study Organization

The TA study was performed on 10 participants in total (5 on the original and 5 on the revised version). The participants were recruited from the stuff of various Cypriot Primary Schools and they were asked to answer a Pre/Post - Questionnaire and complete 5 main tasks while verbalizing their thoughts. These sessions were recorded with a video camera and screen recording application a condition that participants agreed upon with an informed consent form at the start of the study.

4.6.2 Results - Original Version

Ecological Validity

The demographics collected on the 5 participants of the original version show that the participants are a fair representation of our target population. They had on average 10-15 online accounts, all have been using a Windows powered PC and were familiar with the Google Chrome browser application used during the study. No one had ever used a password manager application before the study.

Of course it has to be acknowledged that the participants were recruited from stuff of primary highschools in Cyprus which obviously limits the variability of the results in terms of the carrier sector of the sample collected. This will be further discussed in the limitations section of the report.

TA results

The major outcomes and conclusions drawn from the TA analysis, was that the current design of KeePass2 would be unusable for the average user that tries to use the system for the 1st time. If the user somehow continues to use the system he will probably get himself into erroneous situations. For example he/she will be surprised that he can't access his accounts from a different devise or he will never be able to retrieve his credential information once he has forgotten his Master Password. The reason for this, is attributed in some degree to the **domain language**(e.g. ”Encrypted Database File”), **visual metaphors**(e.g. Database File visual representation is completely absent) and **work flow** design used to create KeePass2.

The usability issues uncovered above, are discussed in more depth during the next chapter where we also present the solutions/changes that we implemented for the design of the revised version.

Chapter 5

Design and Development

5.1 Framework

As discussed in chapter [2], the study aimed to produce a more usable version of KeePass2 for the average user. The results from the two usability studies performed in chapters [3] and [4] suggested that modifying the existing UI of the Original KeePass2 software would not be sufficient to produce the required result. The reason for this is that KeePass2 was developed using .NET framework and Windows Forms for the GUI(Graphical User Interface). As such, to still be able to connect a newly developed, richer GUI to the back-end of the Original KeePass2 required to continue working with the .NET framework but use a newer Windows technology, the WPF(Windows Presentation Foundation) graphical subsystem.

To be able to develop the GUI from scratch, and to allow enough time to test and evaluate the new product, it was decided to completely develop the front-end of the Revised KeePass2 version and also build a ”fake” back-end to mimic a fully functional application for testing purposes. Developing the ”fake” back-end required careful inspection of the Original Source Code in order to allow a connection to the ”real” back-end of the Original.

5.2 Tools

The revised version was developed using the Visual Studio 2017 IDE. The icons used during the development were taken from **flaticons** [`https://www.flaticon.com/`] and re-designed using **GIMP** [`https://www.gimp.org/`] to meet the specific needs of the design. Finally, **metroapps** [`https://mahapps.com/`] WPF package was used to provide us with the flyout control functionality missing from the WPF graphical subsystem.

5.3 Domain Language

One of the most prominent themes emerging through the Think-Aloud study, was that the domain language used in the Original KeePass2 version was confusing for the average user. It failed to induce the correct mental model and made the initial interaction with the application almost unusable. In the following paragraph we outline the changes introduced to the domain language used in the Revised version and summarize them in table [5.1].

During the Cognitive Walkthrough study, it was pointed out by the HCI experts that "**Encrypted Database File**" would be troublesome notation for the average user. The observation was later confirmed with the Think-Aloud study. It was decided to introduce a much more familiar term which implied both the notions of a *container* and *safety*. The **safe** term could be also easily associated with the term Master Password.

The **Group** term, although not complex, appeared to suggest to the participants that different groups possessed different functionality which was not the case. It was decided to be replaced with folder, since that was its only functionality to begin with in the Original version. This notion was reinforced with choosing a generic folder symbol to accompany the "folder" term as shown in the following section in table [5.5].

Original		Revised
Encrypted Database file	→	**Safe**
Composite Master Key	→	**Master Password**
Entry	→	**Credential**
Group	→	**Folder**

Table 5.1: Domain Language Changes

It has to be clarified that **Composite Master Key** is not the same as Master Password. The original version allows the user to form a **Composite Master Key** by combining a Master Password and/or a Key File and/or a Windows User Account. None the less, unless a user has any computer security knowledge, he will always choose to use only a Master Password and ignore the additional options according to the results of our study. As such, we decided to hide the other 2 options into an advanced settings control, and replace the Composite Master Key with Master Password to reduce any confusion.

5.4 Visual Metaphors

Safe

The Think-Aloud results (see chapter [4]) have shown that the most challenging aspect of using KeePass2 successfully was realizing the existence of the preliminary step of creating a new encrypted Database file. We hoped to remedy this with the introduction of the familiar icon and notion of **Safe** (see table [5.2]). Including the safe symbol across most of the GUI screens of the revised version of KeePass2 we hoped to induce the correct mental model to the average user.

Credential

For the credential symbol we decided to maintain the **Key** icon. Care was taken to maintain the icon's appearance consistent through out the new GUI, a feature absent from the original KeePass2 as can be seen in table [5.3].

Password and Master Password

The Original KeePass2 version used the key symbol inconsistently for all three concepts, password, Master Password and Entry. On the contrary we chose to depict password and Master Password with the symbols shown in table [5.6].

Other

For various actions like, `delete, new, open, add, edit` we tried to use consistent, familiar symbols as seen in all of the ”visual metaphors” tables.

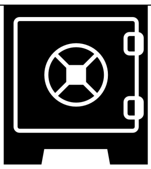
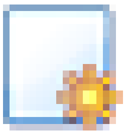
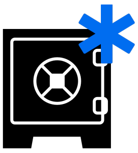
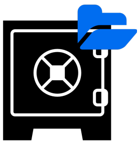
Control Name	Original		Revised	Control Name
Encrypted Database	No Visual	→		Safe
New ...		→		New Safe
Open ...		→		Open Safe

Table 5.2: Visual Metaphors - **Encrypted Database** vs **Safe**

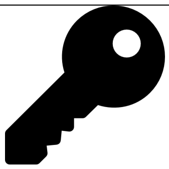
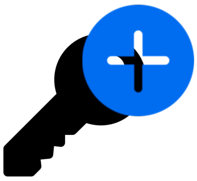
Control Name	Original		Revised	Control Name
Entry	varied	→		Credential
Add Entry		→		Add New Credential

Table 5.3: Visual Metaphors - Entry vs Credential

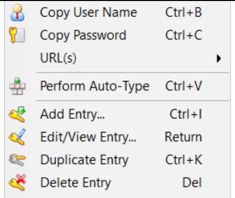
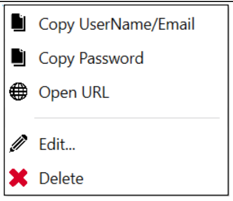
Control Name	Original		Revised	Control Name
Entry Controls	 Copy User Name Ctrl+B Copy Password Ctrl+C URL(s) Perform Auto-Type Ctrl+V Add Entry... Ctrl+I Edit/View Entry... Return Duplicate Entry Ctrl+K Delete Entry Del	→	 Copy UserName/Email Copy Password Open URL Edit... Delete	Credential Controls

Table 5.4: Visual Metaphors - Entry Controls vs Credential Controls

5.5 Work Flow

In this section we describe changes implemented to the user work flow implemented while developing the revised version. The subsections below explain the different design decisions taken compared to the Original KeePass2 version.

5.5.1 Tab Control

The tab control capability was introduced to give the flexibility to the user to open multiple safes at once. The title of each of the opened tabs reflects the state of the application. For example in figure 5.1 there 2 opened tabs and they are both in the welcome state. Once a pre-existing safe is opened or a new one is created, the tab title changes to that safes name.

The introduction of the tab control also allowed us to enclose all the safe relevant controls under the tabitem content, while allowing for global controls that target the application as a whole to be separated onto the top right corner of the screen. This gives the user the freedom to access useful information about the application, whether it is password security advice or application settings, at any time and state of the tab state.

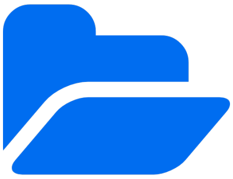
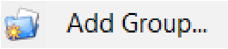
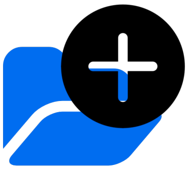
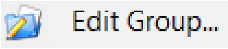
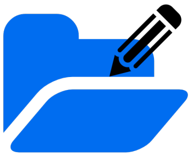
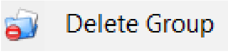
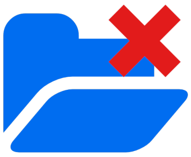
Control Name	Original		Revised	Control Name
Group	varied	→		Folder
Add Group ...		→		Add New Folder
Edit Group ...		→		Edit Folder
Delete Group		→		Delete Folder

Table 5.5: Visual Metaphors - Group vs Folder

5.5.2 Wizards

Depicted on figure 5.1, is the Welcome screen the user encounters once he runs the application. In contrast with the Original version, we have restricted/abstracted most of the complexity leaving the user with only 2 basic choices. This aims to make the user aware of the implicit step of creating an encrypted container before attempting to start adding his/her credentials. The ”New Safe” button leads to a 2 page long wizard while the ”Load Safe” button leads to a 1 page long wizard. Both guide the user into creating/opening a safe which ultimately leads to the Main UI Screen.

The wizard like functionality used for the New/Open Safe controls was designed to provide consistency through out the user’s work flow. Furthermore, it allows the wizards to be easily extended for additional functionality and user guidance with the simple addition of a WPF page control.

As can be seen in figures [5.2, 5.3], the page header clearly states the progress status of the wizard. Also, the series of subtasks a user needs to complete in order to successfully progress through the wizards is enumerated on each page’s main body. Finally on the bottom of the page the navigation controls are used to progress or regress through

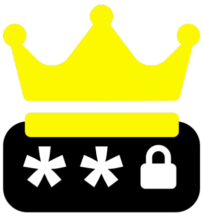
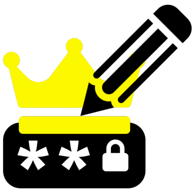
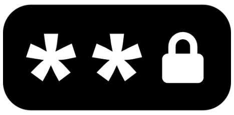
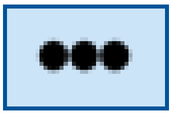
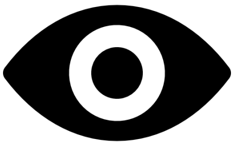
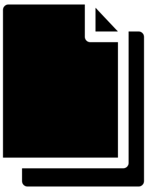
Control Name	Original		Revised	Control Name
Composite Mater Key	No Visual	→		Master Password
Change Master Key ...		→		Edit Master Password
Generate Password		→		Generate Password
Reveal Password		→		Reveal Password
No Control	No Visual	→		Copy Password

Table 5.6: Visual Metaphors - Password Controls

the process where appropriate.

Below the header, a section is reserved for some minimal text that aims to assist first time users. This text was included in the design with some hesitation, since various sources of UX design state that users tend to ignore text. Care was taken to make the text short and enhance its readability with icons where applicable.

Finally, it needs to be stressed out that both New/Open Safe wizards utilized the file explorer functionality of the Windows OS. During the New Safe wizard the functionality was tweaked to allow the selection of just the folder location and not the file name and type extension normally asked by the file explorer. The reason for this was to eliminate the ambiguity that existed in the Original version which asked the user to name both the encrypted Database file through the file explorer and name the Database it self

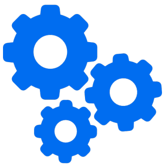
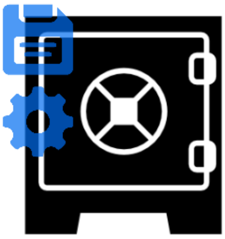
Revised	Control Name
	Help
	KeePass2 Settings
	Safe Controls - Safe Settings and Save button

Table 5.7: Visual Metaphors - Additional Controls

through the Database settings window.

5.5.3 Flyout Control

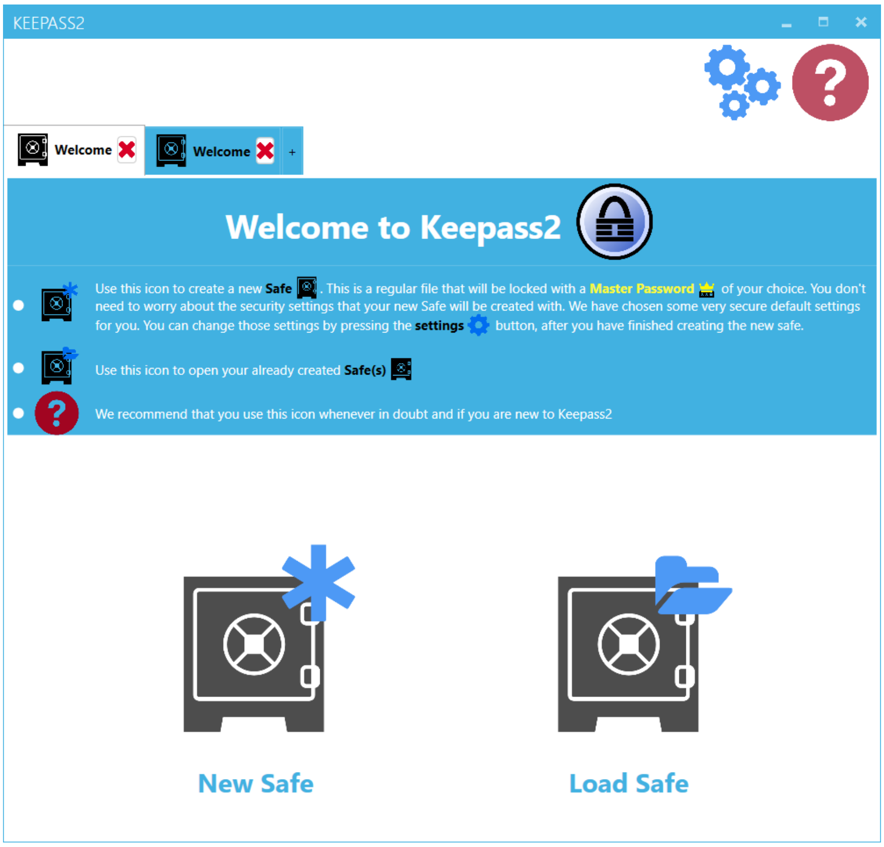
The flyout control allows the existence of wizard like processes to take place in the same screen the user is already working on. Using flyout controls aims to increase the efficiency of use and provide more integration of the various functions of the system.

The pages displayed through the flyout controls retain the same appearance as the wizard like pages used to create the New/Open Safe wizards mentioned above. Following are the main processes presented through flyout controls:

Add New Credential

The ”Add New Credential” wizard (see figure [5.5]) was split into a 2 step process. The rational behind this decision was two fold. Firstly, it aimed to reflect the real world state where most of the online services usually ask for a User-Name/Email and a Password. So we prompted the user to provide those 2 information first. Secondly, this split allows to explain and provide guidance for the correct completion of each of the 5 fields required by the process.

Another important change that was applied to the revised version was the explicit use of the ”User-Name/Email” notation is contrast with the ”User Name” notation used by the original version.



Welcome to Keepass2

- Use this icon to create a new **Safe**. This is a regular file that will be locked with a **Master Password** of your choice. You don't need to worry about the security settings that your new Safe will be created with. We have chosen some very secure default settings for you. You can change those settings by pressing the **settings** button, after you have finished creating the new safe.
- Use this icon to open your already created **Safe(s)**
- We recommend that you use this icon whenever in doubt and if you are new to Keepass2

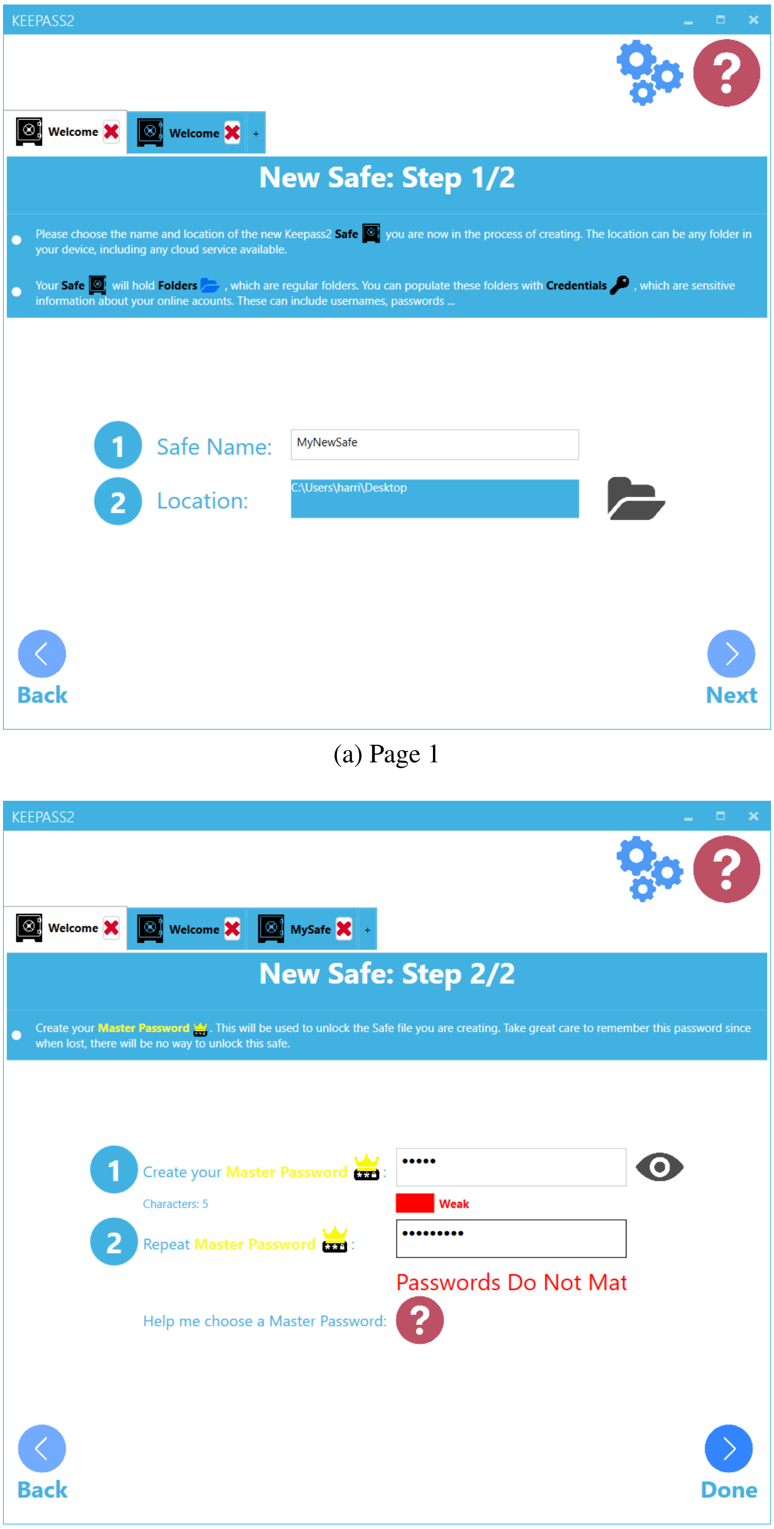
Figure 5.1: Welcome UI Screen

Beside the password field the updated password controls have been also introduced. The ”reveal password” control is now depicted with the eye symbol and is followed by the ”generate password” control depicted by the password symbol. An additional third control, the ”copy password” control, is added to allow users to copy and paste the password to their online accounts without having to reveal the password. This will increase security against malicious screen capture software and/or reduce exposure of the password to any external recording devices.

Finally, we have to note that the URL explanation and advice appears rather lengthy and could potentially be dangerously ignored by the user. Due to the non-web browser implementation of KeePass2 this was unavoidable. A future fix for this will be discussed though in the final chapter.

Edit Credential

The ”Edit Credential” page (see figure [5.6]) architecture was kept similar to the original’s, while following the standard appearance and configuration that was applied to the wizard pages of the revised version. As always care is taken to match the icon on the left of the page’s header with the icon of the control that induced the corresponding effect.



New Safe: Step 1/2

- Please choose the name and location of the new Keepass2 **Safe** you are now in the process of creating. The location can be any folder in your device, including any cloud service available.
- Your **Safe** will hold **Folders** , which are regular folders. You can populate these folders with **Credentials** , which are sensitive information about your online acounts. These can include usernames, passwords ...

1 Safe Name: MyNewSafe

2 Location: C:\Users\harri\Desktop

(a) Page 1

New Safe: Step 2/2

- Create your **Master Password** . This will be used to unlock the Safe file you are creating. Take great care to remember this password since when lost, there will be no way to unlock this safe.

1 Create your Master Password : •••••

Characters: 5 Weak

2 Repeat Master Password : •••••••••

Passwords Do Not Mat

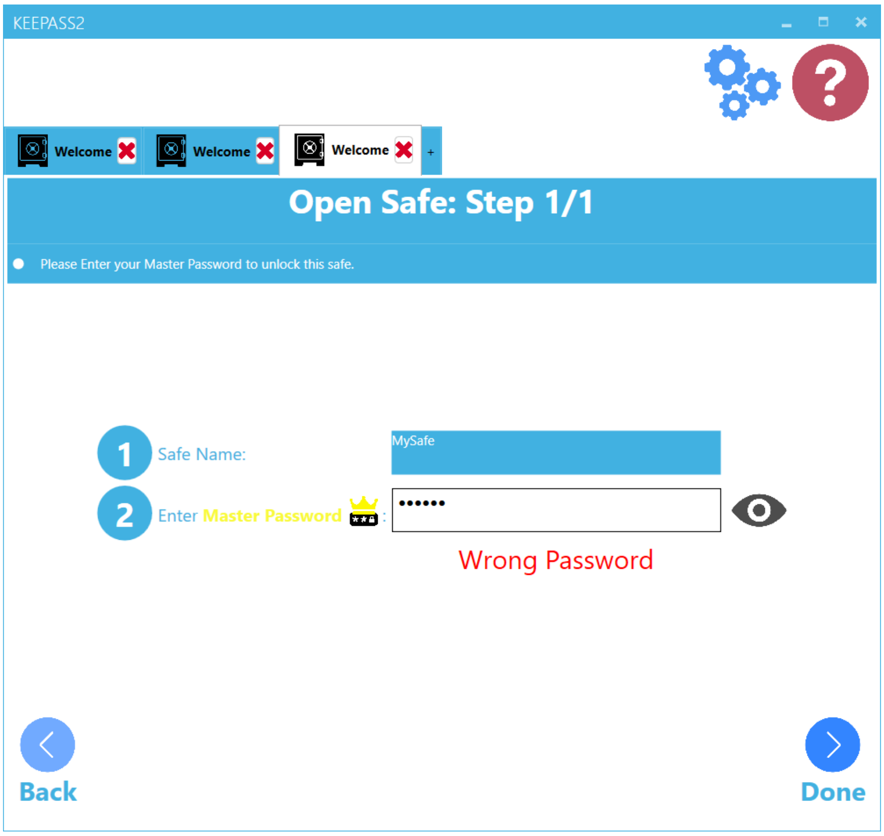
Help me choose a Master Password:

(b) Page 2

Figure 5.2: New Safe Wizard

Folder Controls

The folder controls were never tested, neither during the Cognitive Walkthrough nor the Think-Aloud study since they were not involved in any of the identified main tasks.



KEEPASS2

Welcome
Welcome
Welcome

Open Safe: Step 1/1

● Please Enter your Master Password to unlock this safe.

1 Safe Name: MySafe

2 Enter Master Password : ••••••

Wrong Password

Figure 5.3: Open Safe Wizard

They were designed again to follow the general wizard page appearance and configuration (see figure [5.7]).

Safe Settings

In the original version design, changing the Composite Master Key was kept separate from the database settings. In the design of the revised version however, it thought best to encapsulate both settings under the ”Safe Settings” control. The ”Safe Settings” page then provides a collection of various controls such as the ones shown in figure [5.8(a)]. This page can be then populated with additional features in future releases.

Navigating to the ”Change Master Password” page, through the ”Edit Master Password” control, the user can change the safe’s master password. Additional to the 2 ”New Master Password” fields, we ask the user to provide the safe’s old Master Password and only then to allow him to successfully update the Master Password. This feature was missing from the Original KeePass2 and was added to increase security.

