|  |  |
| --- | --- |
|  | **Qatar University**  **College of Engineering**  **Department of Computer Science and Engineering** |

Shared Notes SDP1

Table of Contents

[1. Essa 4](#_Toc146050428)

[2. Khalifa 4](#_Toc146050429)

[3. Mohamed-Dhia 5](#_Toc146050430)

[4. Youssef 5](#_Toc146050431)

[Extra ideas (Draft) 5](#_Toc146050432)

# Essa

# Khalifa

# Mohamed-Dhia

Security features needed in a password manager: ([ref](https://www.passwordmanager.com/bitwarden-review/))

1. Encryption (compare AES with customized encryption algorithms XChaCha20)
2. No-knowledge architecture or Zero Trust secu [(ref)](https://www.keepersecurity.com/fr_FR/resources/glossary/what-is-zero-trust/) <https://www.keepersecurity.com/fr_FR/resources/glossary/what-is-zero-trust/>
3. 2FA

Learn from LastPass breach

* In remove credential: should be make an optional feature of “recently deleted” that permanently deletes deleted credentials after 1 day/closing the app??
* If we’ll be doing the auto-fill feature then our program should already know the websites before-hand. Therefore, we cannot accommodate any account. Only if we make it dynamically check the list of “known websites” when a new credential is added, then gives it a label “I know this one” (maybe with an auto-fill feature when user enters the account name first time + a warning message if the app don’t know it)
* In Search credentials, we can make it “contains” not only “start with

In 5.5: rule of 60/30/10 for UI color & design

sdp2 survey: question: trust better for local or international company for password manager

why your project??

>> gotta check the -ve of web extensions of saving pass, & why our app is better? (for eg. not only websites)

**Sdp**

Including advice boxes “careful from attacks”

Include tuto at beginning

# Youssef

# Password Manager System Overview Notes:

# Actors

**User**

[*Use Cases*](https://userlab.utk.edu/files/papers/ruoti/2021/simmons2021systematization.pdf)

|  |  |  |
| --- | --- | --- |
| **Essential** | **Recommended** | **Optional** |
| **E1-** Setup Manager | **R1-** Audit Credentials | **O1-** Share Credentials to non-users (with password rotation) |
| **E2-** Register Credential | **R2-** Modify Settings | **O2-** Store Sensitive Data |
| **E3-** Update Credential | **R3-** Recover Access | - |
| **E4-** Remove Credential | **R4-** Wipe Account | - |
| **E5-** AutoFill Credential | - | - |
| **E6-** Manually Enter Credential | - | - |
| **E7-** Generate Password | - | - |
| **E8-** Export Credentials | - | - |
| **E9-** Import Credentials | - | - |
| **E10-** Lock Manager | - | - |
| **E11-** Unlock Manager | - | - |

## *Additional Notes:*

### Possible Testing Method

***Cognitive walkthroughs*** are a form of expert review in which a

usability expert completes a given task with an assigned tool. While

completing this task, the evaluator will role-play, responding to

the tool’s interface and taking actions only as the role-played user

would. As they complete the tasks, the evaluators apply a

think-aloud protocol, describing what they see, identifying how

they discover features, and describing any confusion they encounter

as they complete the task.

### Weaknessess and inconveniences found in most password managers

**Challenge01: ((E6) Manually enter credential):**

Due to the difficulty of reading and entering these

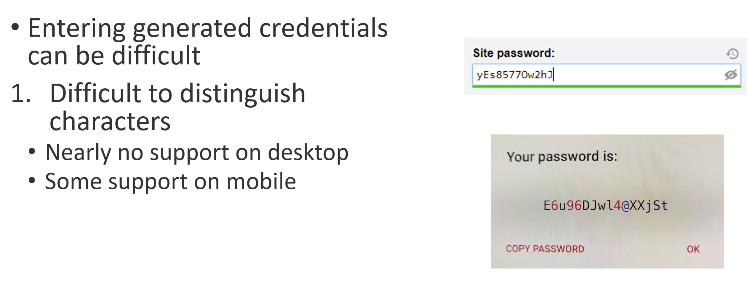
passwords (e.g. 0 and O), they noted that this was the most difficult and

annoying of all the tasks they completed. Only 1Password X aided

this process, highlighting characters based on character class

((E6-P3) Distinguish password characters), making it easier to read

the password to be entered on the mobile device. **Possible Solution: Modality-aware password generation.**



Research could investigate password generation that factors in the devices where

the password will be entered, making it easier to enter generated

passwords. This research could be modeled after and extend the

work of Greene et al. [12], which examined generating passwords

that were easier to enter on mobile keyboards.

**Challenge02: Interface Designs.** While completing tasks, evaluators were

consistently disappointed with various aspects of the managers'

interfaces. Most commonly, problems arose due to confusion when

attempting to locate features or settings **((R2) Modify settings)**. This

difficulty was often caused by a combination of those features

and settings being deeply nested in menus or named using non-

obvious vernacular. For example, Chrome groups manager features

under the label "autofill", which may not be a meaningful term to

many users (as it was not initially to our evaluators), as opposed

to "privacy and security", where many users might expect it. Issues

such as these caused our evaluators to spend considerable time

locating these items, and for some users may prevent them from realizing the feature or setting exists. **Possible Solution: Searchable Settings with proper wording.**

**Challenge03:** s**everal managers used password strength meters within their password generators ((E7) Generate password)**. These meters included visual indicators

and colors to describe the strength of the password. The evaluators

noted that this gave them confidence that their selected generation

settings were secure. Ideally, managers could find ways to

**incorporate more of these simple, easy-to-understand indicators**

**through the manager to help users identify when they are using**

**the managers correctly.**

### Essential Use Cases and Descriptions

**E1: Setup Manager (login, signup)**

- Users can create a new account (signup) or log into an existing one (login) to access their password manager.

**E2: Register Credential**

- Users can add new login credentials (username and password) for websites or applications to their password manager.

**E3: Update Credential**

- Users can edit and update existing login credentials stored in their password manager.

**E4: Remove Credential**

- Users can delete login credentials they no longer need from their password manager.

**E5: AutoFill Credential**

- The password manager can automatically fill in login credentials when users visit websites or apps, streamlining the login process.

**E7: Generate Password**

- Users can use the password manager to generate strong, unique passwords for new accounts or to replace weak passwords.

**E8: Export Credentials**

- The password manager can export login credentials for usage across multiple devices, ensuring users have access to their passwords everywhere.

**E9: Import Credentials:**

- The password manager can import login credentials for availability purposes.

**E9: Lock Manager**

- Users can lock their password manager, preventing access to stored credentials for security purposes or even set a timeout period for the application to lockout.

**E10: Unlock Manager**

- After locking, users can unlock the password manager by only providing their master password.

### Recommened Use Cases and Descriptions

**R1: Audit Credentials**

- Users can review and audit the security of their stored credentials, checking for weak or duplicated passwords.

**R2: Modify Settings**

- Users can customize settings within the password manager, such as changing master passwords, enabling or disabling features, and adjusting security preferences.

**R3: Recover Access**

- In case of a forgotten master password, users can initiate a recovery process to regain access to their password manager account.

**R4: Wipe Account**

- Users can permanently delete all data and credentials stored in their password manager account, ensuring data privacy in the event of an account closure.

# 

# Extra ideas (Draft)

Classify accounts into sub-groups (since they can be many)