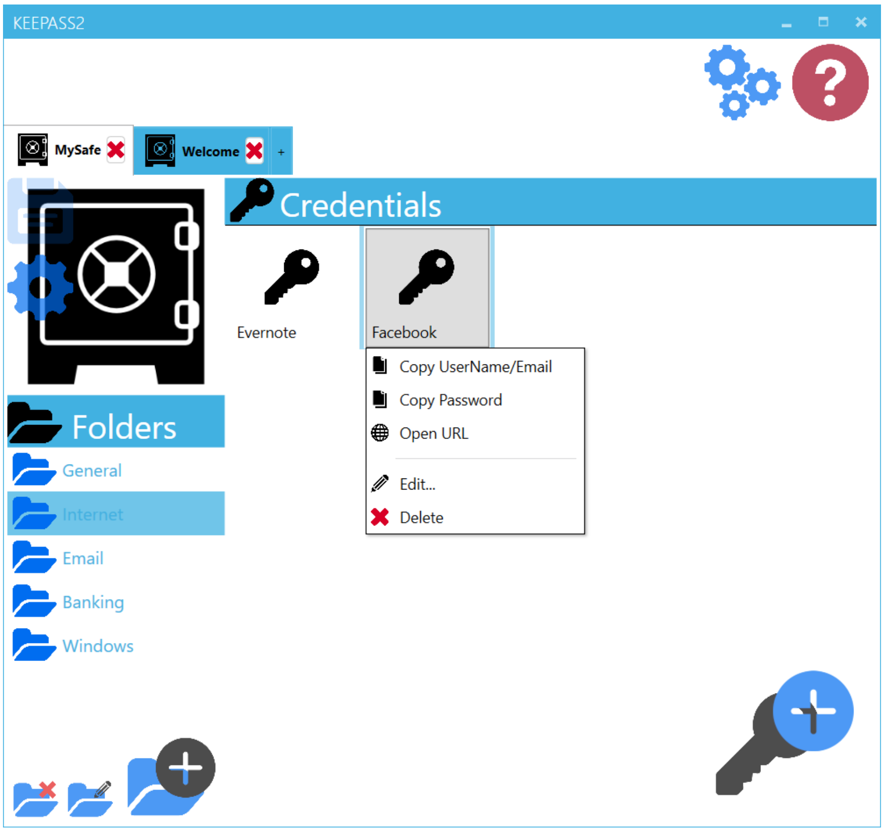


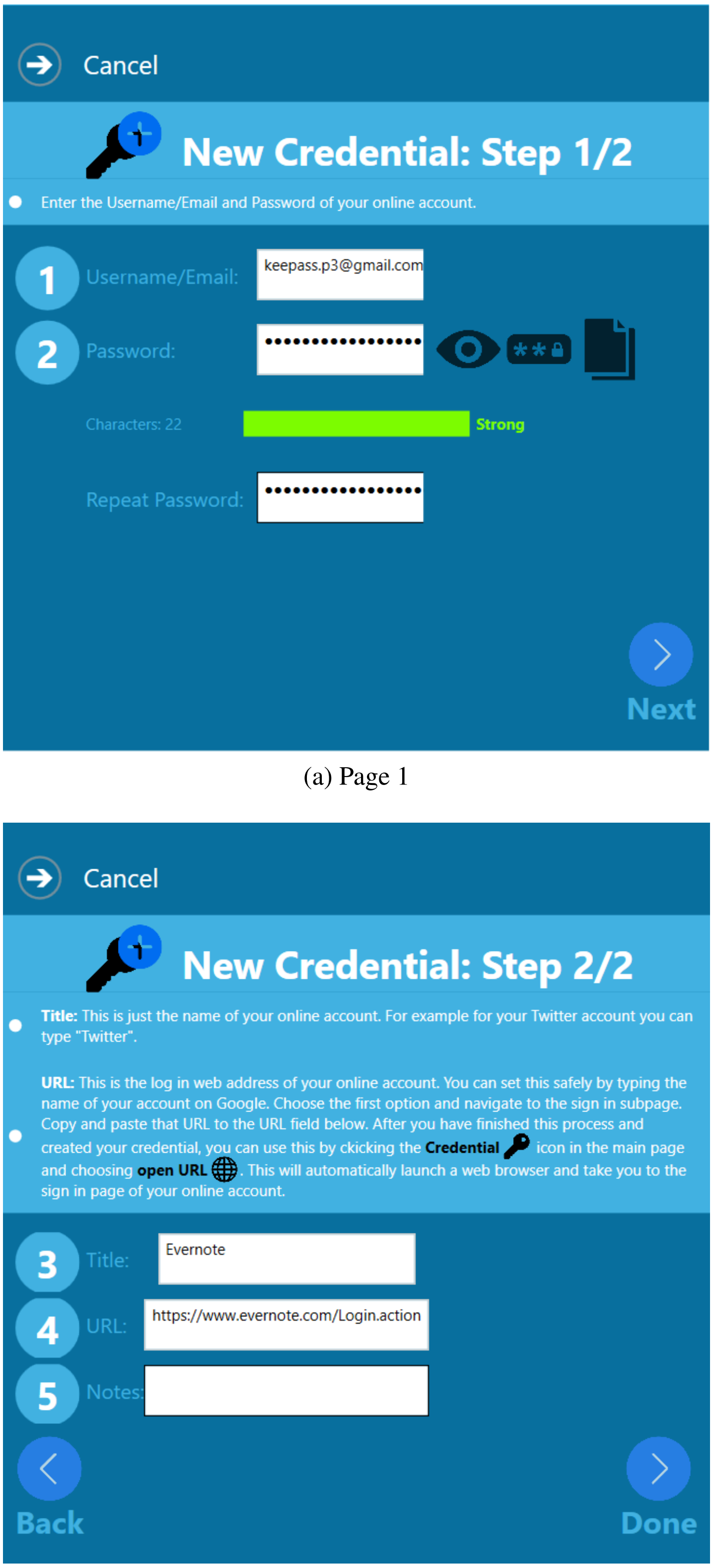
Figure 5.4: Main UI Screen - Facebook credential clicked to display Credential Controls drop-down menu

Help

The global ”Help” button is used to allow the user to access useful information about various functions of the revised version of KeePass2. It was implemented to provide users with a general description of the functionality and work flow of the application. Depicted in figure [5.9] is the ”How to use KeePass2” page which provides a schematic (along with some explanatory text) of the KeePass2 work flow. We advice the reader to actually run the application in order to view the complete schematic using the scroll bar.

5.6 Password Meter

The password meter, unlike other visuals was not changed drastically. Nonetheless, the character field was written in full, the bit strength information was removed and a word was additionally added to characterize password strength. Adding the textual characterization of the password strength hoped to improve password strength awareness which was recorded as poor during the Think-Aloud of the Original version.



New Credential: Step 1/2

- Enter the Username/Email and Password of your online account.

1 Username/Email: keepass.p3@gmail.com

2 Password: ••••••••••••••••••

Characters: 22 Strong

Repeat Password: ••••••••••••••••••

(a) Page 1

New Credential: Step 2/2

- **Title:** This is just the name of your online account. For example for your Twitter account you can type "Twitter".
- **URL:** This is the log in web address of your online account. You can set this safely by typing the name of your account on Google. Choose the first option and navigate to the sign in subpage. Copy and paste that URL to the URL field below. After you have finished this process and created your credential, you can use this by ckicking the **Credential** icon in the main page and choosing **open URL** 🌐. This will automatically launch a web browser and take you to the sign in page of your online account.

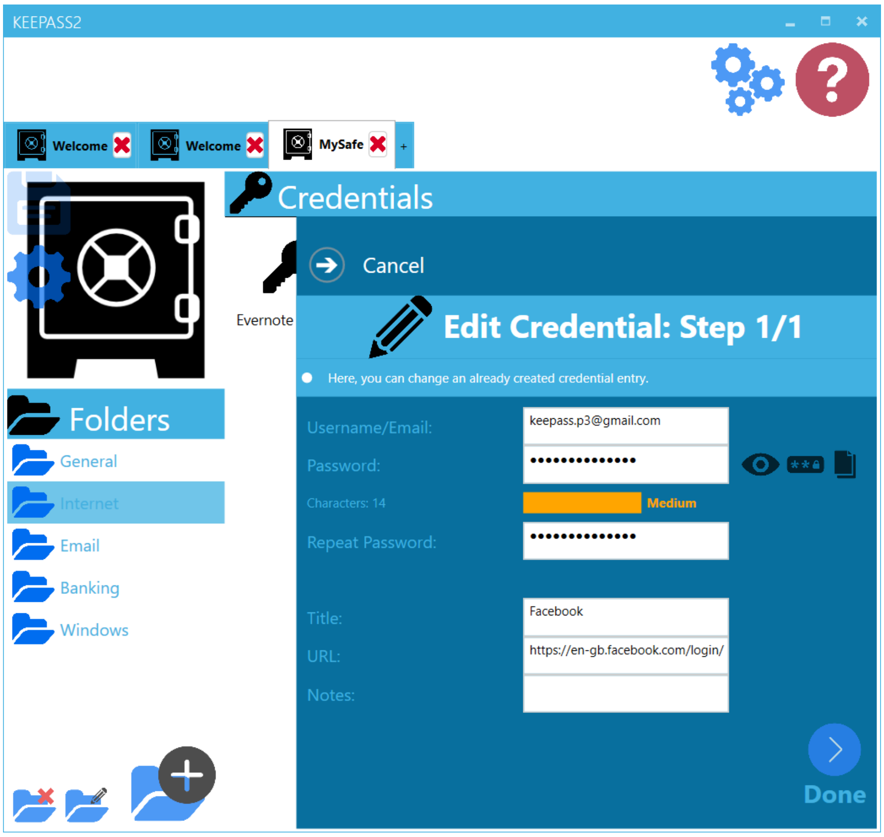
3 Title: Evernote

4 URL: https://www.evernote.com/Login.action

5 Notes:

(b) Page 2

Figure 5.5: Add New Credential Wizard - Only flyout is shown



Here, you can change an already created credential entry.

Username/Email: keepass.p3@gmail.com

Password: ••••••••••••••

Characters: 14 Medium

Repeat Password: ••••••••••••••

Title: Facebook

URL: https://en-gb.facebook.com/login/

Notes:

Figure 5.6: Edit Credential Wizard - The complete GUI of the application is shown

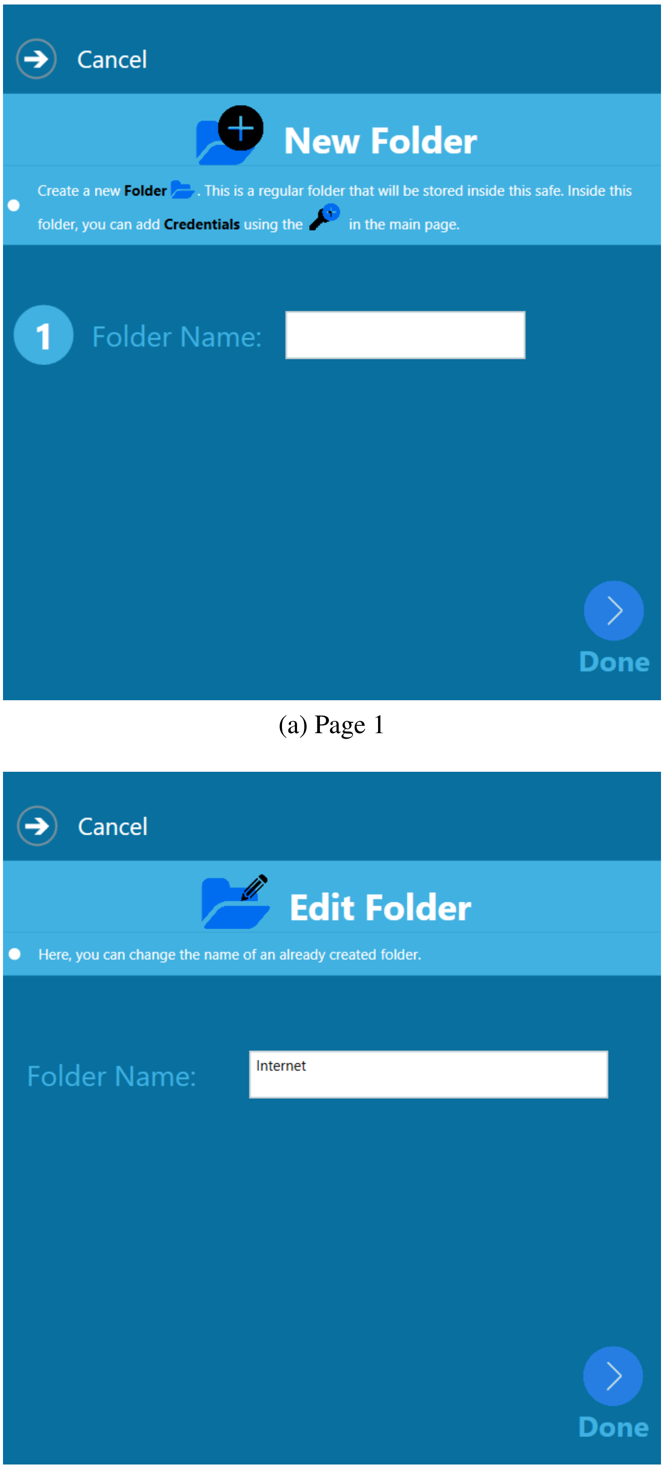
5.7 Miscellaneous

Save

The save functionality of the Original version was a bit of a controversial area. The CW experts thought that changes to the database should be saved automatically and that manual saving is an old and confusing feature. During the TA however, the manual saving feature allowed the users to recover from errors easily although it has to be stated that they were surprised with the manual saving functionality as expected. Since the manual saving functionality did not lead to any erroneous situations during the TA, but allowed for error recovery it was decided to deploy the same functionality to the revised version. The save button however was designed to be more visible than it was in the original version.

Credential Controls

As discussed in earlier sections, the grid-box presentation of the contents of the database(and/or groups) was extremely confusing to the users. We decided to present the content of the safe(and folders) as a collection of items as shown in figure [5.4]. Additionally, we



- Create a new **Folder**. This is a regular folder that will be stored inside this safe. Inside this folder, you can add **Credentials** using the in the main page.

Folder Name:

(a) Page 1

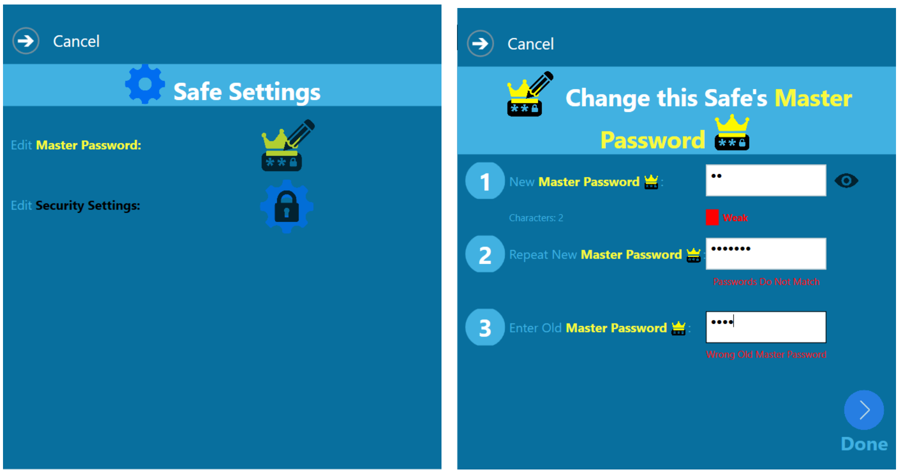
- Here, you can change the name of an already created folder.

Folder Name: Internet

(b) Page 2

Figure 5.7: Folder Controls - Only flyout is shown

restricted the revoking of the credential controls to only one action, which was the ”selection” of the credential with the mouse left click. This invoked a drop-down menu to appear from the selected item as shown in figure [5.4]. In contrast the original version allowed right-clicking, selecting and double-clicking. Although more flexible, it was



1 New Master Password: ••

Characters: 2

Weak

2 Repeat New Master Password: •••••••

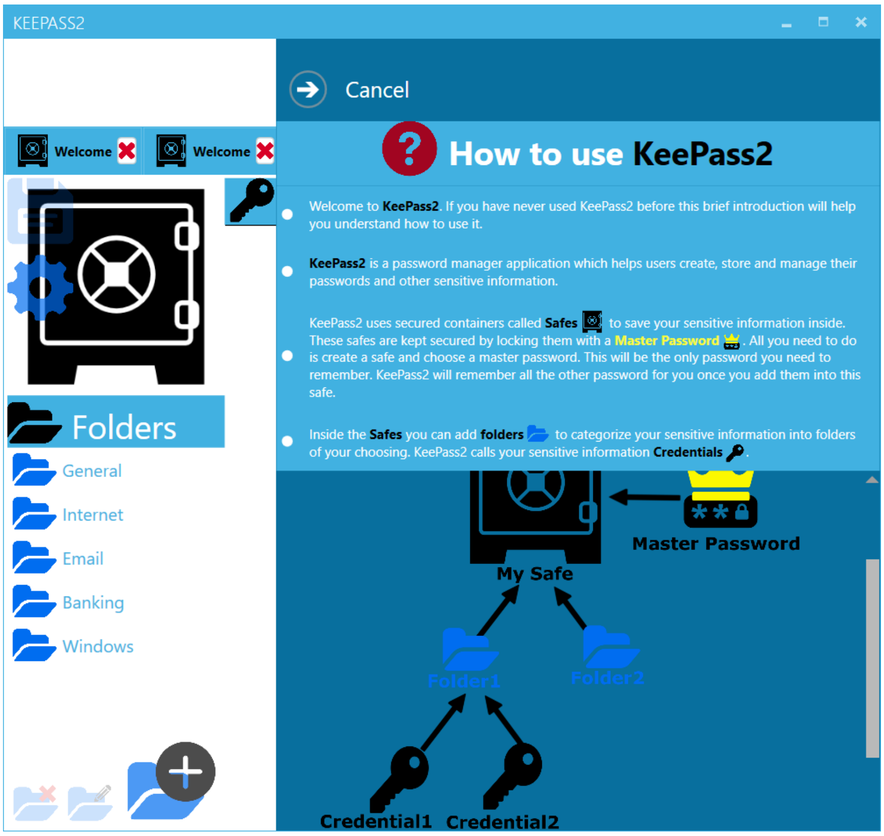
Passwords Do Not Match

3 Enter Old Master Password: ••••

Wrong Old Master Password

(a) Safe Settings page (b) Edit Master Password page

Figure 5.8: Safe Settings Wizard - Only flyout is shown



- Welcome to **KeePass2**. If you have never used KeePass2 before this brief introduction will help you understand how to use it.
- **KeePass2** is a password manager application which helps users create, store and manage their passwords and other sensitive information.
- KeePass2 uses secured containers called **Safes** to save your sensitive information inside. These safes are kept secured by locking them with a **Master Password**. All you need to do is create a safe and choose a master password. This will be the only password you need to remember. KeePass2 will remember all the other password for you once you add them into this safe.
- Inside the **Safes** you can add **folders** to categorize your sensitive information into folders of your choosing. KeePass2 calls your sensitive information **Credentials**.

Figure 5.9: Help control

shown that this was the main reason that caused users to fail to locate the controls.

The ”Copy User Name” control of the original version was replaced with ”Copy UserName/Email” in order to reflect the respective credential ”UserName/Email” field. Finally, the ”Edit/View Entry ...” control was changed to ”Edit ...” in order to eliminate any unintended use of this control while promoting the use of the top three controls to log in to the respective account (see figure [5.4]).

Chapter 6

Think-Aloud(TA) Study - Revised Version

The methodology, study design and study setup used to perform this study was identical to the one used to plan the TA study on the Original version of KeePass2. It is described in chapter [4]. In this chapter, we provide and discuss the results of the TA study on the Revised KeePass2 version and finally compare the results of the 2 TA studies.

6.1 Study Results - Revised KeePass2

6.1.1 Pre-Questionnaire Results

Demographics

3/5 participants reported to be female and 2/5 participant reported to male. 2/5 were between the age of 25-30, 1/5 were between 36-40 and 2/5 between 41-45. All the participants reported to have had a BSc in Teaching. 4/5 reported Greek as a native language and fluency in English, while 1/5 reported English as a native language.

Computer and Online Experience

2/5 participants claimed to sign in to their online banking accounts weekly, 1/5 monthly, 1/5 a few times a year and 1/5 never. 3/5 of participants said that they use their work computer to sign in to their accounts weekly, 1/5 daily and 1/5 participants monthly. When asked what operating system do they use, all participants reported to use Windows regularly and 1/5 reported to be using Mac OS also. 3/5 participants reported to use Chrome only as their primary web browser and 2/5 said they use Chrome and IE.

Password Habits

3/5 participants said to have 0-10 online accounts that require password authentication, 1/5 reported 11-20 and 1/5 21-30. The participants that reported having the least online accounts reported to have 1-3 unique passwords across those accounts and the ones that reported more said to have 4-6. (It should be noted that participants were told to discount the passwords with slight permutations as unique). All participants said that they change their passwords only when prompted by the respective service. 2/5 of the participants reported that they both use memory and writing in physical format to remember their passwords, 2/5 reported using memory and digital format and 1/5 only memory. When asked to comment on the security and convenience of those methods their answers gave no meaningful pattern. Finally, none of the participant reported to be using/used before any password manager application.

6.1.2 Think Aloud Results

As discussed in the corresponding section in chapter [4], we present the results in two ways. Calculate a representative rate of success/failure of task completion and identify the major themes by transcribing and coding the participants verbalization during the think aloud sessions. The former are presented in table 6.1 and the later in the following subsection.

For the explanation of the labels used to mark the success/failure of the TA tasks please refer back to Chapter [4], Subsection [4.5.2]

Think Aloud Task Completion						
	Goal	P1	P2	P3	P4	P5
Task 1	Create New DB	F. comp.	comp.	comp.	comp.	comp.
	Add Existing Account	comp.	comp.	failed	comp.	comp.
Task 2	Use Existing Entry	**D. comp.**	comp.	**D. comp.**	comp.	comp.
Task 3	Add New Account	comp.	comp.	comp.	comp.	comp.
Task 4	Edit Entry	F. comp.	comp.	F. comp.	comp.	F. comp.
Task 5	Change MP	comp.	comp.	comp.	comp.	comp.

Table 6.1: Task completion results of the 5 participants on the Revised KeePass2 version. DB = Database, MP = Master Password

Task Specific Themes

Task 1

Think Aloud Task Performance			
	Goal	Success Rate	Deviation from Path
Task 1	Create New DB	80%	1.0
	Add Existing Account	80%	
Task 2	Use Existing Entry	60%	1.8
Task 3	Add New Account	100%	1.4
Task 4	Edit Entry	40%	0
Task 5	Change MP	100%	0
Mean		77%	0.84

Table 6.2: Task performance measures on the revised KeePass2 version. - DB = Database, MP = Master Password

- **Create New DB** - 4/5 participants were able to quickly understand that creating a new safe by no means was the same as creating the Gmail account. They reported during the TA that ”I will firstly create a safe and then add my Gmail account”. 2 of those used the ”questionmark” control before they were completely sure about progressing with the ”New Safe” control. The 1/5 participant that had a false completion however did create a new Safe using his Gmail account credentials, but realized his mistake as soon as he reached the end of the ”New Safe” wizard.

- **Add Existing Account** - Removing the Grid-box and Grid-box titles seem to alleviate most of the participants confusion. Additionally the large, prominent key with add symbol on top was an obvious bet for the participants and 4/5 of them went straight for that. The term ”Credential”, although explained in the beginning of the TA briefing stage seemed to be somewhat alien to the 4/5 participants that were native Greek speakers. This did not stop them however from completing the task. It has to be further investigated whether this term is common for English speakers in general. The 1/5 participant that did not complete this task was unable to realize that to activate the ”Add New Credential” button he/she had to select a folder for the credential to be saved in.

Task 2

- **Use Existing Entry** - Locating both an existing credential by navigating through the different safe folders and locating the controls for using that credential was very easy for all 5 participants. However, only 3/5 used the intended way of using these controls which was 1) opening the URL(or navigating to the official website of the Evernote service), 2) Copying/Pasting the Username/Email of the Evernote credential, 3) Copying/Pasting the Password of the Evernote credential. 2/5 of the participants that used the Edit Credential control and revealed the password field in order to copy/paste it were marked as Dangerous Completion.

Task 3

- **Create New Account** - The controls for this task were located again very easily by all participants. Sadly, non of them used KeePass2 to create a random password for their Facebook account and instead chose their own. Also 3/5 of the participants ignored the URL advice given in the 2/2 step of creating a new Credential and left that field blank. One of those participants said ”I’ve already stopped reading when I reached the URL explanation”.

Task 4

- **Edit Entry** - Although the ”Edit ...” control was easily found by the participants, 3/5 of them failed to realize that changing the information on KeePass2 would not have changed their Evernote account credentials.
  On the other hand, we were able to eliminate the Dangerous Completions for this task due to the updated password meter visual used. Although the participants did not use the password generator feature that KeePass2 provides, they none the less tried to append characters until they reached the ”Strong” word in the password meter. This was not the case with the original version where participants appended only a few characters to increase the strength only a little, producing a still weak password.

Task 5

- **Change MP** - This step was not immediately obvious to all participants since the Edit Master Password control was re-factored into the Safe Settings control. Nonetheless, they were able to find the control once they tried the Safe Settings button.

General Themes

Save Functionality
Since we decided to maintain the manual save functionality to the revised version, it was not surprising that again participants noted that they expected the changes to be saved automatically. Nonetheless, when they tried to exit without saving, they were prompted by the system to save and they were able to identify the save control with ease.

6.1.3 Post-Questionnaire Results

Participants’ Satisfaction

Participants satisfaction was captured as mentioned earlier using the System Usability Scale (SUS) [Brooke et al., 1996]. The results are summarized in table 6.3. P2’s results were quite interesting both during the TA and during the discussion at the end. Although he completed the tasks almost perfectly and faster than any participant, he started focusing and commenting about more advanced attributes of the system. He explained that he did not give higher scores to the SUS scale since he found the solution to reduce productivity. He commented during the TA, ”Ohh, do I have to do this all over again ...” referring to the process of adding a new credential. He went on to say” It would be nice if it did this automatically for you ...” referring again to the add new credential process.

SUS Results								
	P1	P2	P3	P4	P5	Mean	Sd	SE
Score	52.5/100	77.5/100	65/100	82.5/100	67.5/100	69	11.67	5.22

Table 6.3: SUS scores for the Original KeePass2 version - Sd: Standard deviation, SE: Standard Error

Participants’ Mental Model

When asked about the location keepass2 saved the database file it created, 3/5 participants had formed the wrong impression. All three of them thought that the application has saved the credentials online. the other 2/5 reported that the application saved the credentials on the computer. When asked about losing their master password, 4/5 participants again erroneously thought that they could recover their master password by contacting the KeePass2 services. Only 1/5 reported correctly that his credentials will be lost for ever. Surprisingly when asked about a hacker stealing their database file, 3/5 reported that their database will be hacked where as the remaining 2 said they wouldn’t know. Finally when asked about the strength of the passwords used during the study (the ones the researcher had pre-chosen), 4/5 stated that the passwords were weak and 1/5 said they hadn’t noticed.

It was interesting to observe that participant P2 was the only one that actually developed the correct mental model of KeePass2 since he answered correctly all 4 of the questions. In that, when evaluating the changes in user’s mental models the revised version induced on users, we will not attribute his/her improvement to our design. Nonetheless, his success rate, SUS and deviation-from-path scores will be measured normally.

6.2 Evaluation

In this section we use the results from the TA study on the original version as a control and compare with the results of the TA on the revised version to evaluate the new solution developed.

Task	Goal	Success Rate	
		Revised	Original
Task 1	Create New DB	80%	20%
	Add Existing Account	80%	20%
Task 2	Use Existing Entry	60%	20%
Task 3	Add New Account	100%	40%
Task 4	Edit Entry	40%	0%
Task 5	Change MP	100%	80%

Table 6.4: Success rate comparison - Revised vs Original

Demographics and Ecological Validity

Looking at the demographics of both samples, there does not seem to be a significant difference between the 2 groups. The participants of both groups belonged into similar age groups, they had similar job descriptions and computer-password experience and habits. In this way any changes identified between the 2 versions can be attributed to the differences in their design and not due to difference in the 2 groups.

It has to be noted however that participant 2 from the revised version study seemed to behave more confidently than the other participants and his/her performance during the TA was better than all the other participants. The same observation was seen by looking at his mental model results where he answered all 4 of the questions correctly. This suggests that this participant could have been more skilled and experienced with using computers than the rest of the participants. This however was not considered as reason to remove him as an outlier from the study.

Finally, it should be acknowledged that statistical significance with samples of size 5 is very unstable and that is why we refrain from using the quantitative results of the TA studies by them selves.

Success Rates

As shown in table 6.5, there has been a significant overall increase in task completion between the two versions. Also shown in table 6.4, there have been substantial increases in success rates for each individual tasks. Although the numbers by them selves seem very appealing care must be taken to consider the qualitative results of the studies along with these percentages. For instance Task 4, although it displayed substantial increase, its TA results are still alarming since participants were not able to comprehend that changes in the password manager did not automatically affect the respective account credential.

Average Success Rate		Average Deviation		SUS score	
Original	Revised	Original	Revised	Original	Revised
30%	**77%**	2.28	**0.84**	41.5	**69**

Table 6.5: Comparison of TA results

Deviation from Path

Deviation from path also received substantial improvement. The decrease in Deviation from the intended path can be an indication of better work flow design. Again these results however should be taken with a grain of salt, since the revised version did not implement the complete functionality of the original and so its interface was less crowded.

SUS score

SUS scores can be used to reflect the overall usability of the software and the user satisfaction. As shown in table 6.5, there was again an increase in SUS scores. A product that scores above 64 is considered usable according to [Brooke et al., 1996].

Mental Model

Unfortunately the results from the mental model of the 2 groups did not follow the same improvement as other areas. We were able however to make the participants aware of the strength of the passwords used during the study by including a verbal description to the password meter visual shown in the design and development chapter. Although participant 2, of the revised version had a perfectly correct mental model, his increased performance was not attributed to our design but rather were linked to his increased computer skill and awareness that he displayed during the TA.

Chapter 7

Conclusion

7.1 Summary

The CW and TA studies found the Original Version of KeePass2 unusable in the hands of the average user. This was attributed to dad **Visual Metaphors**, **Work Flow** and **Domain Language** used for its design. The revised version was mostly successful in correcting those issues and producing a more usable software. Although some progress was observed for the users' mental model, both versions seem to fail to induce the complete and correct mental model to its users. This will definitely lead to erroneous situations, for the users that do decide to continue to use either versions. We believe that this can be attributed to the type of password manager KeePass2 was developed to be, rather than its UI design. Being a desktop application and a local password manager and not a web browser plug in, it is constrained to more ”manual” processes for its correct usage by the user. Storing the correct URL for a credential entry, choosing a password for a new account, saving credentials into encrypted local files are a few of the processes that have to be manually performed by the user and are hard to automate since KeePass2 has no direct access to a browser.

Not surprisingly, we have observed that abstracting irrelevant and/or rarely used and/or security dense options and controls increased the usability of the software. Doing so it allowed the user to focus on the task at hand and complete it with much less mental effort and strain. On the other hand, bringing other concepts like the encrypted container more into focus but with the appropriate terminology (example encrypted database VS safe) is beneficial to the users work flow and again decreases mental effort and frustration.

7.2 Limitations

Our study of course doesn't come without limitations. Our sample space was limited to teacher stuff of Cypriot Primary schools and although being a closer match to the average user than sampling from university students, it can definitely be expanded to include a wider range of people. Furthermore, as discussed previously, KeePass2 was found to have basic usability issues and thus our focus was shifted to first remedy those issues. In that, we were not able to uncover/observe more subtle usable security related aspects of KeePass2 like the PM's performance against phishing, key-logging, and other social engineering attacks. Although not a direct limitation to the study's observations, the revised version remains to be fully connected to the back-end source code of the original version.

7.3 Future Work

There are various aspects of this project that could be expanded and quite a few different directions that can be explored.

Firstly, the new developed User Interface needs to be connected to the back-end of the original KeePass2 application and ensure that the connection is secured. Then various aspects of this solution can then be easily expanded or tweaked so more specific aspects of password authentication user habits can be explored in more depth. For example different password strength meter can be used to explore their effects on password creation by average users and different password generators can be tested and evaluated. Several aspects of the system can also be easily modified in the current design, for example the number and length of the various wizard like processes can be varied or the amount of textual advice given to the users can be changed to study their effects on users.

It will be very interesting to implement an automated solution to assist users with choosing the correct URL and so avoiding phishing attacks. Our study unsurprisingly, showed that most users ignored the advice on URLs and thought of them as a "secondary" credential information when saving their account credentials in KeePass2. A potential extension to the "Add New Credential" wizard can be a dedicated page that can host a URL explainer service to at least inform the user if he has inserted the correct URL.

A different direction from our study can be taken to compare the various ports and/or extensions developed for the Original KeePass2 over the years.

As discussed through out the study, KeePass2 is a desktop application and a local password manager. This might be one of the reasons that users find it hard to form a correct mental model for it. An attempt to make the application a browser based password manager might be a better direction for a usable alternative.

Furthermore, our study has focused on average users, but it will be as interesting to perform a larger usability evaluation study to uncover how securely existing KeePass2 users are interacting with the application. A study on the security of various types of password manager database formats [Gasti and Rasmussen, 2012] has suggested that KeePass2's database format became insecure when the user maintained the database file in a cloud service. In that, observing how many of the existing KeePass2 users use a cloud service to store their database file would be interesting.

Bibliography

[Adams and Sasse, 1999] Adams, A. and Sasse, M. A. (1999). Users are not the enemy. *Communications of the ACM*, 42(12):40–46.

[Biddle et al., 2012] Biddle, R., Chiasson, S., and Van Oorschot, P. C. (2012). Graphical passwords: Learning from the first twelve years. *ACM Computing Surveys (CSUR)*, 44(4):19.

[Bonneau et al., 2012] Bonneau, J., Herley, C., Van Oorschot, P. C., and Stajano, F. (2012). The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. *Proceedings - IEEE Symposium on Security and Privacy*, pages 553–567.

[Borsci et al., 2013] Borsci, S., Macredie, R. D., Barnett, J., Martin, J., Kuljis, J., and Young, T. (2013). Reviewing and Extending the Five-User Assumption. *ACM Transactions on Computer-Human Interaction*, 20(5):1–23.

[Brooke et al., 1996] Brooke, J. et al. (1996). Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7.

[Burr et al., 2004] Burr, W. E., Dodson, D. F., Polk, W. T., et al. (2004). *Electronic authentication guideline*. Citeseer.

[Chiasson et al., 2006] Chiasson, S., Oorschot, P. V., and Biddle, R. (2006). A usability study and critique of two password managers. *15th USENIX Security ...*, (August):1–16.

[Dhamija et al., 2006] Dhamija, R., Tygar, J. D., and Hearst, M. (2006). Why phishing works. *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06*, (November 2005):581.

[Florencio and Herley, 2007] Florencio, D. and Herley, C. (2007). A large-scale study of web password habits. *Proceedings of the 16th international conference on World Wide Web - WWW '07*, page 657.

[Florêncio et al., 2016] Florêncio, D., Herley, C., and Van Oorschot, P. C. (2016). Pushing on string: The Don't Care Region of Password Strength. *Communications of the ACM*, 59(11):66–74.

[Gasti and Rasmussen, 2012] Gasti, P. and Rasmussen, K. B. (2012). On the security of password manager database formats. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7459 LNCS:770–787.

[Herley, 2009] Herley, C. (2009). So long, and no thanks for the externalities: the rational rejection of security advice by users. *Security*, pages 133–144.

[Karole et al., 2011] Karole, A., Saxena, N., and Christin, N. (2011). A comparative usability evaluation of traditional password managers. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6829 LNCS:233–251.

[Mathematical, 1910] Mathematical, a. (1910). Model of the Finding of Usability Problems. *Transport*, pages 206–213.

[McCarney, 2013] McCarney, D. (2013). Password managers: comparative evaluation, design, implementation and empirical analysis. *Carleton University*.

[McCarney et al., 2012] McCarney, D., Barrera, D., Clark, J., Chiasson, S., and van Oorschot, P. C. (2012). Tapas: Design, Implementation, and Usability Evaluation of a Password Manager. *Proceedings of the 28th Annual Computer Security Applications Conference on - ACSAC '12*, pages 89–98.

[Nielsen, 1995] Nielsen, J. (1995). 10 usability heuristics for user interface design. *Nielsen Norman Group*, 1(1).

[Nielsen, 2012] Nielsen, J. (2012). Thinking aloud: The no1 usability tool. *Nielsen Norman Group*, 1(1).

[Nielsen and Molich, 1990] Nielsen, J. and Molich, R. (1990). Heuristic Evaluation of user interfaces. *CHI '90 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, (April):249–256.

[Reeder et al., 2017] Reeder, R. W., Ion, I., and Consolvo, S. (2017). 152 Simple Steps to Stay Safe Online: Security Advice for Non-Tech-Savvy Users. *IEEE Security and Privacy*, 15(5):55–64.

[Schneier, 2008] Schneier, B. (2008). The psychology of security. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5023 LNCS(4):50–79.

[Shay et al., 2016] Shay, R., Cranor, L. F., Komanduri, S., Durity, A. L., Huh, P. S., Mazurek, M. L., Segreti, S. M., Ur, B., Bauer, L., and Christin, N. (2016). Designing Password Policies for Strength and Usability. *ACM Transactions on Information and System Security*, 18(4):1–34.

[Shay et al., 2010] Shay, R., Komanduri, S., Kelley, P. G., Leon, P. G., Mazurek,

M. L., Bauer, L., Christin, N., and Cranor, L. F. (2010). Encountering stronger password requirements: user attitudes and behaviors. In *Proceedings of the Sixth Symposium on Usable Privacy and Security*, page 2. ACM.

[Ur et al., 2012] Ur, B., Kelley, P. G., Komanduri, S., Lee, J., Maass, M., Mazurek, M. L., Passaro, T., Shay, R., Vidas, T., Bauer, L., Christin, N., and Cranor, L. F. (2012). How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation Blase. *Security'12 Proceedings of the 21st USENIX conference on Security symposium*, pages 5–16.

[Veras et al., 2014] Veras, R., Collins, C., and Thorpe, J. (2014). On semantic patterns of passwords and their security impact. In *NDSS*.

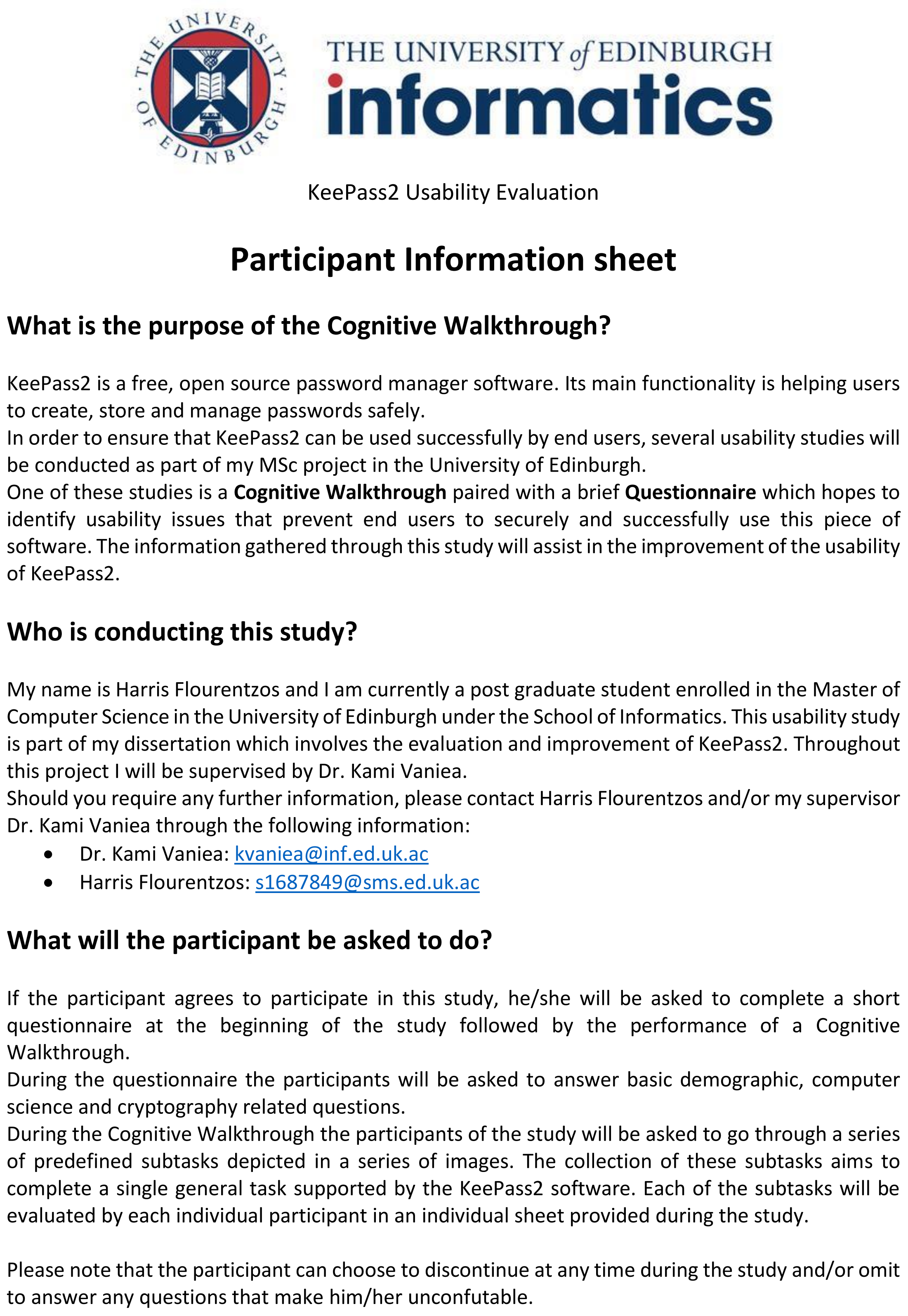
[Whitten and Tygar, 1999] Whitten, A. and Tygar, J. (1999). Why Johnny can't encrypt: A usability evaluation of PGP 5.0. *Proceedings of the 8th USENIX Security Symposium*, pages 169–184.

Appendices

Appendix A

Cognitive Walkthrough

A.1 Consent Form



KeePass2 Usability Evaluation

Participant Information sheet

What is the purpose of the Cognitive Walkthrough?

KeePass2 is a free, open source password manager software. Its main functionality is helping users to create, store and manage passwords safely.
In order to ensure that KeePass2 can be used successfully by end users, several usability studies will be conducted as part of my MSc project in the University of Edinburgh.
One of these studies is a **Cognitive Walkthrough** paired with a brief **Questionnaire** which hopes to identify usability issues that prevent end users to securely and successfully use this piece of software. The information gathered through this study will assist in the improvement of the usability of KeePass2.

Who is conducting this study?

My name is Harris Flourentzos and I am currently a post graduate student enrolled in the Master of Computer Science in the University of Edinburgh under the School of Informatics. This usability study is part of my dissertation which involves the evaluation and improvement of KeePass2. Throughout this project I will be supervised by Dr. Kami Vaniea.
Should you require any further information, please contact Harris Flourentzos and/or my supervisor Dr. Kami Vaniea through the following information:

- Dr. Kami Vaniea: kvaniea@inf.ed.uk.ac
- Harris Flourentzos: s1687849@sms.ed.uk.ac

What will the participant be asked to do?

If the participant agrees to participate in this study, he/she will be asked to complete a short questionnaire at the beginning of the study followed by the performance of a Cognitive Walkthrough.
During the questionnaire the participants will be asked to answer basic demographic, computer science and cryptography related questions.
During the Cognitive Walkthrough the participants of the study will be asked to go through a series of predefined subtasks depicted in a series of images. The collection of these subtasks aims to complete a single general task supported by the KeePass2 software. Each of the subtasks will be evaluated by each individual participant in an individual sheet provided during the study.

Please note that the participant can choose to discontinue at any time during the study and/or omit to answer any questions that make him/her unconfutable.

THE UNIVERSITY *of* EDINBURGH
informatics

KeePass2 Usability Evaluation

Confidentiality

All information collected through the project will be treated confidentially. Only the research team and faculty staff will see participant's names. No names of individuals will be released to any other organization, nor will they be identified in any reports or publications arising from the study.

Use of results

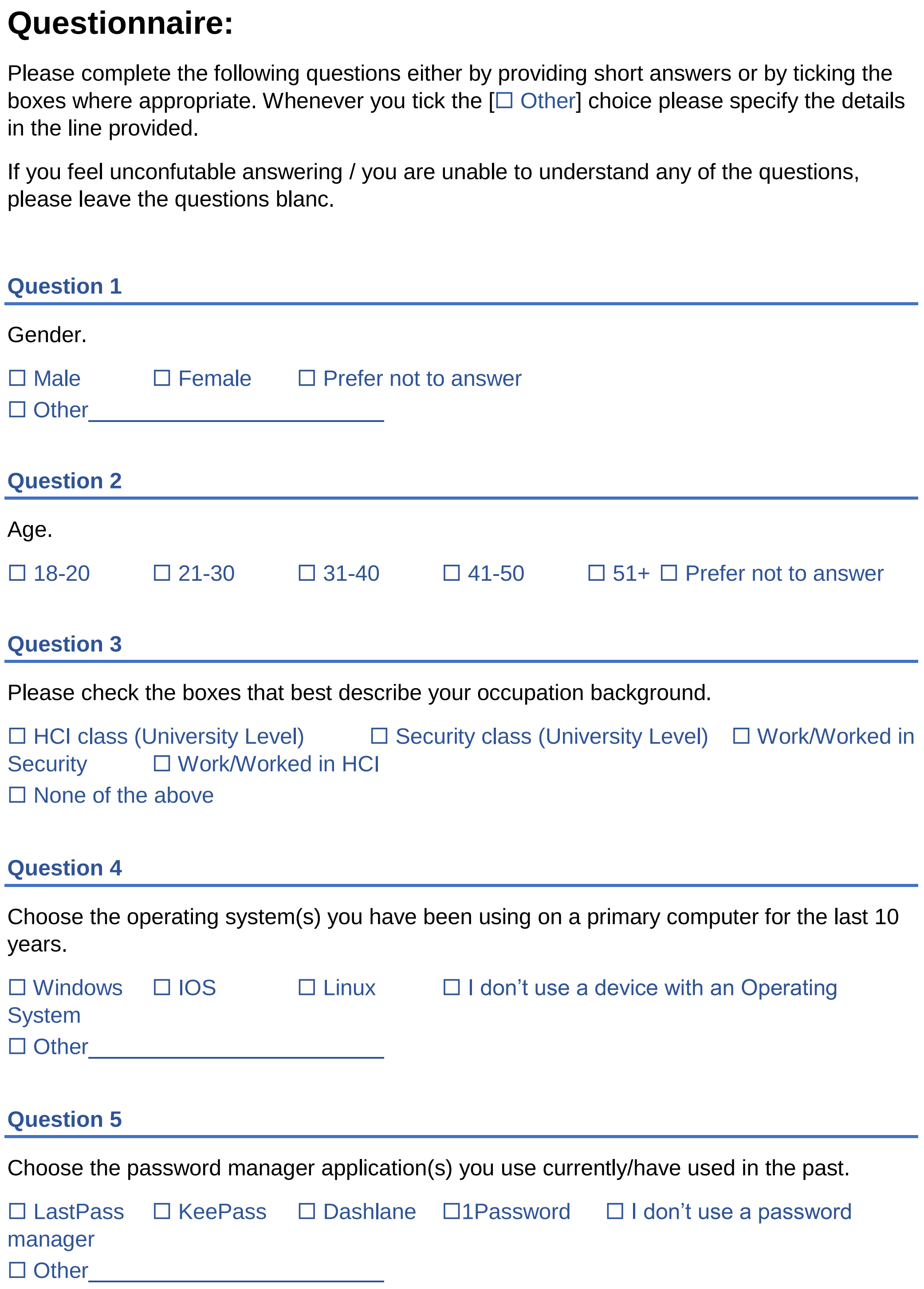
We intend to use the main findings from this research only to understand how usable and secure the KeePass2 software is. Any confidential information will be disposed shortly after the MSc project is concluded.

I hereby acknowledge that I have read and understand what participating in this study entails and I agree with how the information and data I provide will be treated as stated above.

_____________________________ _______________________________
Participant's name Date

____
Signature

A.2 Questionnaire



Questionnaire:

Please complete the following questions either by providing short answers or by ticking the boxes where appropriate. Whenever you tick the [□ Other] choice please specify the details in the line provided.

If you feel unconfutable answering / you are unable to understand any of the questions, please leave the questions blanc.

Question 1

Gender.

□ Male □ Female □ Prefer not to answer
□ Other__________________________

Question 2

Age.

□ 18-20 □ 21-30 □ 31-40 □ 41-50 □ 51+ □ Prefer not to answer

Question 3

Please check the boxes that best describe your occupation background.

□ HCI class (University Level) □ Security class (University Level) □ Work/Worked in Security □ Work/Worked in HCI
□ None of the above

Question 4

Choose the operating system(s) you have been using on a primary computer for the last 10 years.

□ Windows □ IOS □ Linux □ I don’t use a device with an Operating System
□ Other__________________________

Question 5

Choose the password manager application(s) you use currently/have used in the past.

□ LastPass □ KeePass □ Dashlane □1Password □ I don’t use a password manager
□ Other__________________________

Question 6

Please check the box that describes best your knowledge about the following terms.

Bit strength of a password

☐ Expert ☐ Knowledgeable ☐ Heard of it before ☐ Never heard of it

Dictionary Attack

☐ Expert ☐ Knowledgeable ☐ Heard of it before ☐ Never heard of it

A.3 Cognitive Walkthrough Sheet

Cognitive Walkthrough

Persona: Alice

Alice is a 38-year-old real estate agent working for the real estate company, House of Cards. Alice regularly uses a Windows Personal Computer both for her professional and personal life.

Professionally, she uses her pc to manage information about houses as well as sensitive information about her clients' private data. Her day to day usage of her pc involves using Microsoft Office Products, Image and Video editing software along with a basic usage of the Microsoft filing system. Due to the company's security policy, Alice needs to use strong passwords for her accounts, which she must provide every day at the office computer.

At home, she uses her pc to manage her email, online banking, social, dating, and Netflix accounts. As well as the sites she uses rarely.

The 4 Questions

1. Will users want to produce whatever effect the action has?
2. Will users see the control (button, menu, label, etc.) for the action?
3. Once users find the control, will they recognize that it will produce the effect they want?
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?

Main Task 1

Scenario: Alice needs an application to take notes during work. A colleague of hers suggests Evernote, but to use it she needs to first create an online Evernote account. She navigates to the Evernote official website and clicks the sign-up button. Alice has already installed Keepass2 in her personal computer and created an encrypted database by choosing a master password. She now plans to use KeePass to create and save the Evernote password.

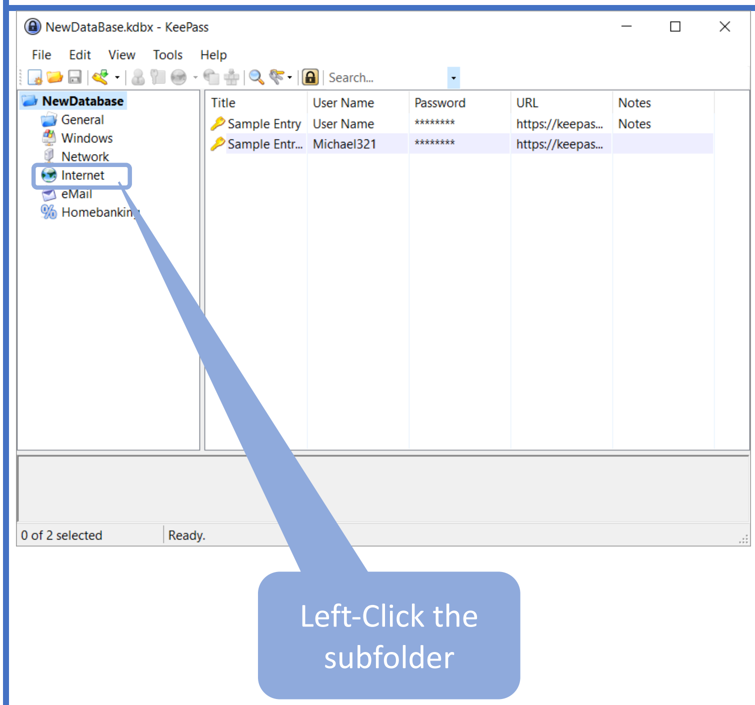
Create a new entry for your newly created Evernote account. The new entry should be created in the database file called "NewDatabase" under the "Online" subfolder.

Subtasks

Subtask 1

Navigate to the "*Internet*" subfolder of the current database, named "*NewDatabase*"

UI Screen



Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

Left-Click the subfolder

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

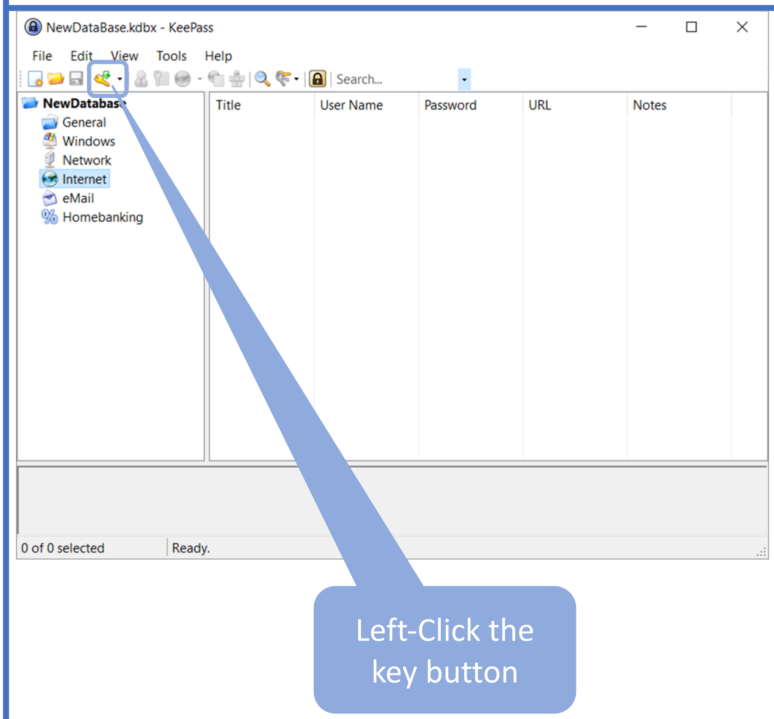
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 2

Add an entry to the internet subfolder

UI Screen



Left-Click the key button

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

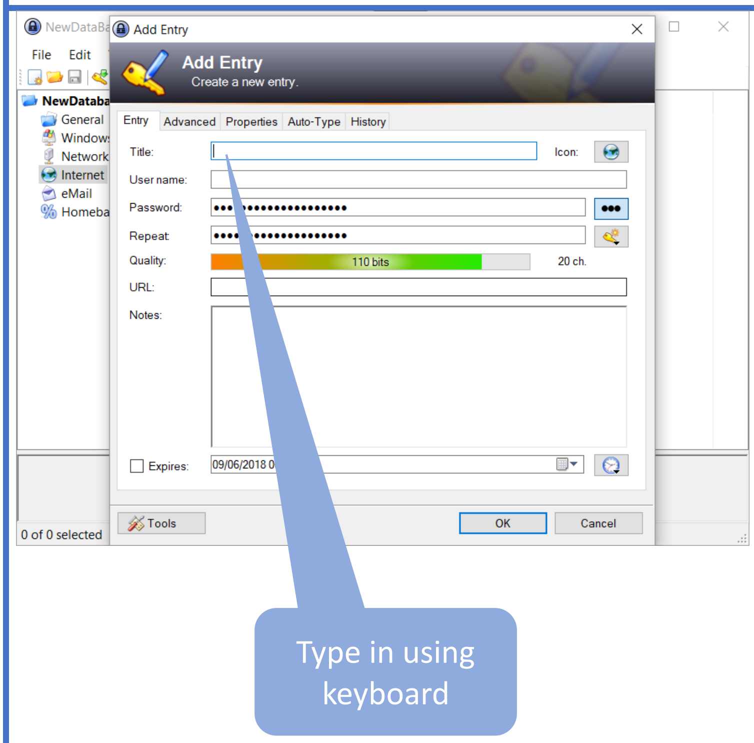
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 3

Fill in the details of the new entry

UI Screen



Type in using keyboard

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

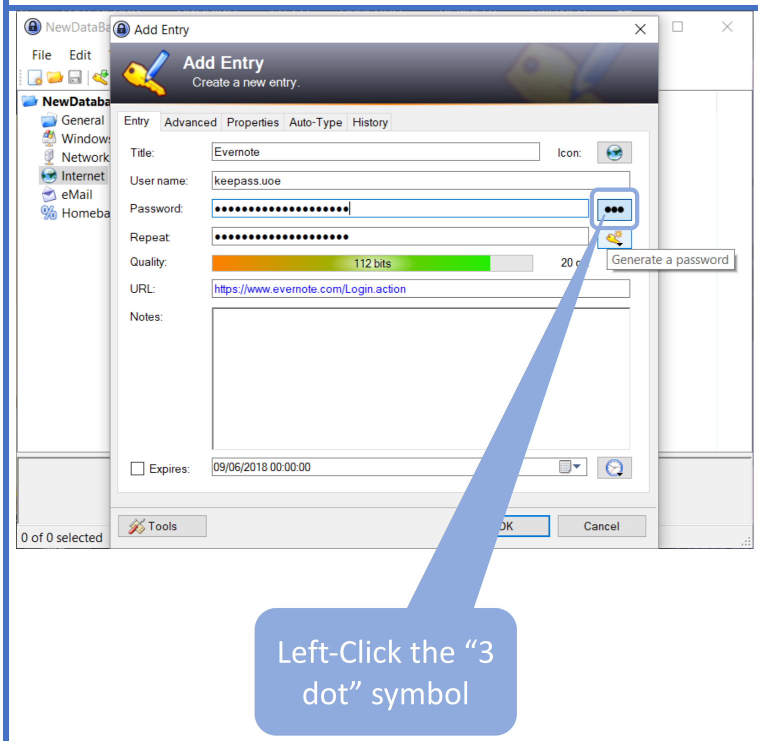
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 4

Reveal automatically created password and use it to create the Evernote online profile

UI Screen



Add Entry

Title: Evernote

User name: keepass.uoe

Password: ••••••••••••••••••••

Repeat: ••••••••••••••••••••

Quality: 112 bits

URL: https://www.evernote.com/Login.action

Notes:

□ Expires: 09/06/2018 00:00:00

Left-Click the “3 dot” symbol

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

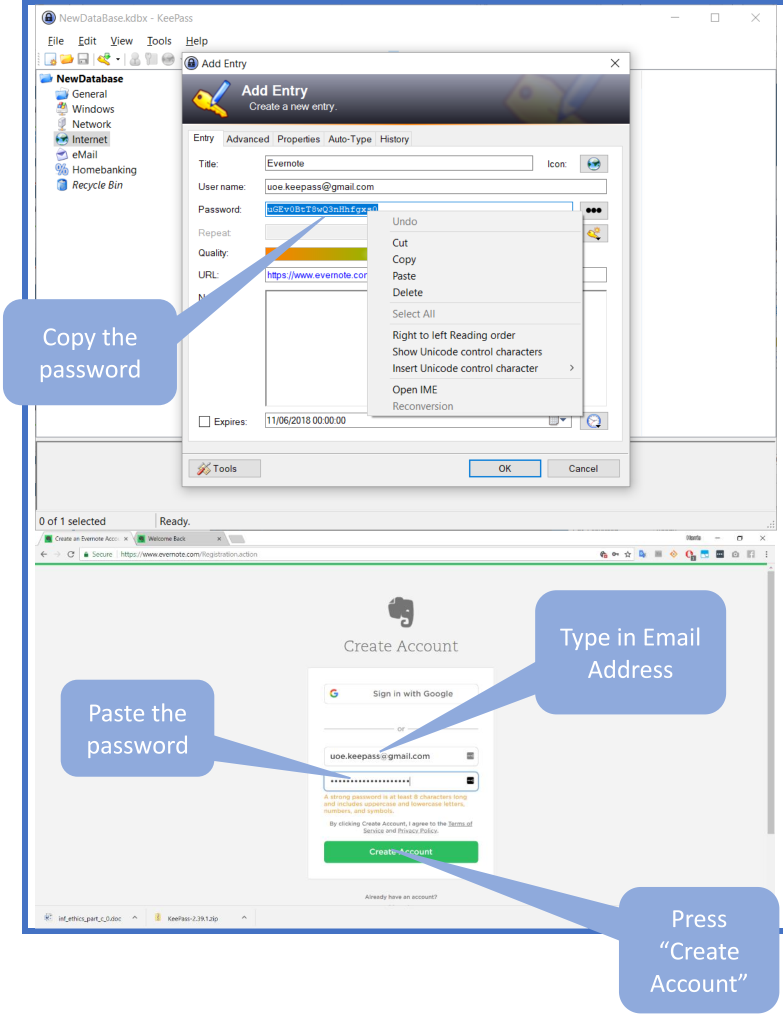
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 5

Copy and paste the revealed password to the Evernote sign up webpage and create your Evernote account

UI Screen



Title: Evernote

User name: uoe.keepass@gmail.com

Password: uGEv0BtT8wQ3nHhfgxaD

URL: https://www.evernote.com

Expires: 11/06/2018 00:00:00

https://www.evernote.com/Registration.action

uoe.keepass@gmail.com

A strong password is at least 8 characters long and includes uppercase and lowercase letters, numbers, and symbols.

By clicking Create Account, I agree to the Terms of Service and Privacy Policy.

Copy the password

Type in Email Address

Paste the password

Press "Create Account"

4 Questions

1. Will users want to produce whatever effect the action has?

Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?

Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?

Yes No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?

Yes No

____

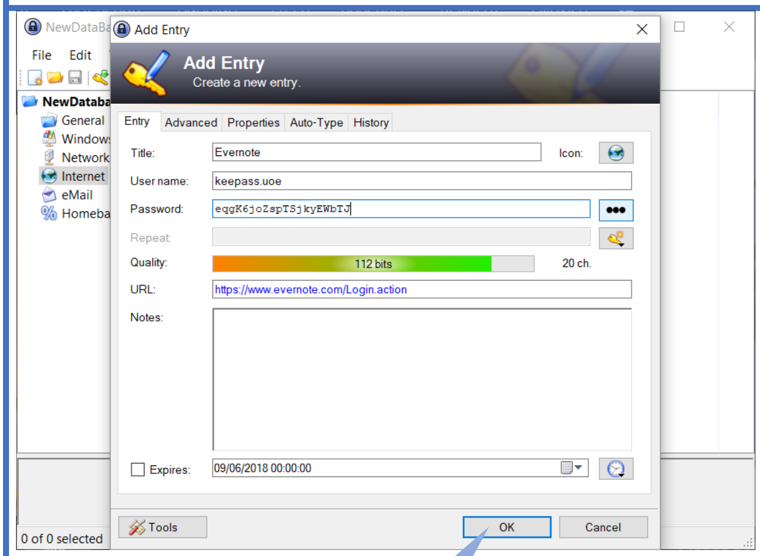
____

____

Subtask 6

Finalize the creation of the Evernote Entry

UI Screen



Add Entry

Create a new entry.

Title: Evernote

User name: keepass.uoe

Password: eqgK6joZspTSjkyEWbTJ

Quality: 112 bits 20 ch.

URL: https://www.evernote.com/Login.action

□ Expires: 09/06/2018 00:00:00

0 of 0 selected

Left-Click the OK button

4 Questions

1. Will users want to produce whatever effect the action has?

Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?

Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?

Yes No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?

Yes No

____

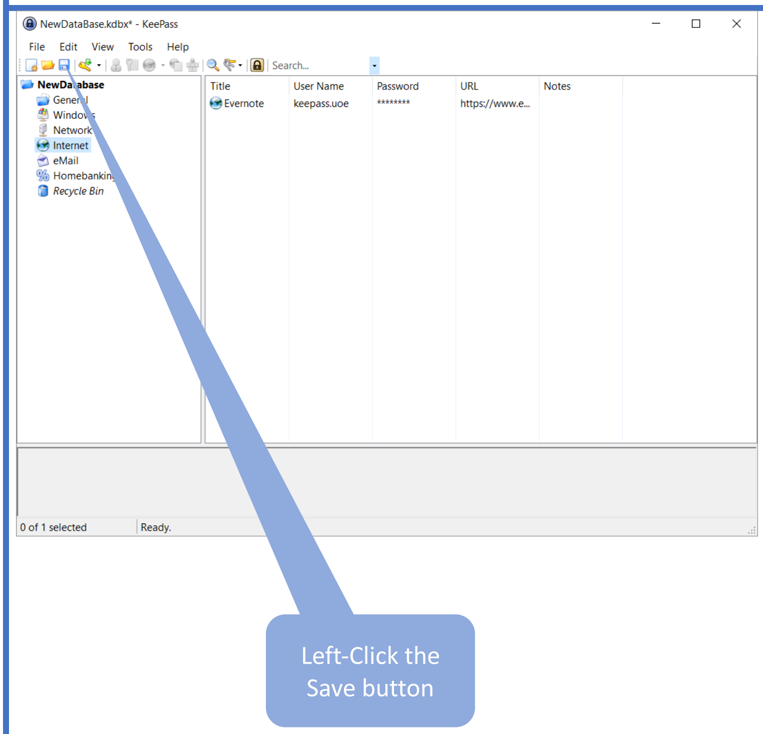
____

____

Subtask 7

Save Changes

UI Screen



Title	User Name	Password	URL	Notes
Evernote	keepass.uoe	********	https://www.e...	

0 of 1 selected

Ready.

Left-Click the Save button

4 Questions

1. Will users want to produce whatever effect the action has?

Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?

Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?

Yes No

____

____

____

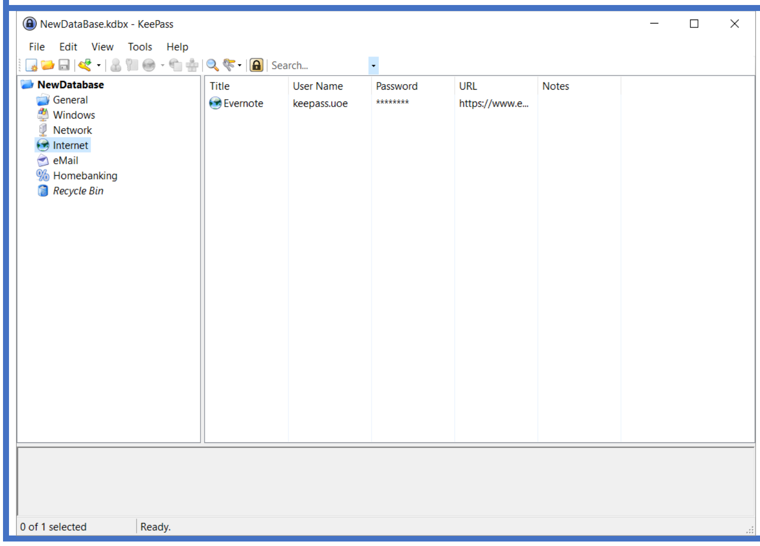
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?

Yes No

____

____

____

Final UI Screen	Comments
	

Title	User Name	Password	URL	Notes
Evernote	keepass.uoe	********	https://www.e...	

0 of 1 selected

Ready.

Main Task 2

Scenario: Alice has decided she has too many passwords that she has to remember, so she decides to start using KeePass to manage them. Her computer at work already has KeePass installed and a co-worker recommended she use it. Because she has never used it before, when she starts she has to setup a new password database before she can enter any new passwords.

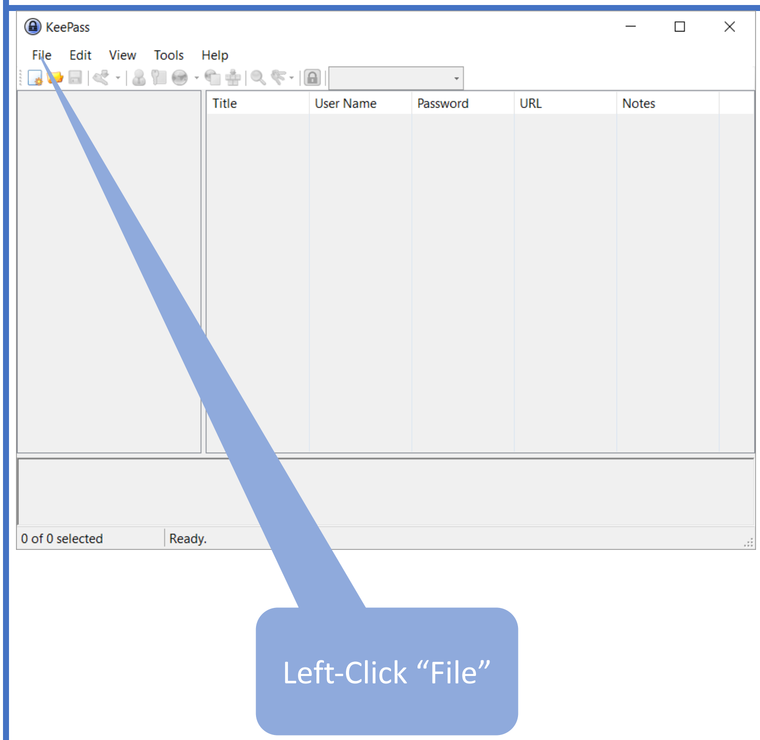
Create a new encrypted database.

Subtasks

Subtask 1

Click the “File” menu item

UI Screen



Left-Click “File”

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

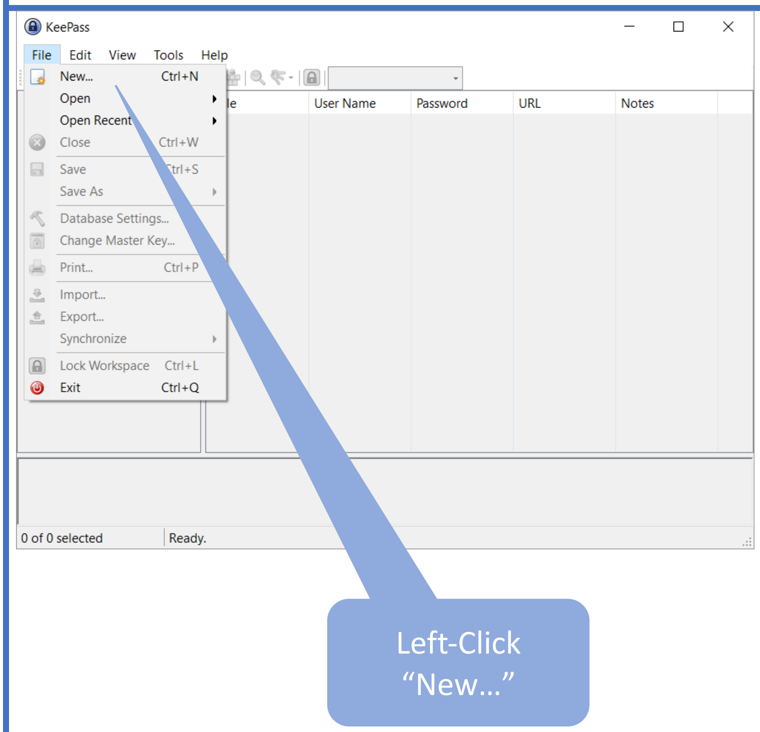
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 2

Click “New…”

UI Screen



4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____

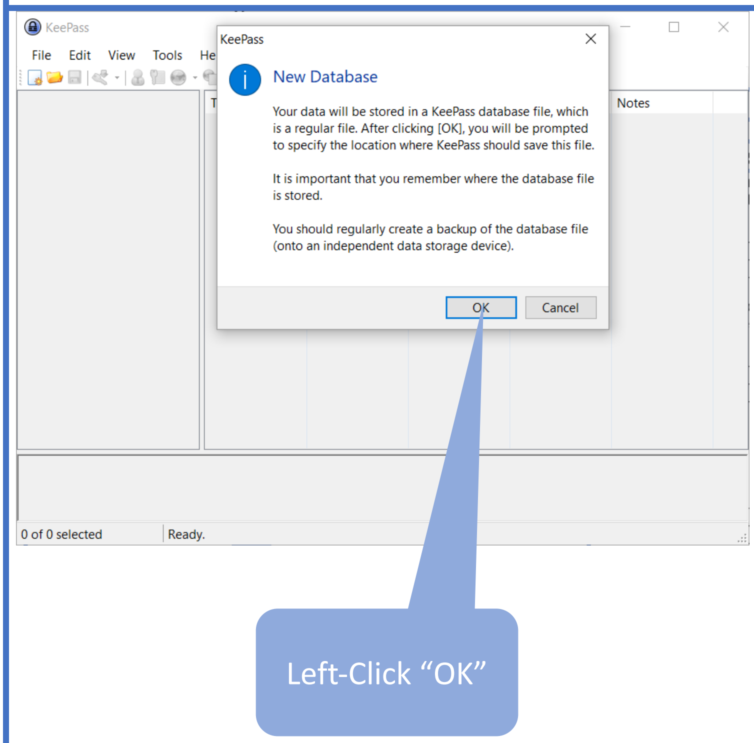
____

____

Subtask 3

Read the notification and Click “Ok”

UI Screen



New Database

Your data will be stored in a KeePass database file, which is a regular file. After clicking [OK], you will be prompted to specify the location where KeePass should save this file.

It is important that you remember where the database file is stored.

You should regularly create a backup of the database file (onto an independent data storage device).

Left-Click “OK”

4 Questions

1. Will users want to produce whatever effect the action has?
Yes   No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes   No
____
____
____

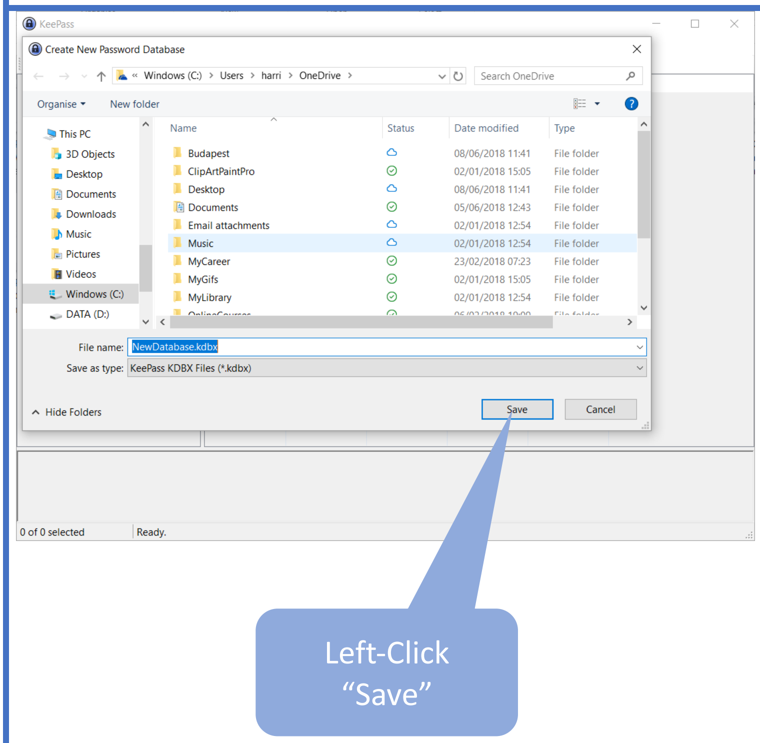
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes   No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes   No
____
____
____

Subtask 4

Choose the Directory where your new database will be saved, and press save

UI Screen



Name	Status	Date modified	Type
Budapest		08/06/2018 11:41	File folder
ClipArtPaintPro		02/01/2018 15:05	File folder
Desktop		08/06/2018 11:41	File folder
Documents		05/06/2018 12:43	File folder
Email attachments		02/01/2018 12:54	File folder
Music		02/01/2018 12:54	File folder
MyCareer		23/02/2018 07:23	File folder
MyGifs		02/01/2018 15:05	File folder
MyLibrary		02/01/2018 12:54	File folder
OnlineCourses		06/03/2018 10:00	File folder

File name: NewDatabase.kdbx

Save as type: KeePass KDBX Files (*.kdbx)

Left-Click
"Save"

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____
____
____

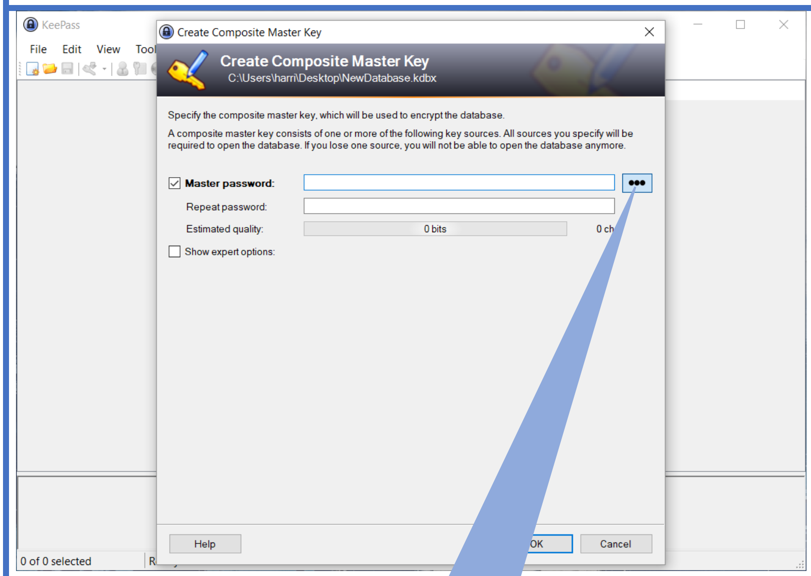
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____
____
____

Subtask 5 – Path 1

Reveal the password field to view what you are typing

UI Screen



Create Composite Master Key

C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** [ ] •••

Repeat password: [ ]

Estimated quality: 0 bits 0 ch

☐ Show expert options:

Help OK Cancel

Left-Click the “3 dot” button

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____

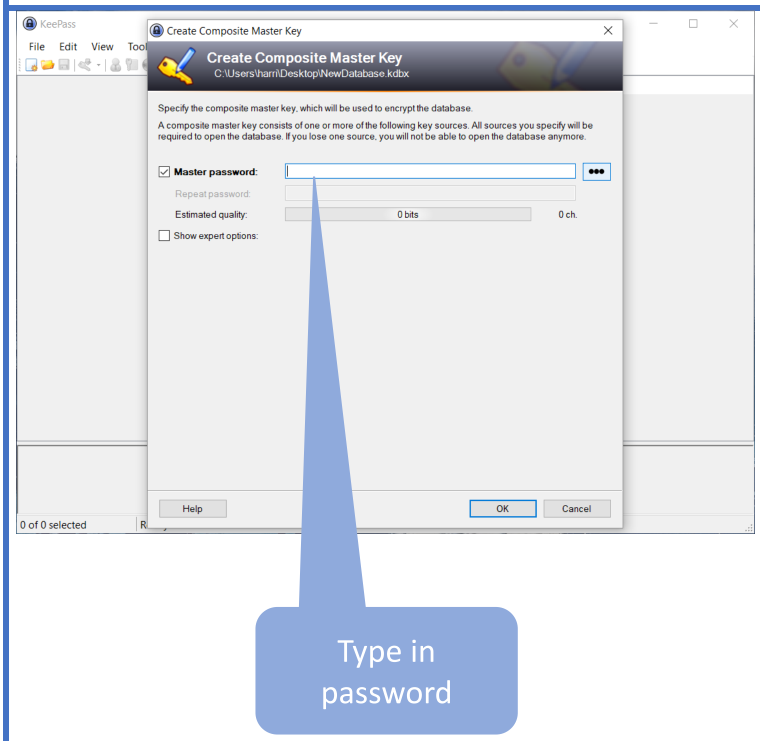
____

____

Subtask 6 – Path 1

Type in your password of choice

UI Screen



KeePass

File Edit View Tool

Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:**

Repeat password:

Estimated quality: 0 bits 0 ch.

□ Show expert options:

Help OK Cancel

0 of 0 selected

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____
____
____

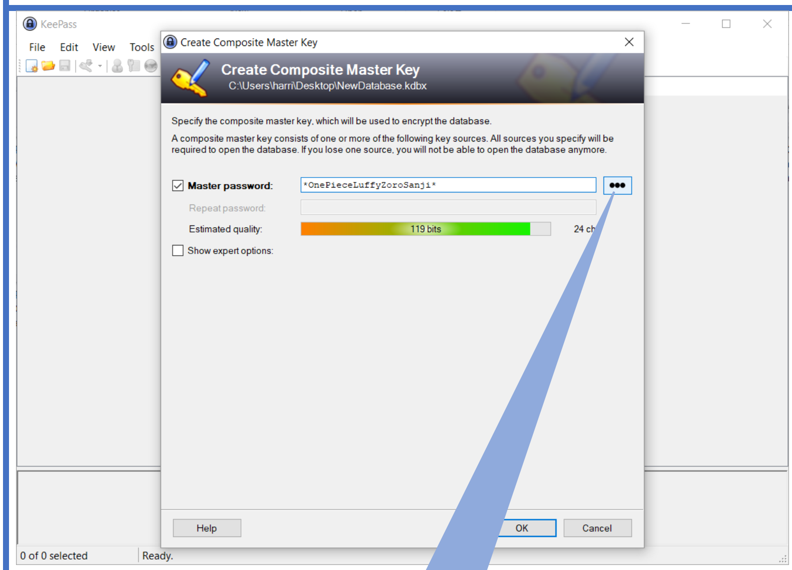
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____
____
____

Subtask 7 – Path 1

Hide the password field

UI Screen



Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** *OnePieceLuffyZoroSanji* ●●●

Repeat password:

Estimated quality: 119 bits 24 ch

□ Show expert options:

0 of 0 selected | Ready.

Left-Click the “3 dot” button

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

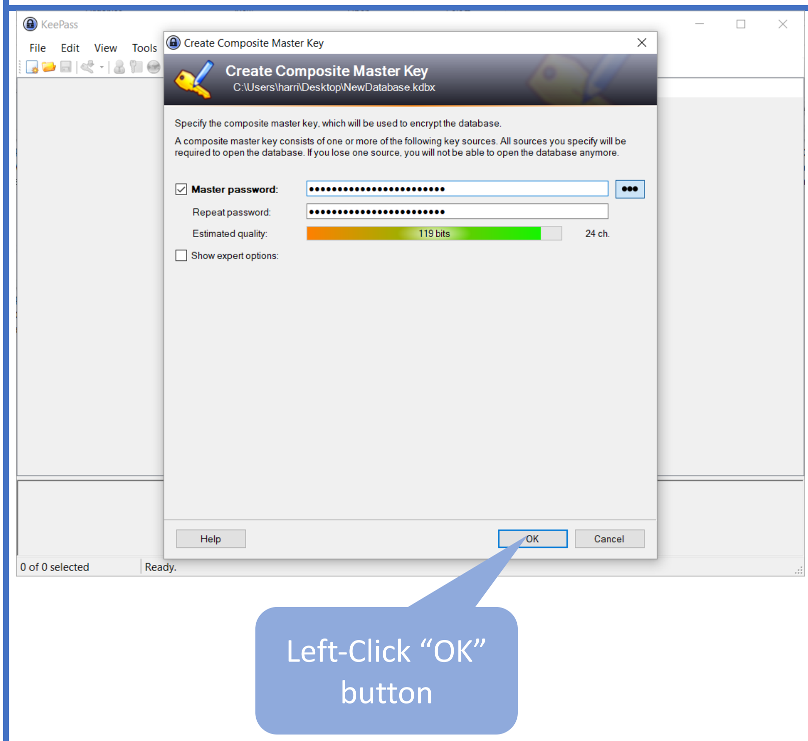
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 8 – Path 1

Accept the master password settings you have chosen

UI Screen



Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** ••••••••••••••••••••••••

Repeat password: ••••••••••••••••••••••••

Estimated quality: 119 bits 24 ch.

☐ Show expert options:

Left-Click “OK” button

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____

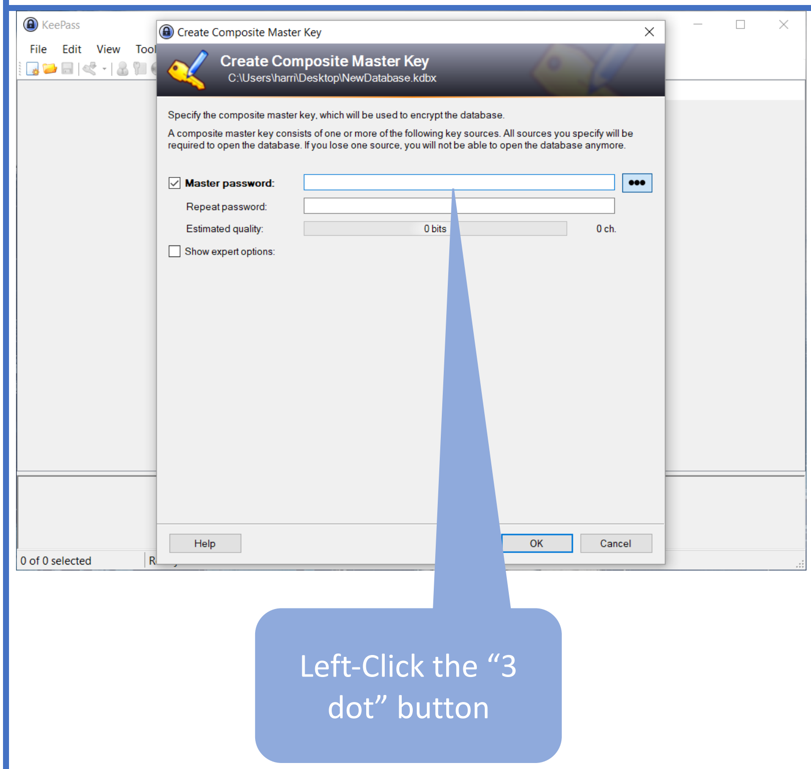
____

____

Subtask 5 – Path 2

Start to type the master password of your choice (without revealing the password field)

UI Screen



Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** ●●●

Repeat password:

Estimated quality: 0 bits 0 ch.

□ Show expert options:

Left-Click the “3 dot” button

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____
____
____

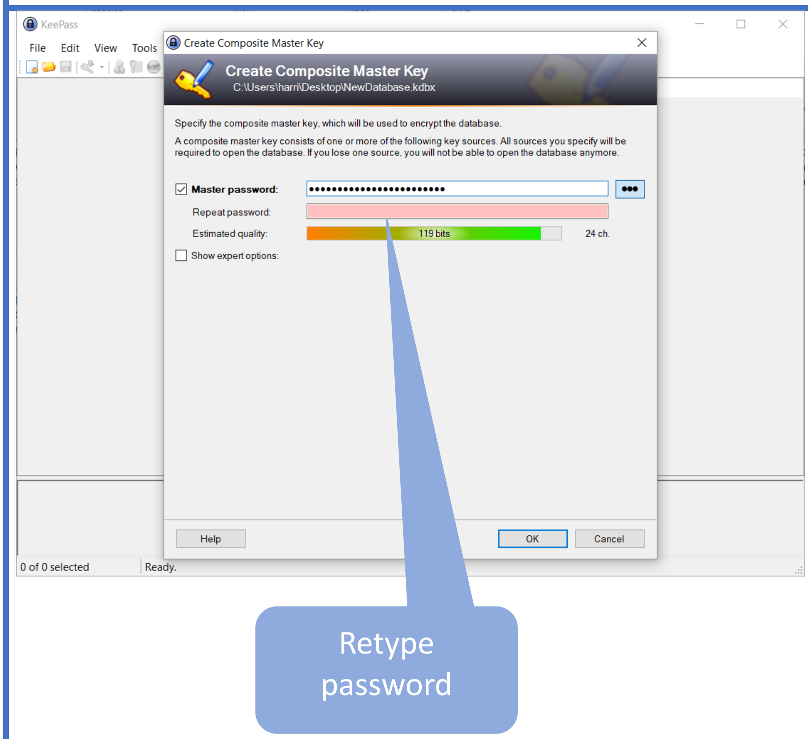
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____
____
____

Subtask 6 – Path 2

Start to retype the master password in to the Repeat password field

UI Screen



Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** ••••••••••••••••••••••••

Repeat password:

Estimated quality: 119 bits 24 ch.

□ Show expert options:

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____

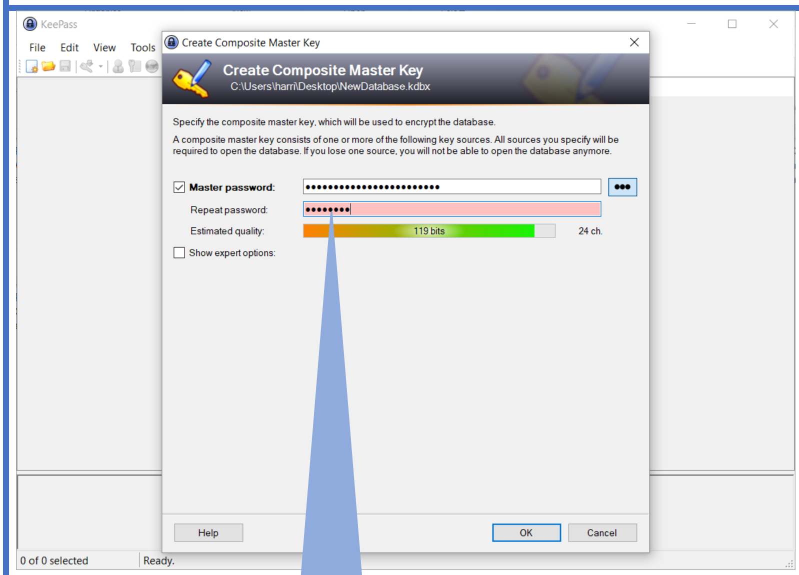
____

____

Subtask 7 – Path 2

Continue to retype password until you have entered the correct/complete password

UI Screen



Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ Master password: ••••••••••••••••••••••••

Repeat password: ••••••••

Estimated quality: 119 bits 24 ch.

□ Show expert options:

0 of 0 selected
Ready.

Continue to retype password

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____
____
____

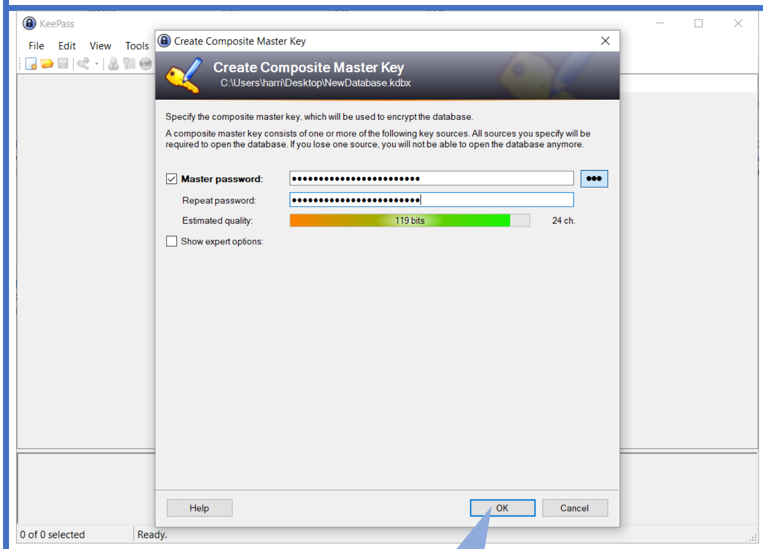
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____
____
____

Subtask 8 – Path 2

Accept the master password settings you have chosen

UI Screen



Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** ••••••••••••••••••••••••

Repeat password: ••••••••••••••••••••••••

Estimated quality: 119 bits 24 ch.

☐ Show expert options:

Left-Click “OK”

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____
____
____

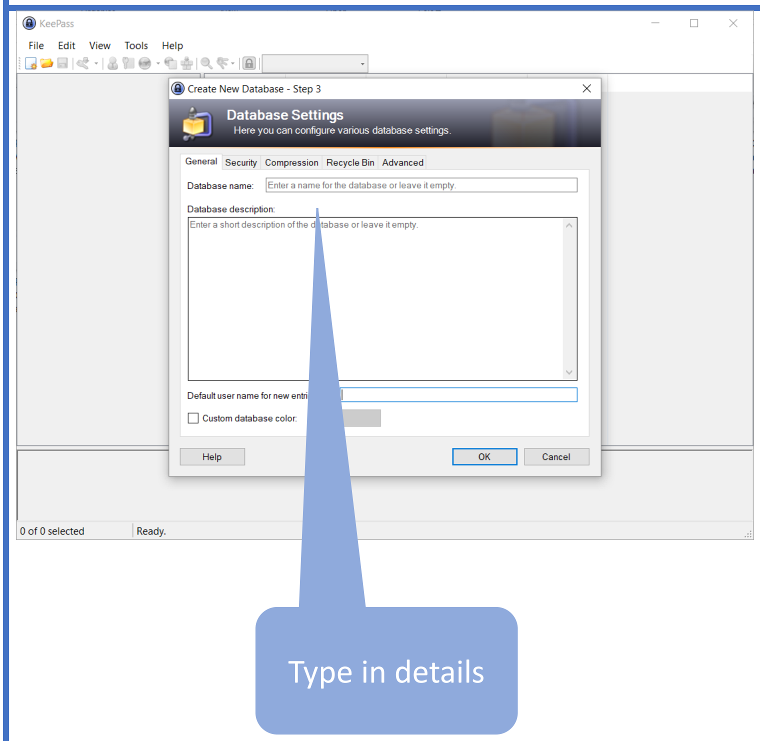
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____
____
____

Subtask 9

Fill in the database details

UI Screen


KeePass
File
Edit
View
Tools
Help
Create New Database - Step 3
Database Settings
Here you can configure various database settings.
General
Security
Compression
Recycle Bin
Advanced
Database name:
Enter a name for the database or leave it empty.
Database description:
Enter a short description of the database or leave it empty.
Default user name for new entri
Custom database color:
Help
OK
Cancel
0 of 0 selected
Ready.
Type in details

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No

____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No

____
____
____

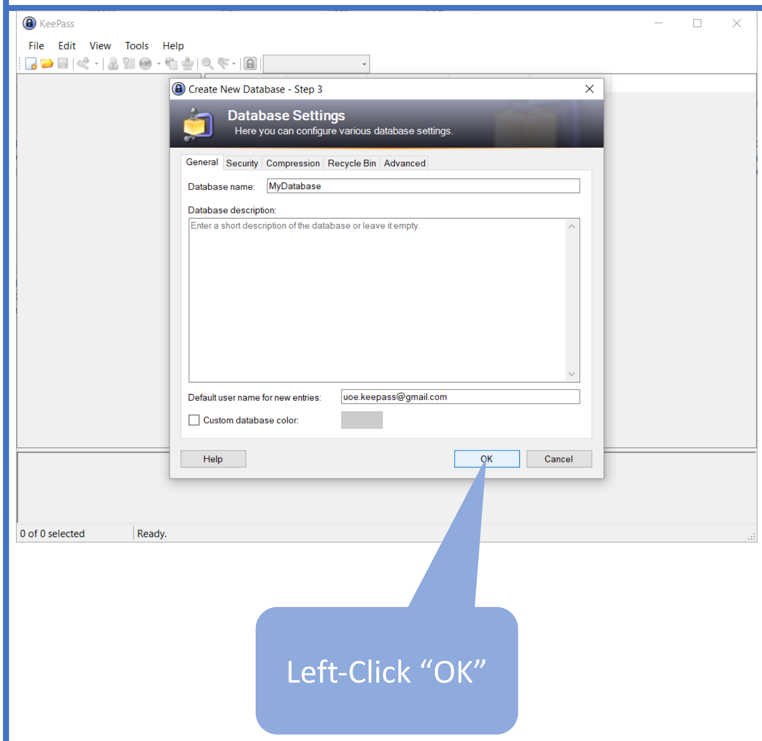
4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No

____
____
____

Subtask 10

Accept the changes Database settings

UI Screen



Database Settings

Here you can configure various database settings.

General | Security | Compression | Recycle Bin | Advanced

Database name: MyDatabase

Database description:

Enter a short description of the database or leave it empty.

Default user name for new entries: uoe.keepass@gmail.com

□ Custom database color:

0 of 0 selected | Ready.

4 Questions

1. Will users want to produce whatever effect the action has?

Yes   No

____

____

____

2. Will users see the control (button, menu, label, etc.) for the action?

Yes   No

____

____

____

3. Once users find the control, will they recognize that it will produce the effect they want?

Yes   No

____

____

____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?

Yes   No

____

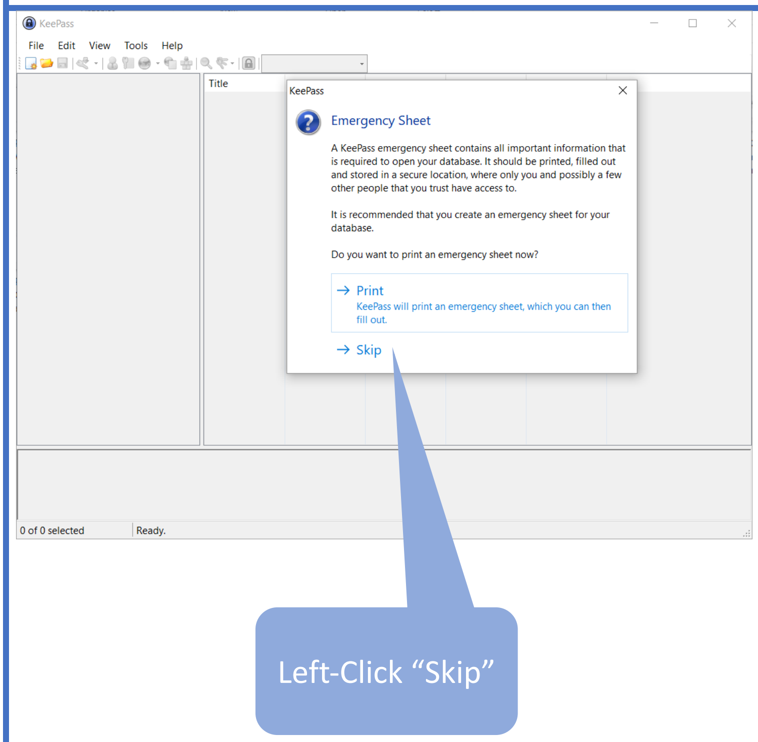
____

____

Subtask 11

Skip this step

UI Screen



KeePass

Emergency Sheet

A KeePass emergency sheet contains all important information that is required to open your database. It should be printed, filled out and stored in a secure location, where only you and possibly a few other people that you trust have access to.

It is recommended that you create an emergency sheet for your database.

Do you want to print an emergency sheet now?

→ Print
KeePass will print an emergency sheet, which you can then fill out.

→ Skip

Left-Click “Skip”

4 Questions

1. Will users want to produce whatever effect the action has?
Yes No
____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes No
____
____
____

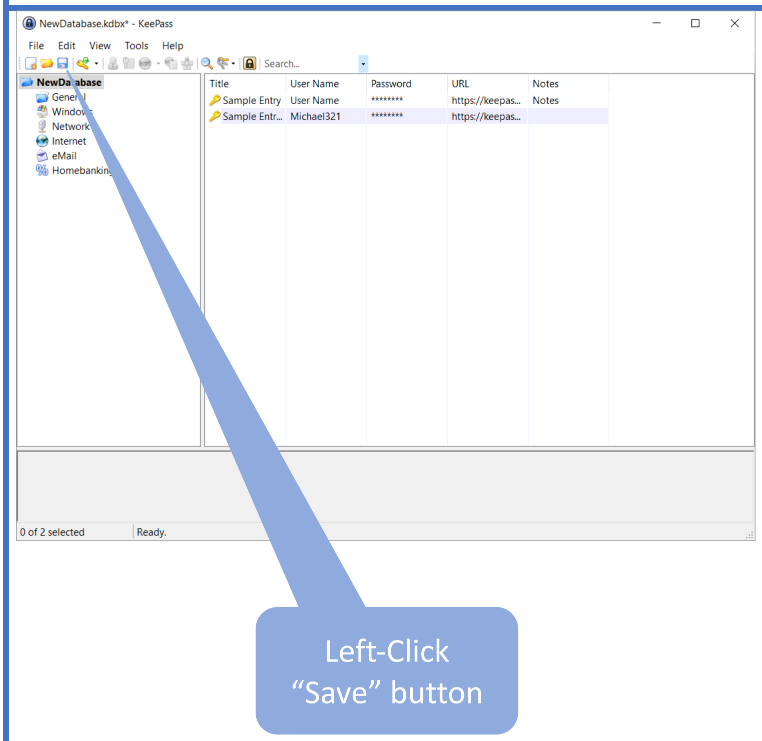
3. Once users find the control, will they recognize that it will produce the effect they want?
Yes No
____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes No
____
____
____

Subtask 12

Save the newly created database

UI Screen



Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

0 of 2 selected

Ready.

Left-Click “Save” button

4 Questions

1. Will users want to produce whatever effect the action has?
Yes   No

____
____
____

2. Will users see the control (button, menu, label, etc.) for the action?
Yes   No

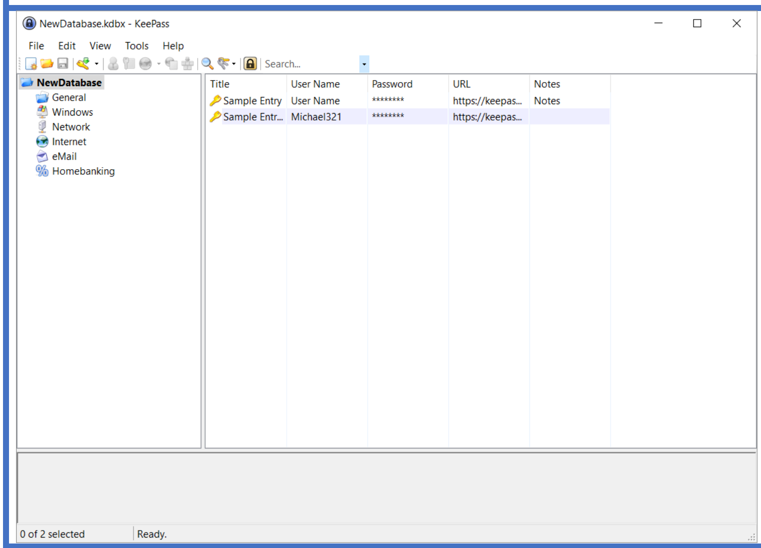
____
____
____

3. Once users find the control, will they recognize that it will produce the effect they want?
Yes   No

____
____
____

4. After the action is taken, will users understand the feedback they get, so they can confidently continue to the next action?
Yes   No

____
____
____

Final UI Screen	Comments
	

Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

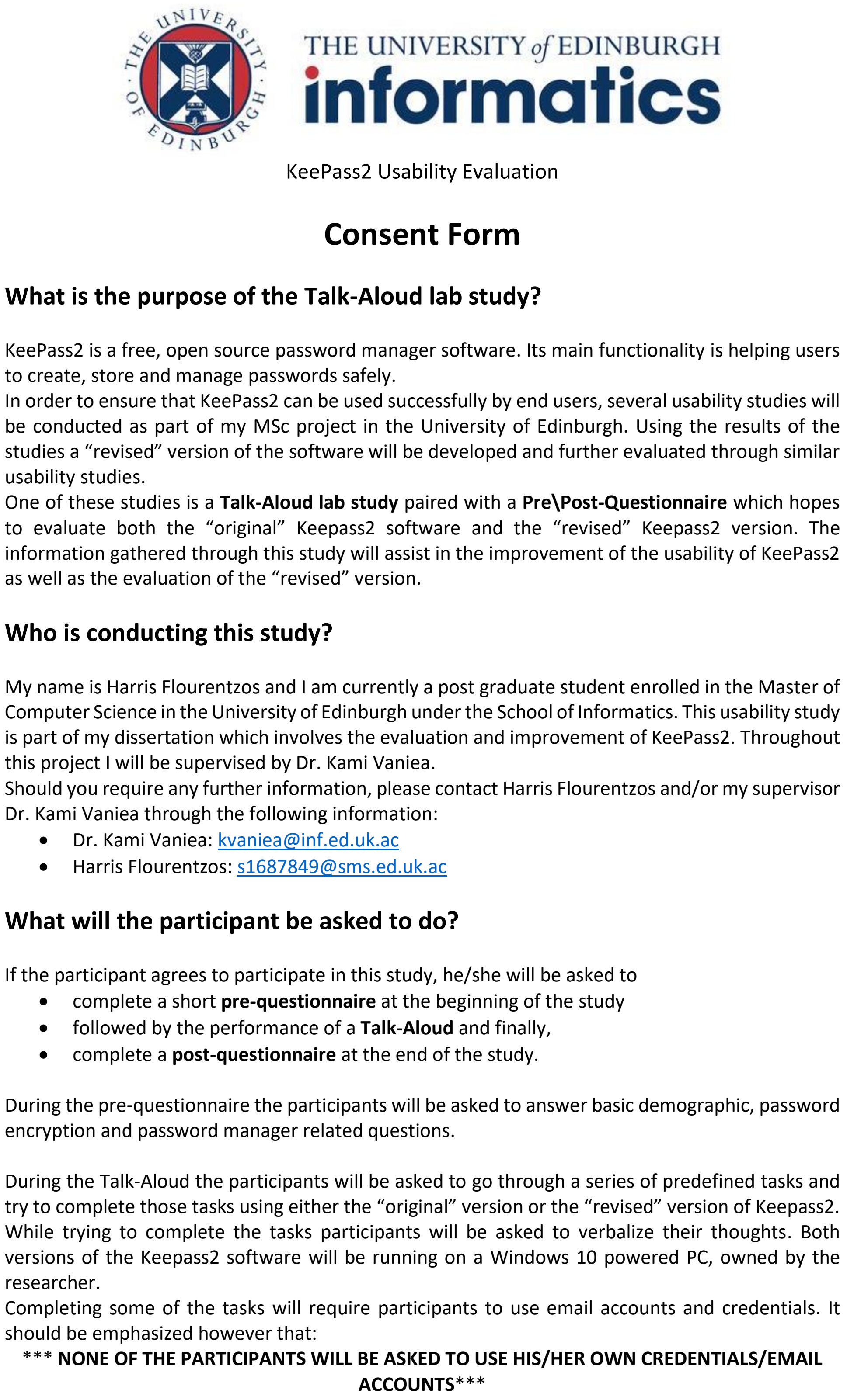
0 of 2 selected

Ready.

Appendix B

Think Aloud

B.1 Consent Form



THE UNIVERSITY *of* EDINBURGH
informatics

KeePass2 Usability Evaluation

Consent Form

What is the purpose of the Talk-Aloud lab study?

KeePass2 is a free, open source password manager software. Its main functionality is helping users to create, store and manage passwords safely.
In order to ensure that KeePass2 can be used successfully by end users, several usability studies will be conducted as part of my MSc project in the University of Edinburgh. Using the results of the studies a “revised” version of the software will be developed and further evaluated through similar usability studies.
One of these studies is a **Talk-Aloud lab study** paired with a **Pre\Post-Questionnaire** which hopes to evaluate both the “original” Keepass2 software and the “revised” Keepass2 version. The information gathered through this study will assist in the improvement of the usability of KeePass2 as well as the evaluation of the “revised” version.

Who is conducting this study?

My name is Harris Flourentzos and I am currently a post graduate student enrolled in the Master of Computer Science in the University of Edinburgh under the School of Informatics. This usability study is part of my dissertation which involves the evaluation and improvement of KeePass2. Throughout this project I will be supervised by Dr. Kami Vaniea.
Should you require any further information, please contact Harris Flourentzos and/or my supervisor Dr. Kami Vaniea through the following information:

- Dr. Kami Vaniea: kvaniea@inf.ed.uk.ac
- Harris Flourentzos: s1687849@sms.ed.uk.ac

What will the participant be asked to do?

If the participant agrees to participate in this study, he/she will be asked to

- complete a short **pre-questionnaire** at the beginning of the study
- followed by the performance of a **Talk-Aloud** and finally,
- complete a **post-questionnaire** at the end of the study.

During the pre-questionnaire the participants will be asked to answer basic demographic, password encryption and password manager related questions.

During the Talk-Aloud the participants will be asked to go through a series of predefined tasks and try to complete those tasks using either the “original” version or the “revised” version of Keepass2. While trying to complete the tasks participants will be asked to verbalize their thoughts. Both versions of the Keepass2 software will be running on a Windows 10 powered PC, owned by the researcher.
Completing some of the tasks will require participants to use email accounts and credentials. It should be emphasized however that:

***** NONE OF THE PARTICIPANTS WILL BE ASKED TO USE HIS/HER OWN CREDENTIALS/EMAIL ACCOUNTS*****

On the contrary participants will be using dummy accounts set up by the researcher before the study to ensure their privacy and security.
With the participant's agreement, the Talk-Aloud session will be **video recorded** for later analysis. The recording will include both, the upper part of his/her body (from chest to head) and the app's interface. The conditions and location of the test will ensure that each participant can perform the tasks in a comfortable environment

During the post-questionnaire the participant will be asked to answer questions about his/her impression of the software.

Please note that the participant can choose to discontinue at any time during the study and/or omit to answer any questions that make him/her uncomfortable.

Confidentiality

All information collected through the project will be treated confidentially. The questionnaire will be anonymized and will not be linked with the participant's name. The multimedia recordings will be reviewed only by the researcher. No names of individuals will be released to any other organization, nor will they be identified in any reports or publications arising from the study.

Use of results

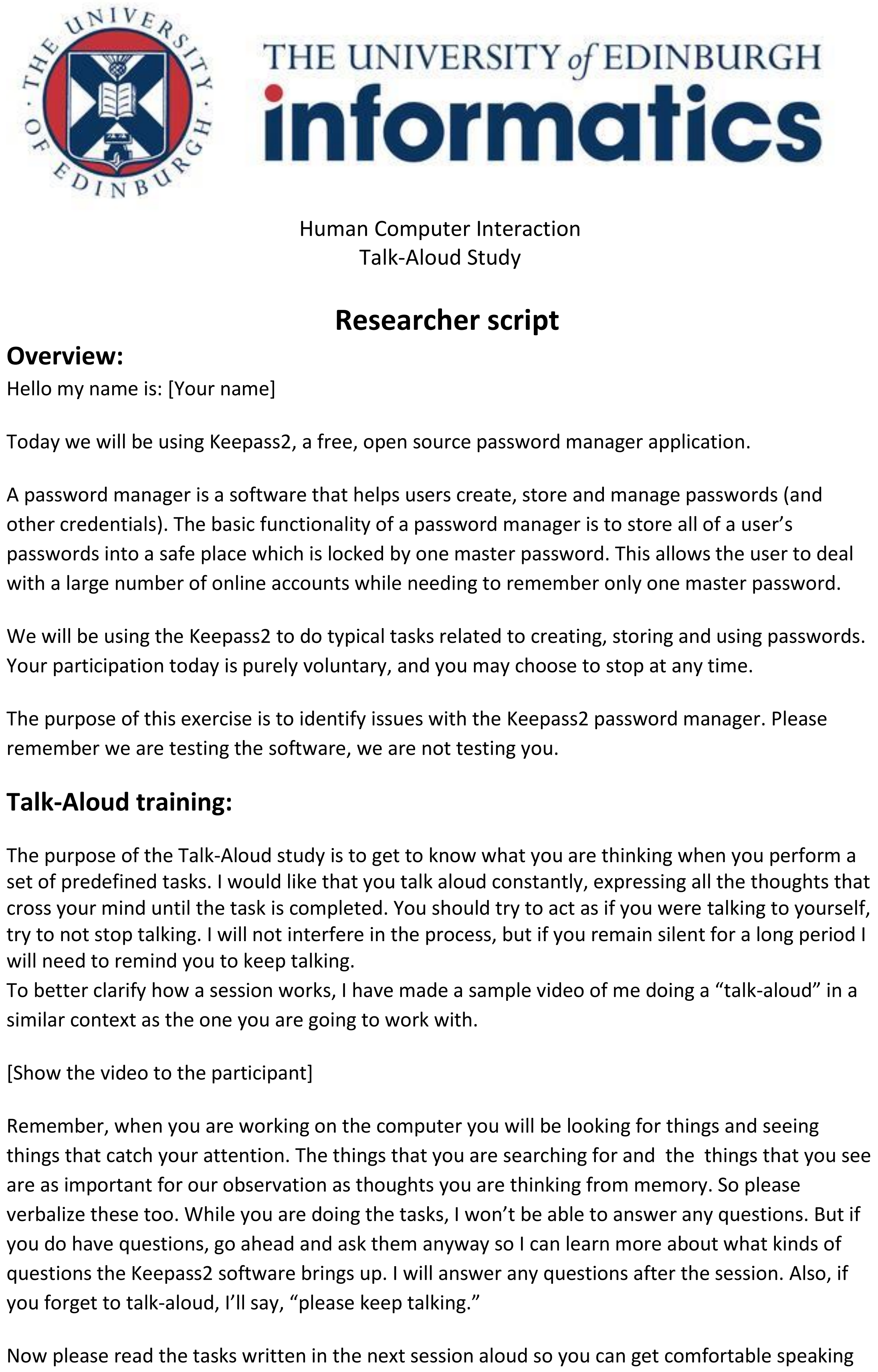
We intend to use the main findings from this research only to understand how usable and secure the KeePass2 software is. Any confidential information along with any multimedia recordings will be disposed shortly after the MSc project is concluded.

I hereby acknowledge that I have read and understand what participating in this study entails and I agree with how the information and data I provide will be treated as stated above.

____________________________ ______________________________
Participant's name Date

____
Signature

B.2 Researcher Script



THE UNIVERSITY *of* EDINBURGH
informatics

Human Computer Interaction
Talk-Aloud Study

Researcher script

Overview:

Hello my name is: [Your name]

Today we will be using Keepass2, a free, open source password manager application.

A password manager is a software that helps users create, store and manage passwords (and other credentials). The basic functionality of a password manager is to store all of a user's passwords into a safe place which is locked by one master password. This allows the user to deal with a large number of online accounts while needing to remember only one master password.

We will be using the Keepass2 to do typical tasks related to creating, storing and using passwords. Your participation today is purely voluntary, and you may choose to stop at any time.

The purpose of this exercise is to identify issues with the Keepass2 password manager. Please remember we are testing the software, we are not testing you.

Talk-Aloud training:

The purpose of the Talk-Aloud study is to get to know what you are thinking when you perform a set of predefined tasks. I would like that you talk aloud constantly, expressing all the thoughts that cross your mind until the task is completed. You should try to act as if you were talking to yourself, try to not stop talking. I will not interfere in the process, but if you remain silent for a long period I will need to remind you to keep talking.
To better clarify how a session works, I have made a sample video of me doing a "talk-aloud" in a similar context as the one you are going to work with.

[Show the video to the participant]

Remember, when you are working on the computer you will be looking for things and seeing things that catch your attention. The things that you are searching for and the things that you see are as important for our observation as thoughts you are thinking from memory. So please verbalize these too. While you are doing the tasks, I won't be able to answer any questions. But if you do have questions, go ahead and ask them anyway so I can learn more about what kinds of questions the Keepass2 software brings up. I will answer any questions after the session. Also, if you forget to talk-aloud, I'll say, "please keep talking."

Now please read the tasks written in the next session aloud so you can get comfortable speaking out loud and ask me if you have any questions about the tasks.

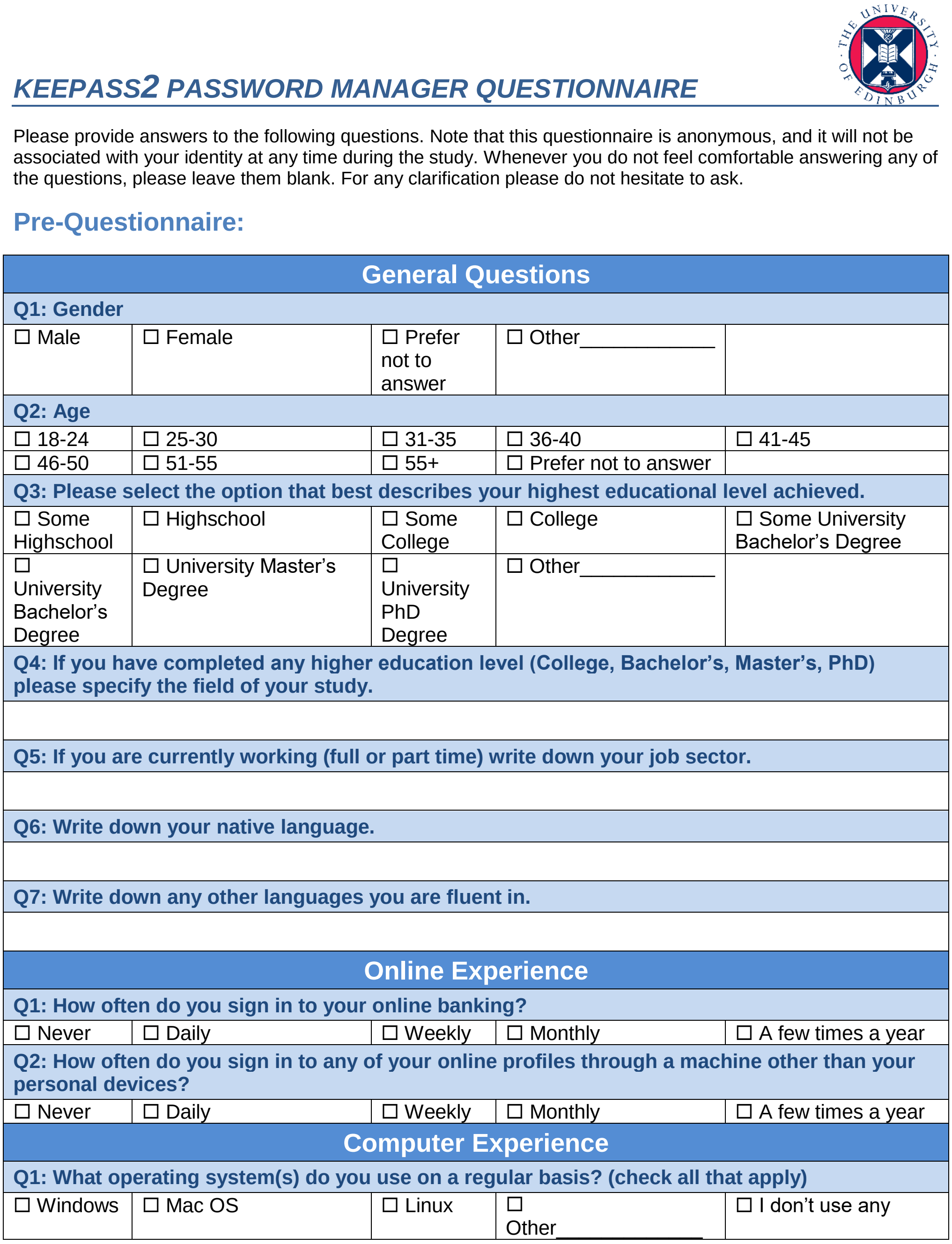
[answer any of the participants questions]

Additional Information/Guidance:

Please note that during the study, you might be asked to create passwords for some online service/account. Please *** **DO NOT USE ANY OF YOUR REAL PASSWORDS** *** at any point during this study.

You may begin.

B.3 Pre/Post Questionnaire



KEEPASS2 PASSWORD MANAGER QUESTIONNAIRE

Please provide answers to the following questions. Note that this questionnaire is anonymous, and it will not be associated with your identity at any time during the study. Whenever you do not feel comfortable answering any of the questions, please leave them blank. For any clarification please do not hesitate to ask.

Pre-Questionnaire:

General Questions				
Q1: Gender				
□ Male	□ Female	□ Prefer not to answer	□ Other____________	
Q2: Age				
□ 18-24	□ 25-30	□ 31-35	□ 36-40	□ 41-45
□ 46-50	□ 51-55	□ 55+	□ Prefer not to answer	
Q3: Please select the option that best describes your highest educational level achieved.				
□ Some Highschool	□ Highschool	□ Some College	□ College	□ Some University Bachelor's Degree
□ University Bachelor's Degree	□ University Master's Degree	□ University PhD Degree	□ Other____________	
Q4: If you have completed any higher education level (College, Bachelor's, Master's, PhD) please specify the field of your study.				
Q5: If you are currently working (full or part time) write down your job sector.				
Q6: Write down your native language.				
Q7: Write down any other languages you are fluent in.				
Online Experience				
Q1: How often do you sign in to your online banking?				
□ Never	□ Daily	□ Weekly	□ Monthly	□ A few times a year
Q2: How often do you sign in to any of your online profiles through a machine other than your personal devices?				
□ Never	□ Daily	□ Weekly	□ Monthly	□ A few times a year
Computer Experience				
Q1: What operating system(s) do you use on a regular basis? (check all that apply)				
□ Windows	□ Mac OS	□ Linux	□ Other_____________	□ I don't use any

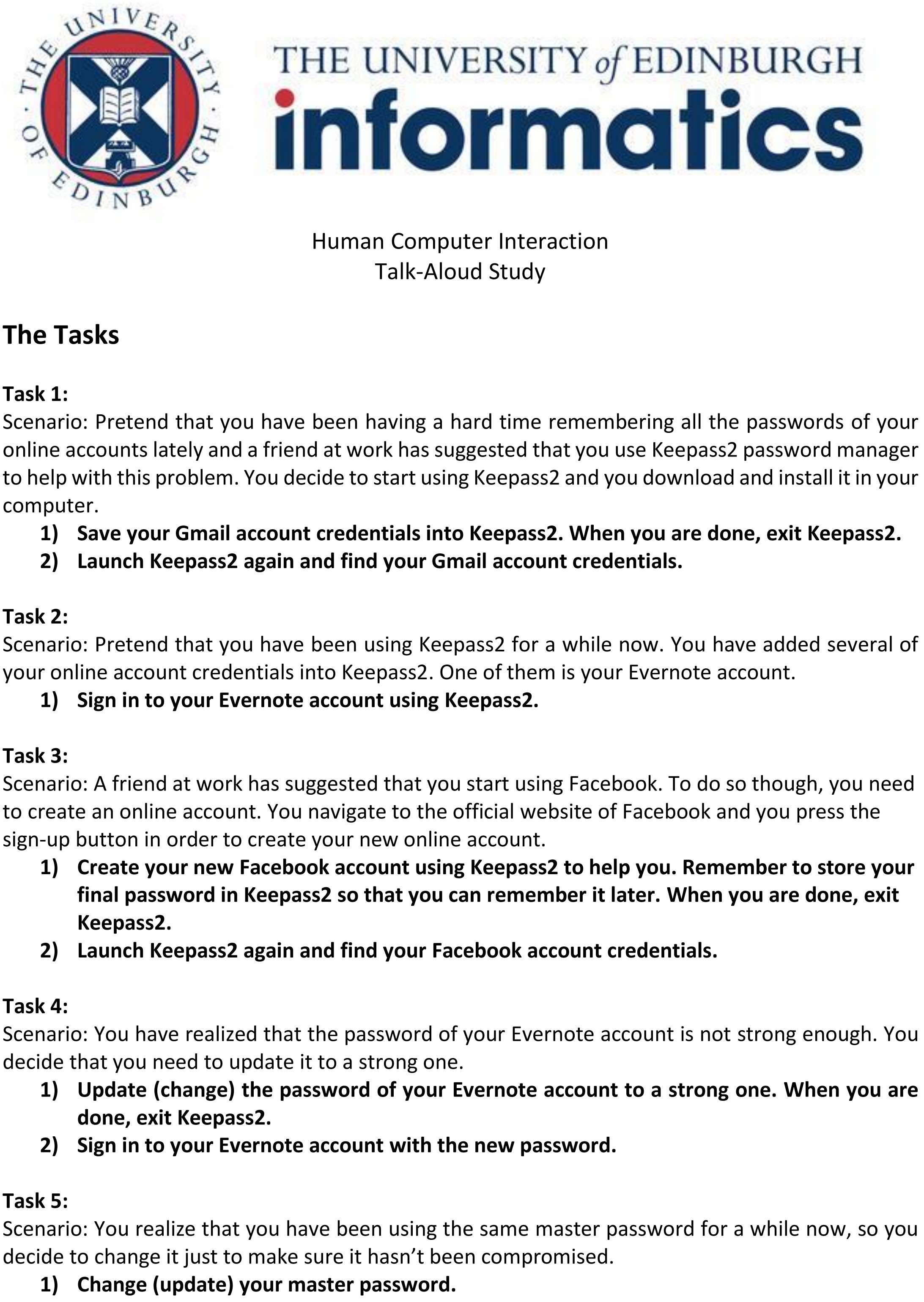
Q2: What web browser(s) do you use on a regular basis? (check all that apply)				
☐ Chrome	☐ Internet Explorer/Edge	☐ Firefox	☐ Opera	☐ I don’t use a web browser
☐ Safari	☐ Other______________			
Password Experience				
Q1: How many accounts do you have that require passwords?				
☐ 0-10	☐ 11-20	☐ 21-30	☐ 31-40	☐ 41+
Q2: How many different (unique) passwords do you have?				
☐ 1-3	☐ 4-6	☐ 7-9	☐ 10+	☐ Don’t use any passwords
Q3: How often do you change your passwords(s)?				
☐ Weekly	☐ Monthly	☐ A few times a year	☐ Only when asked by service	☐ Never
Q4: How do you remember your passwords? (check all that apply)				
☐ memory	☐ write them down in digital format (e.g. notepad)	☐ write them down in physical form (e.g. paper)	☐ Use a dedicated service (e.g. password manager)	☐ Other_____________
Q5: Does the method you specified above feel secure?				
☐ Yes	☐ No	☐ I don’t know		
Q6: How easy/convenient is the method you specified above to use?				
☐ Hard	☐ Medium	☐ Easy		
Password Manager Experience				
Q1: What password manager application(s) are you/have you been using?				
☐ LastPass	☐ 1Password	☐ Keepass	☐ Other___________	☐ I don’t use any

Post-Questionnaire:

Statement	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Experience using Keepass2					
I think that I would like to use this system frequently					
I found the system unnecessarily complex					
I thought the system was easy to use					
I think that I would need the support of a technical person to be able to use this system					
I found the various functions in this system were well integrated					
I thought there was too much inconsistency in this system					
I would imagine that most people would learn to use this system very quickly					
I found the system very cumbersome to use					
I felt very confident using the system					
I needed to learn a lot of things before I could get going with this system					

Mental Model				
Q1: Where do you think Keepass2 saved the encrypted Safe file?				
□ Online	□ On the computer	□ On an external storage device	□ Other___________	□ I don't know
Q2: What will happen if you lost your master password?				
□ I will contact the Keepass2 company to help me change it	□ I will not be able to open the Safe again and so lose all access to my credentials stored in the Safe	□ I will choose I forgot my master password	□ Other___________	□ I don't know
Q3: What will happen if a hacker steals your encrypted Safe file?				
□ My credentials are safe because he doesn't know the master password	□ The hacker will be able to unlock the Safe even without knowing the master password	□ Other___________	□ I don't know	
Q4: How strong were the pre-chosen passwords (e.g. keepass_p*) used in the study?				
□ Low Strength	□ Medium Strength	□ High Strength	□ I don't remember	□ I didn't check
□ I didn't know how to check				

B.4 Tasks



Human Computer Interaction
Talk-Aloud Study

The Tasks

Task 1:
Scenario: Pretend that you have been having a hard time remembering all the passwords of your online accounts lately and a friend at work has suggested that you use Keepass2 password manager to help with this problem. You decide to start using Keepass2 and you download and install it in your computer.

1) **Save your Gmail account credentials into Keepass2. When you are done, exit Keepass2.**
2) **Launch Keepass2 again and find your Gmail account credentials.**

Task 2:
Scenario: Pretend that you have been using Keepass2 for a while now. You have added several of your online account credentials into Keepass2. One of them is your Evernote account.

1) **Sign in to your Evernote account using Keepass2.**

Task 3:
Scenario: A friend at work has suggested that you start using Facebook. To do so though, you need to create an online account. You navigate to the official website of Facebook and you press the sign-up button in order to create your new online account.

1) **Create your new Facebook account using Keepass2 to help you. Remember to store your final password in Keepass2 so that you can remember it later. When you are done, exit Keepass2.**
2) **Launch Keepass2 again and find your Facebook account credentials.**

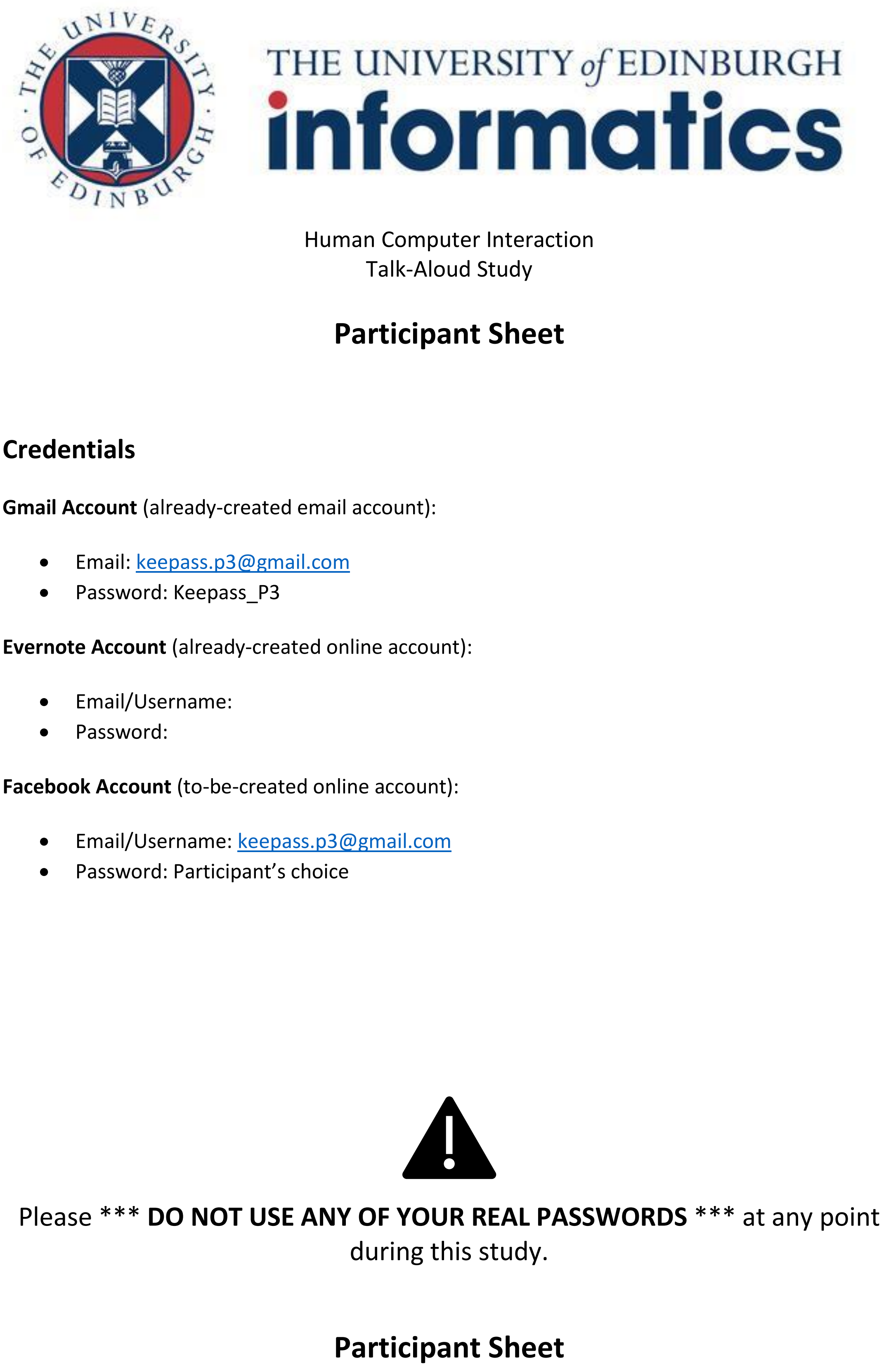
Task 4:
Scenario: You have realized that the password of your Evernote account is not strong enough. You decide that you need to update it to a strong one.

1) **Update (change) the password of your Evernote account to a strong one. When you are done, exit Keepass2.**
2) **Sign in to your Evernote account with the new password.**

Task 5:
Scenario: You realize that you have been using the same master password for a while now, so you decide to change it just to make sure it hasn't been compromised.

1) **Change (update) your master password.**

B.5 Participant’s Credential Sheet



THE UNIVERSITY *of* EDINBURGH
informatics

Human Computer Interaction
Talk-Aloud Study

Participant Sheet

Credentials

Gmail Account (already-created email account):

- Email: keepass.p3@gmail.com
- Password: Keepass_P3

Evernote Account (already-created online account):

- Email/Username:
- Password:

Facebook Account (to-be-created online account):

- Email/Username: keepass.p3@gmail.com
- Password: Participant’s choice

⚠

Please *** **DO NOT USE ANY OF YOUR REAL PASSWORDS** *** at any point during this study.

Participant Sheet

THE UNIVERSITY *of* EDINBURGH
informatics

Human Computer Interaction
Talk-Aloud Study

Credentials

Existing Database

- Database file name: p3.kdbx
- Master Password: keepassP3

New Database

- Mater Password:

Please *** **DO NOT USE ANY OF YOUR REAL PASSWORDS** *** at any point during this study.

Appendix C

Usable Security

C.1 Extra difficulties of Usable Security

Extra difficulties of Usable Security Identified in [Whitten and Tygar, 1999]:

1. **The unmotivated user property:** Security is usually a secondary goal. People do not generally sit down at their computers wanting to manage their security; rather, they want to send email, browse web pages, or download software, and they want security in place to protect them while they do those things. It is easy for people to put off learning about security, or to optimistically assume that their security is working, while they focus on their primary goals. Designers of user interfaces for security should not assume that users will be motivated to read manuals or to go looking for security controls that are designed to be unobtrusive. Furthermore, if security is too difficult or annoying, users may give up on it altogether.

2. **The abstraction property:** Computer security management often involves security policies, which are systems of abstract rules for deciding whether to grant accesses to resources. The creation and management of such rules is an activity that programmers take for granted, but which may be alien and unintuitive to many members of the wider user population. User interface design for security will need to take this into account.

3. **The lack of feedback property:** The need to prevent dangerous errors makes it imperative to provide good feedback to the user, but providing good feedback for security management is a difficult problem. The state of a security configuration is usually complex and attempts to summarize it are not adequate. Furthermore, the correct security configuration is the one which does what the user really wants, and since only the user knows what that is, it is hard for security software to perform much useful error checking.

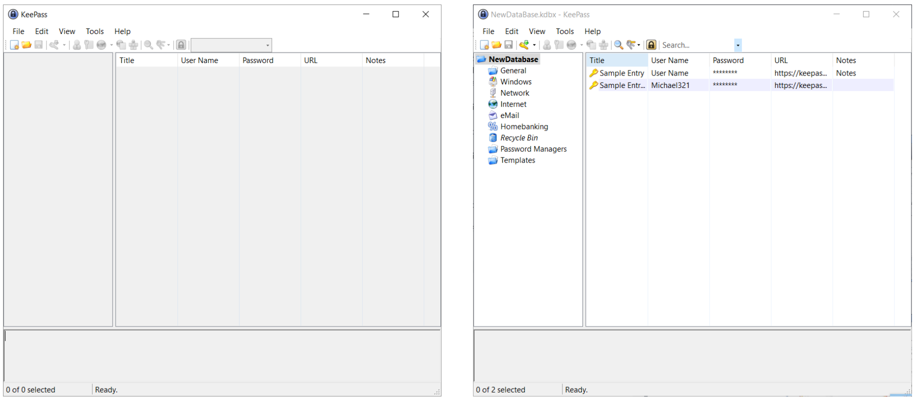
4. **The barn door property:** The proverb about the futility of locking the barn door after the horse is gone is descriptive of an important property of computer security: once a secret has been left accidentally unprotected, even for a short time, there is no way to be sure that it has not already been read by an attacker. Because of this, user interface design for security needs to place a very high priority on making sure users understand their security well enough to keep from making potentially high-cost mistakes.

5. **The weakest link property:** It is well known that the security of a networked computer is only as strong as its weakest component. If a cracker can exploit a single error, the game is up. This means that users need to be guided to attend to all aspects of their security, not left to proceed through random exploration as they might with a word processor or a spreadsheet.

Appendix D

KeePass2

D.1 UI Original KeePass2

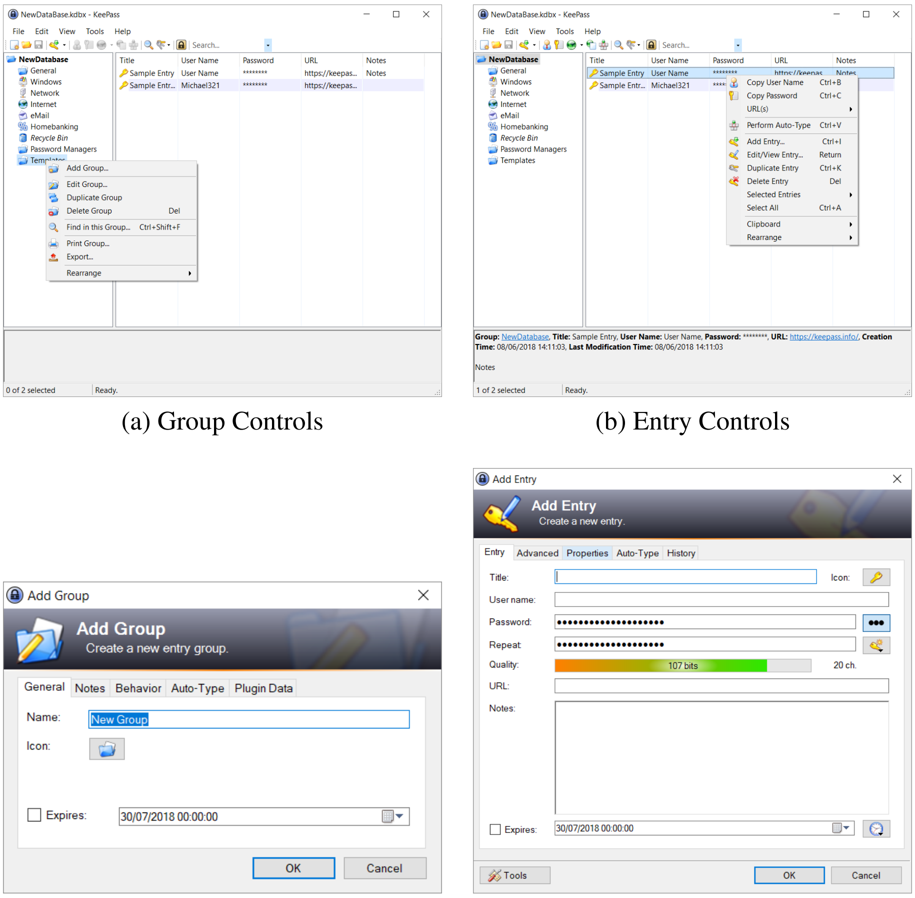


Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

(a) No opened database

(b) Opened Database

Figure D.1: Main UI screen with and without a database opened



Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

0 of 2 selected | Ready.

(a) Group Controls

Group: NewDatabase, **Title:** Sample Entry, **User Name:** User Name, **Password:** ********, **URL:** https://keepass.info/, **Creation Time:** 08/06/2018 14:11:03, **Last Modification Time:** 08/06/2018 14:11:03

Notes

1 of 2 selected | Ready.

(b) Entry Controls

Name: New Group

Icon:

□ Expires: 30/07/2018 00:00:00

(c) Add Group

Title:

Icon:

User name:

Password: ••••••••••••••••••••

Repeat: ••••••••••••••••••••

Quality: 107 bits 20 ch.

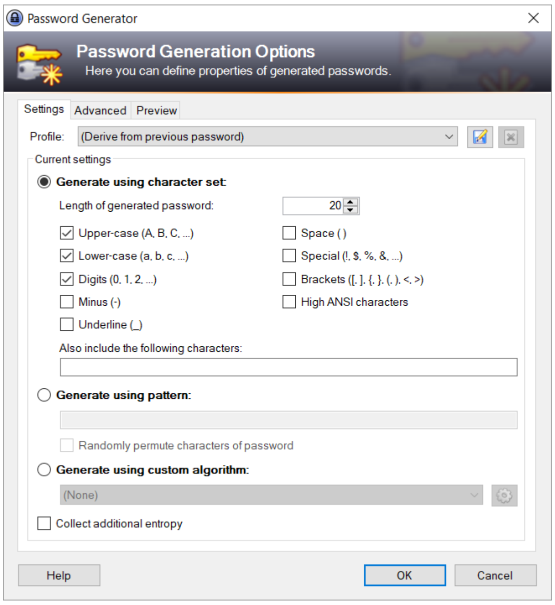
URL:

Notes:

□ Expires: 30/07/2018 00:00:00

(d) Add Entry

Figure D.2: Group and Entry controls - Add group and entry UI screens


Password Generator
Password Generation Options
Here you can define properties of generated passwords.
Settings
Advanced
Preview
Help
OK
Cancel

Profile: (Derive from previous password)

Current settings

● Generate using character set:

Length of generated password: 20

☑ Upper-case (A, B, C, ...)
☑ Lower-case (a, b, c, ...)
☑ Digits (0, 1, 2, ...)
□ Minus (-)
□ Underline (_)
□ Space ( )
□ Special (!, $, %, &, ...)
□ Brackets ([,], {, }, (,), <, >)
□ High ANSI characters

Also include the following characters:

○ Generate using pattern:

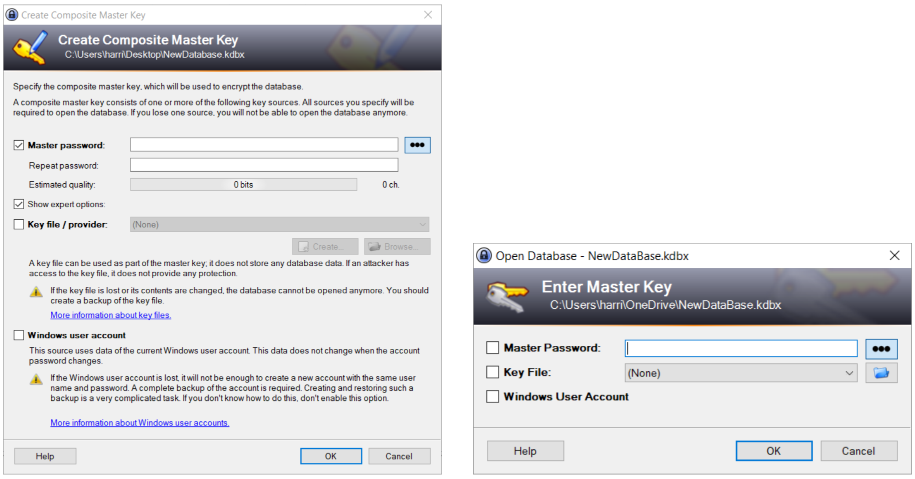
□ Randomly permute characters of password

○ Generate using custom algorithm:

(None)

□ Collect additional entropy

Figure D.3: Generate password UI screen



Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:**

Repeat password:

Estimated quality: 0 bits 0 ch.

☑ Show expert options:

□ **Key file / provider:** (None)

A key file can be used as part of the master key; it does not store any database data. If an attacker has access to the key file, it does not provide any protection.

⚠ If the key file is lost or its contents are changed, the database cannot be opened anymore. You should create a backup of the key file.

More information about key files.

□ **Windows user account**

This source uses data of the current Windows user account. This data does not change when the account password changes.

⚠ If the Windows user account is lost, it will not be enough to create a new account with the same user name and password. A complete backup of the account is required. Creating and restoring such a backup is a very complicated task. If you don't know how to do this, don't enable this option.

More information about Windows user accounts.

Open Database - NewDataBase.kdbx

Enter Master Key
C:\Users\harri\OneDrive\NewDataBase.kdbx

□ **Master Password:**

□ **Key File:** (None)

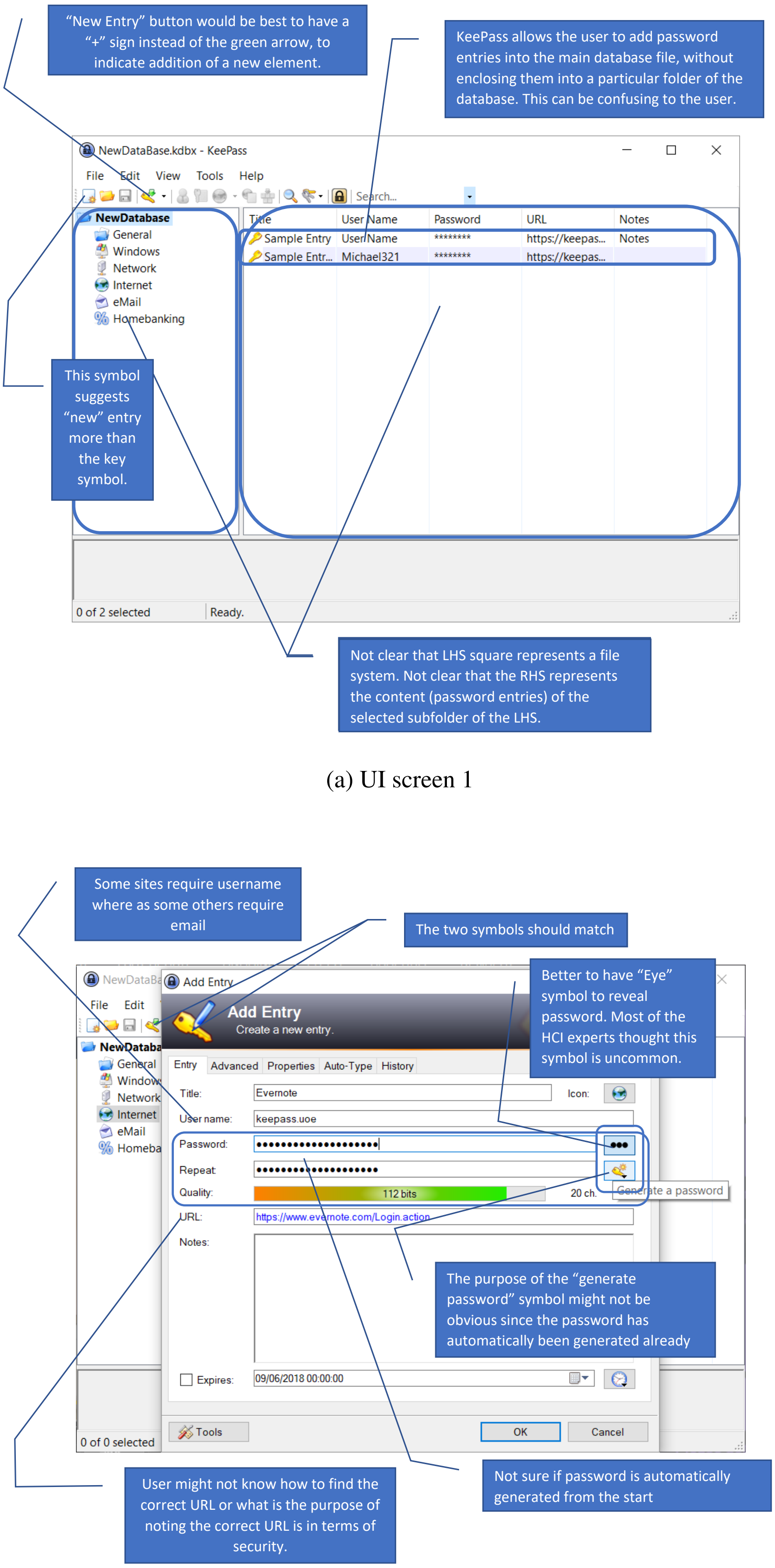
□ **Windows User Account**

(a) Create Master Password (b) Use Master Password

Figure D.4: Create and Use Master Password

Appendix E

Cognitive Walkthrough Depicted Results



"New Entry" button would be best to have a "+" sign instead of the green arrow, to indicate addition of a new element.

KeePass allows the user to add password entries into the main database file, without enclosing them into a particular folder of the database. This can be confusing to the user.

This symbol suggests "new" entry more than the key symbol.

Not clear that LHS square represents a file system. Not clear that the RHS represents the content (password entries) of the selected subfolder of the LHS.

NewDataBase.kdbx - KeePass

Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

0 of 2 selected Ready.

(a) UI screen 1

Some sites require username where as some others require email

The two symbols should match

Better to have "Eye" symbol to reveal password. Most of the HCI experts thought this symbol is uncommon.

The purpose of the "generate password" symbol might not be obvious since the password has automatically been generated already

User might not know how to find the correct URL or what is the purpose of noting the correct URL in terms of security.

Not sure if password is automatically generated from the start

Add Entry

Create a new entry.

Title: Evernote

User name: keepass.uoe

Password: ••••••••••••••••••••

Repeat: ••••••••••••••••••••

Quality: 112 bits 20 ch.

URL: https://www.evernote.com/Login.action

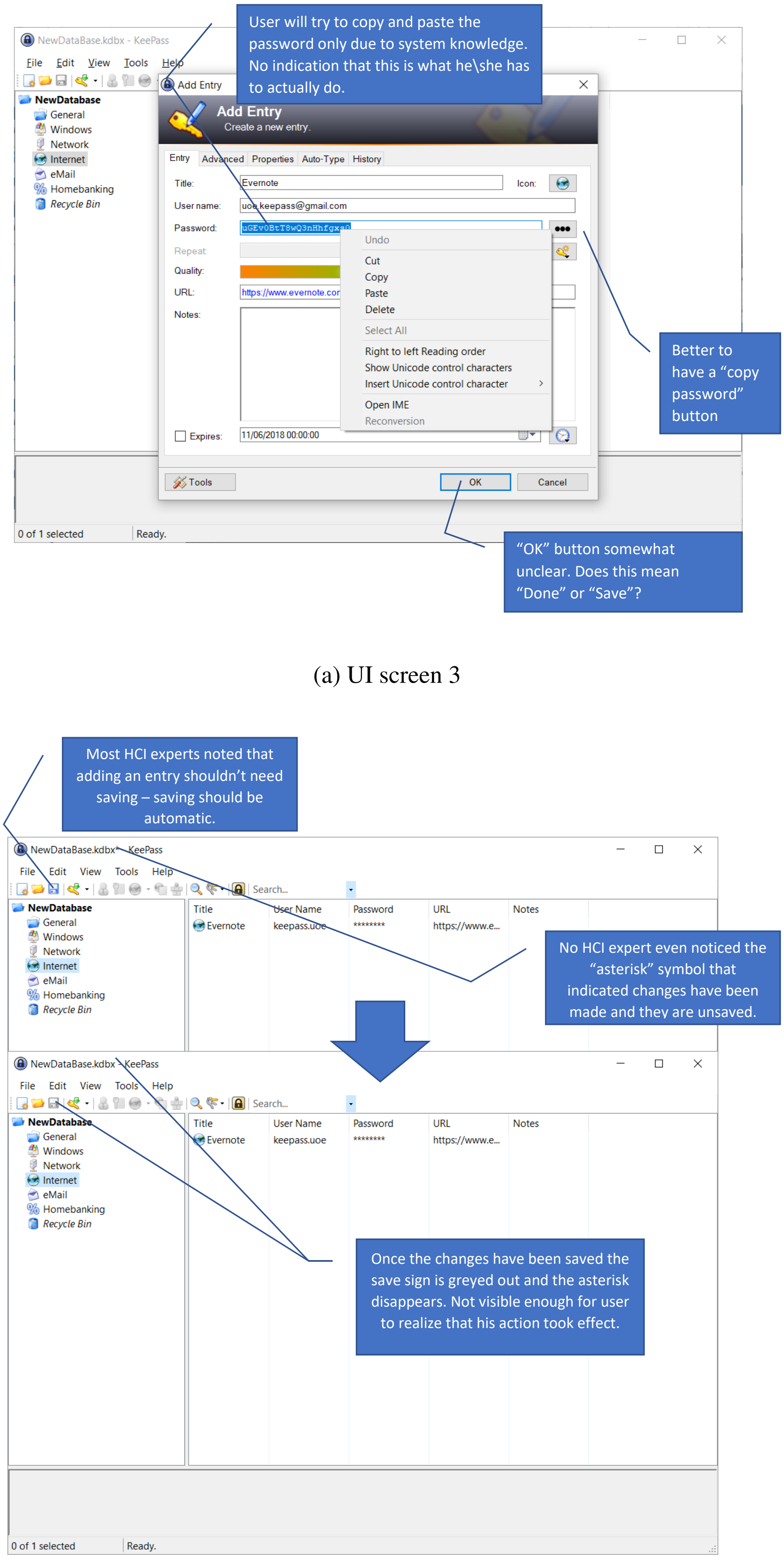
Notes:

☐ Expires: 09/06/2018 00:00:00

0 of 0 selected

(b) UI screen 2

Figure E.1: **Main Task 2** - Depicted Summary of Group Cognitive Walkthrough study



User will try to copy and paste the password only due to system knowledge. No indication that this is what he\she has to actually do.

Better to have a "copy password" button

"OK" button somewhat unclear. Does this mean "Done" or "Save"?

Title: Evernote

User name: uoe.keepass@gmail.com

Password: uGEv0BtT8wQ3nHhfgxs0

URL: https://www.evernote.com

Expires: 11/06/2018 00:00:00

(a) UI screen 3

Most HCI experts noted that adding an entry shouldn't need saving – saving should be automatic.

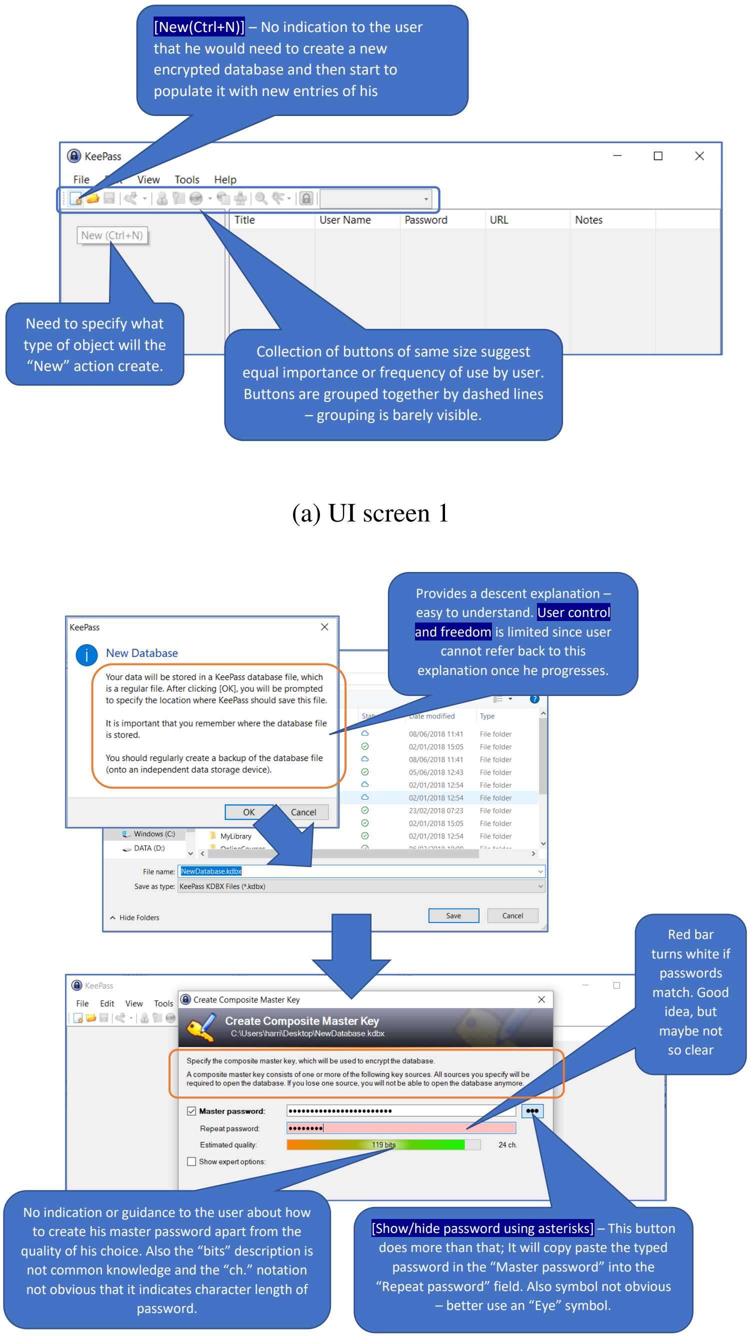
No HCI expert even noticed the "asterisk" symbol that indicated changes have been made and they are unsaved.

Once the changes have been saved the save sign is greyed out and the asterisk disappears. Not visible enough for user to realize that his action took effect.

Title	User Name	Password	URL	Notes
Evernote	keepass.uoe	********	https://www.e...	

(b) UI screen 4

Figure E.2: **Main Task 2** - Depicted Summary of Group Cognitive Walkthrough study - Blue arrows indicate transitions between different UI screens



[New(Ctrl+N)] – No indication to the user that he would need to create a new encrypted database and then start to populate it with new entries of his

Need to specify what type of object will the “New” action create.

Collection of buttons of same size suggest equal importance or frequency of use by user. Buttons are grouped together by dashed lines – grouping is barely visible.

(a) UI screen 1

KeePass

New Database

Your data will be stored in a KeePass database file, which is a regular file. After clicking [OK], you will be prompted to specify the location where KeePass should save this file.

It is important that you remember where the database file is stored.

You should regularly create a backup of the database file (onto an independent data storage device).

File name: NewDatabase.kdbx

Save as type: KeePass KDBX Files (*.kdbx)

Provides a descent explanation – easy to understand. User control and freedom is limited since user cannot refer back to this explanation once he progresses.

Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ Master password: ••••••••••••••••••••••••

Repeat password: ••••••••

Estimated quality: 119 bits 24 ch.

□ Show expert options:

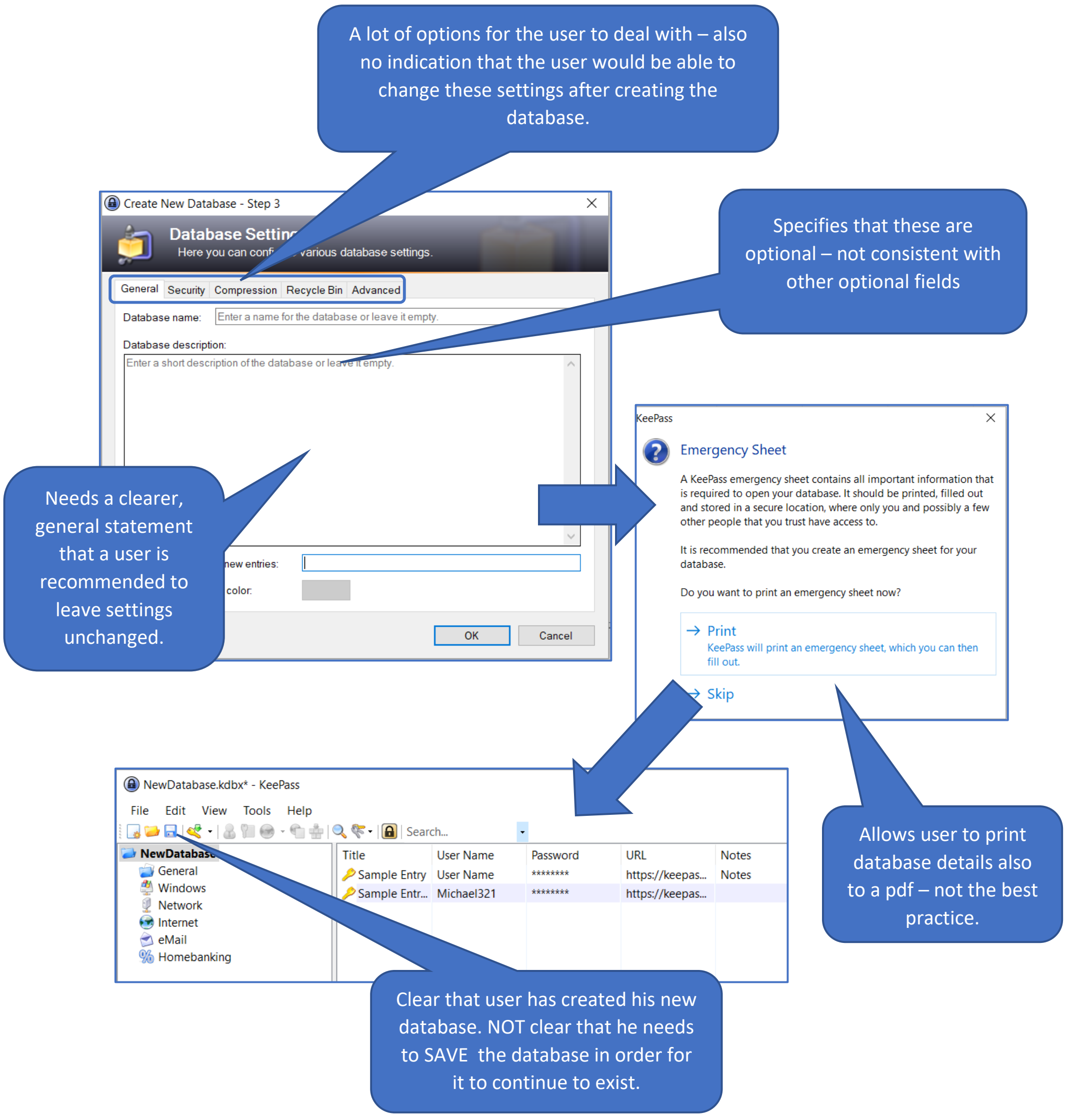
Red bar turns white if passwords match. Good idea, but maybe not so clear

No indication or guidance to the user about how to create his master password apart from the quality of his choice. Also the “bits” description is not common knowledge and the “ch.” notation not obvious that it indicates character length of password.

[Show/hide password using asterisks] – This button does more than that; It will copy paste the typed password in the “Master password” into the “Repeat password” field. Also symbol not obvious – better use an “Eye” symbol.

(b) UI screen 2

Figure E.3: **Main Task 1** - Depicted Summary of Individual Cognitive Walkthrough study - Blue arrows indicate transitions between different UI screens



A lot of options for the user to deal with – also no indication that the user would be able to change these settings after creating the database.

Specifies that these are optional – not consistent with other optional fields

Needs a clearer, general statement that a user is recommended to leave settings unchanged.

Allows user to print database details also to a pdf – not the best practice.

Clear that user has created his new database. NOT clear that he needs to SAVE the database in order for it to continue to exist.

Create New Database - Step 3

Database Settin…

Here you can conf… various database settings.

Database name: Enter a name for the database or leave it empty.

Database description: Enter a short description of the database or leave it empty.

KeePass

Emergency Sheet

A KeePass emergency sheet contains all important information that is required to open your database. It should be printed, filled out and stored in a secure location, where only you and possibly a few other people that you trust have access to.

It is recommended that you create an emergency sheet for your database.

Do you want to print an emergency sheet now?

→ Print

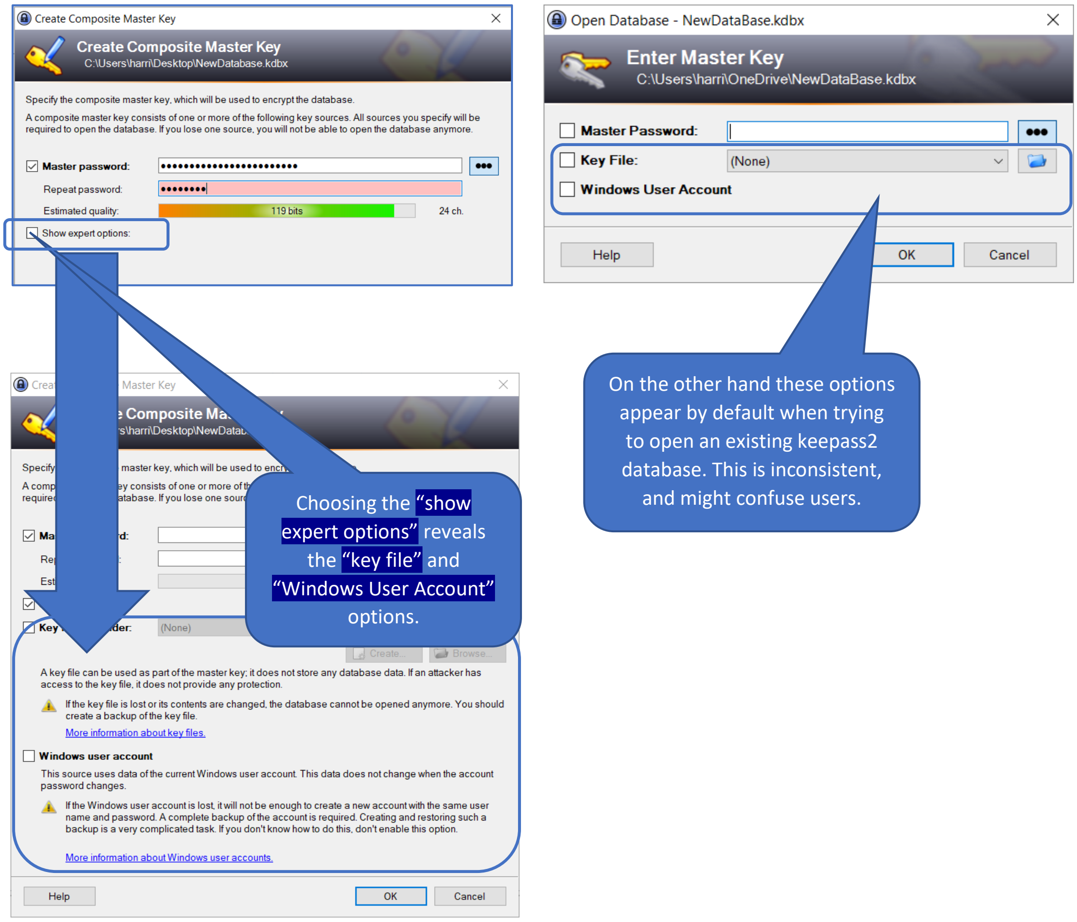
KeePass will print an emergency sheet, which you can then fill out.

→ Skip

NewDatabase.kdbx* - KeePass

Title	User Name	Password	URL	Notes
Sample Entry	User Name	********	https://keepas...	Notes
Sample Entr...	Michael321	********	https://keepas...	

Figure E.4: **Main Task 1** - Depicted Summary of Individual Cognitive Walkthrough study - Blue arrows indicate transitions between different UI screens



Create Composite Master Key

Create Composite Master Key
C:\Users\harri\Desktop\NewDatabase.kdbx

Specify the composite master key, which will be used to encrypt the database.

A composite master key consists of one or more of the following key sources. All sources you specify will be required to open the database. If you lose one source, you will not be able to open the database anymore.

☑ **Master password:** ••••••••••••••••••••••••
Repeat password: ••••••••
Estimated quality: 119 bits 24 ch.
□ Show expert options:

Choosing the "show expert options" reveals the "key file" and "Windows User Account" options.

A key file can be used as part of the master key; it does not store any database data. If an attacker has access to the key file, it does not provide any protection.

If the key file is lost or its contents are changed, the database cannot be opened anymore. You should create a backup of the key file.

More information about key files.

□ **Windows user account**

This source uses data of the current Windows user account. This data does not change when the account password changes.

If the Windows user account is lost, it will not be enough to create a new account with the same user name and password. A complete backup of the account is required. Creating and restoring such a backup is a very complicated task. If you don't know how to do this, don't enable this option.

More information about Windows user accounts.

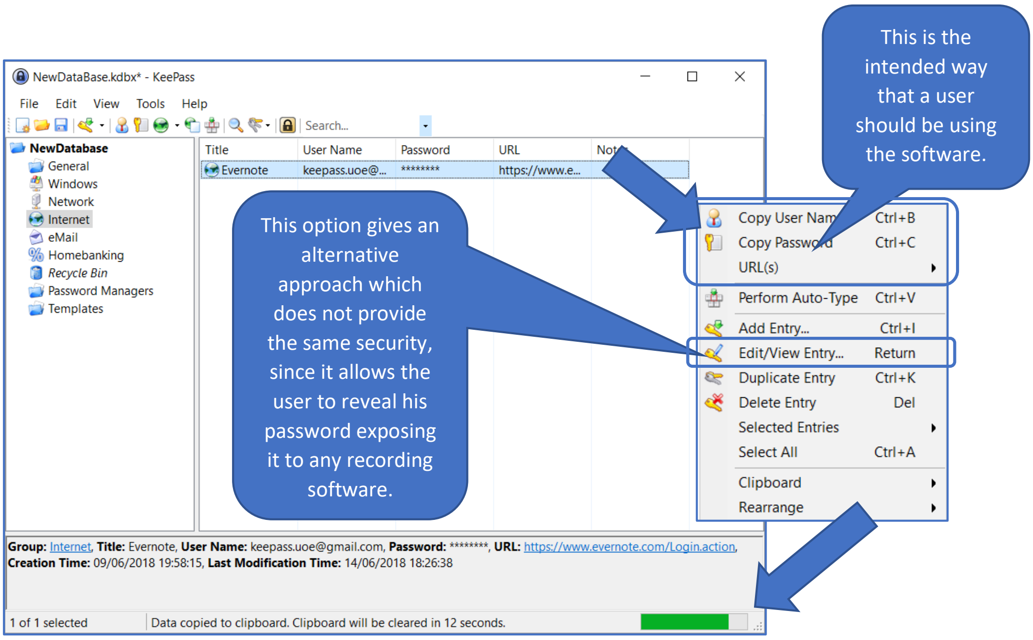
Open Database - NewDataBase.kdbx

Enter Master Key
C:\Users\harri\OneDrive\NewDataBase.kdbx

□ **Master Password:**
□ **Key File:** (None)
□ **Windows User Account**

On the other hand these options appear by default when trying to open an existing keepass2 database. This is inconsistent, and might confuse users.

Figure E.5: **Main Task 3** - Depicted Summary of Individual Cognitive Walkthrough study - Blue arrows indicate transitions between different UI screens. On the left hand side of the figure we depict the UI screen for creating the Composite Master key which is part of the ”new database” wizard. On the right hand side we depict the UI screen for unlocking an already created database



Group: Internet, **Title:** Evernote, **User Name:** keepass.uoe@gmail.com, **Password:** ********, **URL:** https://www.evernote.com/Login.action, **Creation Time:** 09/06/2018 19:58:15, **Last Modification Time:** 14/06/2018 18:26:38

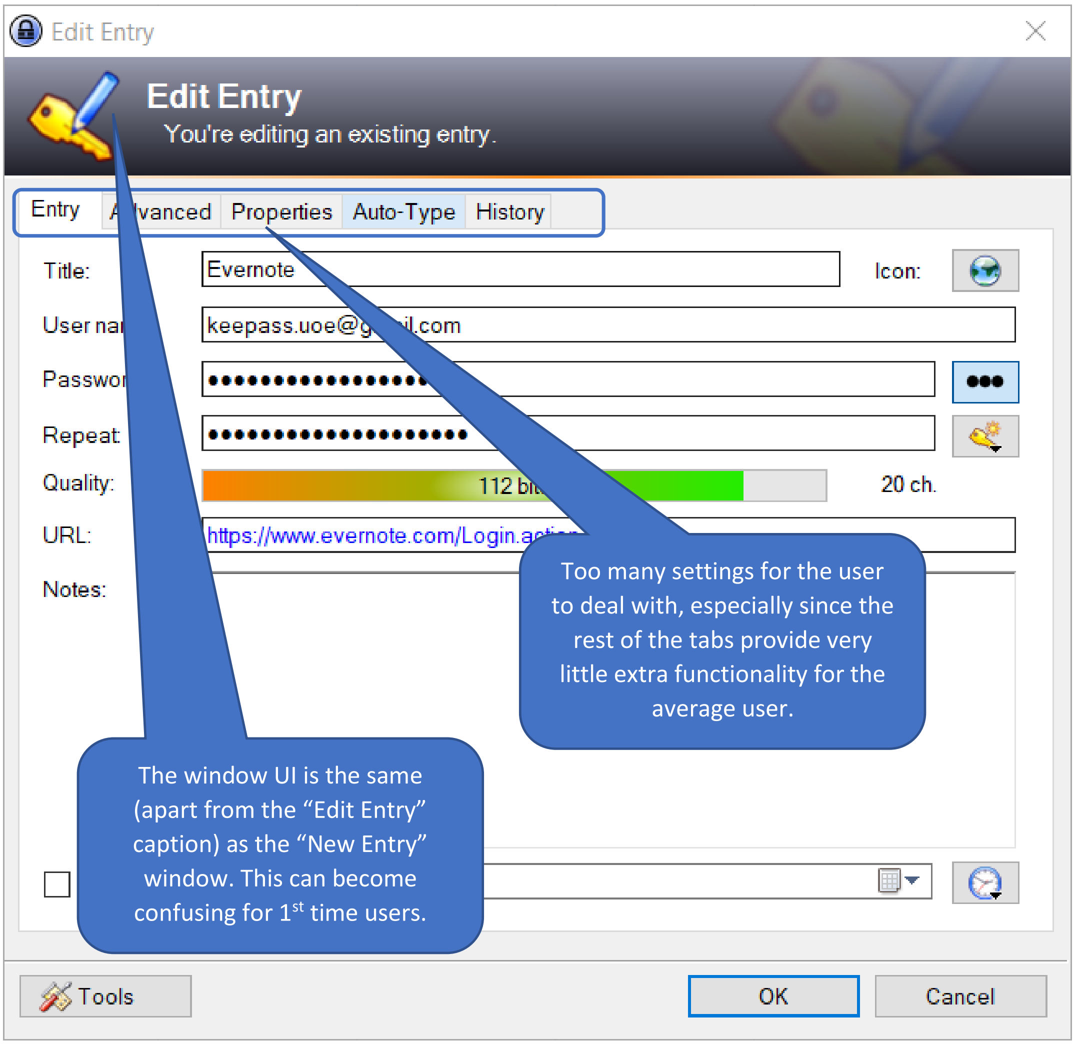
1 of 1 selected

Data copied to clipboard. Clipboard will be cleared in 12 seconds.

This is the intended way that a user should be using the software.

This option gives an alternative approach which does not provide the same security, since it allows the user to reveal his password exposing it to any recording software.

Figure E.6: **Main Task 3** - Depicted Summary of Individual Cognitive Walkthrough study - Blue arrows indicate transitions between different UI screens



Title: Evernote

User name: keepass.uoe@gmail.com

Password: ••••••••••••••••••••

Repeat: ••••••••••••••••••••

Quality: 112 bits 20 ch.

URL: https://www.evernote.com/Login.action

Notes:

Too many settings for the user to deal with, especially since the rest of the tabs provide very little extra functionality for the average user.

The window UI is the same (apart from the “Edit Entry” caption) as the “New Entry” window. This can become confusing for 1st time users.

Figure E.7: **Main Task 4** - Depicted Summary of Individual Cognitive Walkthrough study