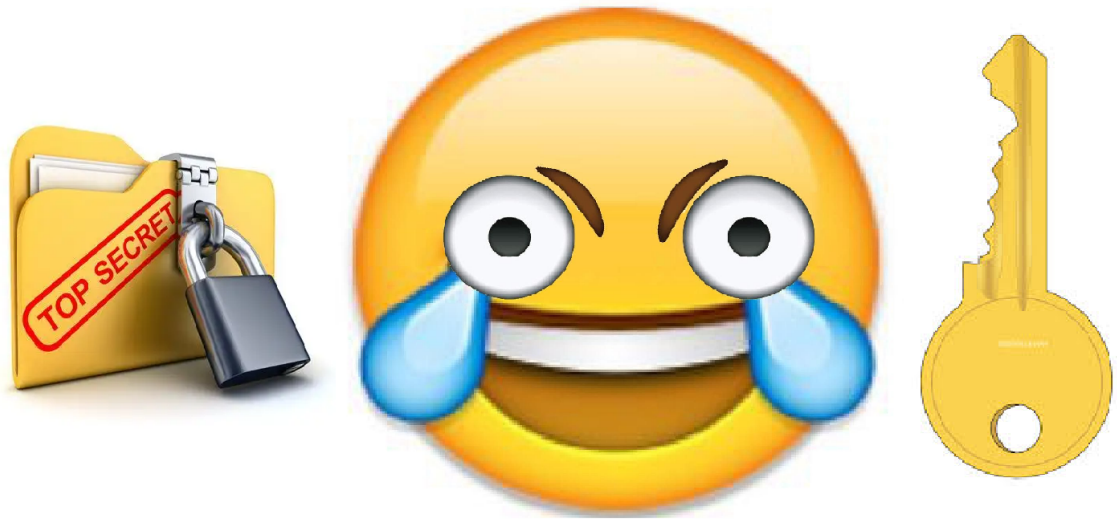
Year 12 HSC Testing and Evaluation Report 2022

FunnyPass



Ken Liu

HSC Software Major Work

Contents

Testing Methods

The product was tested via various methods. Locally performed unit, module and system testing was conducted using a driver program which entered in data from a test data table containing randomly generated and specifically selected data to test boundaries and special cases. Beta testers were gathered via Discord and their feedback was collected via direct messages on the application. Their feedback was analysed for general sentiment and if mentioned, any specific areas with which questions were raised.

Evaluation Methods

The product was evaluated based on how much it conformed to the standards as set out in the planning and designing phase. Some changes to the criteria were necessary as issues came up during implementation which caused some of the original criteria to be inappropriate or unachievable. These include the usage of specific encryption schemes being more complex than initially expected and issues with time coming towards the end of the development process.

Beta Test Results

General sentiment:

- Interface works but is relatively basic.
- Buttons need icons.
- The extra features included are nice but there could be more.
- Help button needed on each screen.
- User documentation needed.

Specific areas to improve:

- Need a button to go back to the login screen once logged in.
- Vault screen needs a scrollbar to see more login details.

Test Data Table

Test table for password strength check

Input	Expected output	Actual output
Null	Password too weak	Password too weak
L	Password too weak	Password too weak
';<OK#p9 FOKVDLvo	Password success	Password success
password	Password too common	Password too common

Q;9LDSKD	Password too weak	Password too weak
kIP(#OUFJPidi;., fe; ,Ef ,	Password too weak	Password too weak

Benchmarking and Quality Assurance

All feasible key features as promised in the original have been included.

Analysis of final product

Generate:

Success criteria:

- ☐ Generates a password in < 0.3 seconds.
- ☐ Generated password is at least 8 characters long.
- ☐ Generated password contains at least one symbol, number, uppercase and lowercase letter.

Encrypt:

Success criteria:

- ☑ ~~Encrypts all data in < 0.5 seconds.~~
- ☑ ~~Encrypted data only decryptable with the original key.~~
- ☑ ~~Proper usage of an encryption scheme.~~

Decrypt:

Success criteria:

- ☑ ~~Decrypts data in < 0.5 seconds.~~
- ☑ ~~Decrypted data is only decryptable with the original key.~~
- ☑ ~~Proper usage of the same encryption scheme.~~

Login:

Success criteria:

- ☑ ~~Able to allow access in < 0.3 seconds. I.e. compare entered and master passwords.~~
- ☑ ~~Keeps master password hashed.~~
- ☐ Shows main menu in < 0.2 seconds.

Storage:

Success criteria:

- ☑ ~~Stores all passwords in encrypted form.~~
- ☑ ~~Keeps master password and recovery key securely hashed.~~
- ☑ ~~Able to store new and remove entries without causing errors.~~

Add:

Success criteria:

- ☑ ~~Able to check dictionary of passwords in < 0.1 seconds.~~
- ☑ ~~Able to send entered details to encrypt while keeping them intact.~~
- ☑ ~~Switches screen back to main menu in < 0.2 seconds.~~

GUI:

Success criteria:

- ☑ ~~All algorithm buttons perform their function in < 0.7 seconds.~~
- ☐ All required elements have standard icons.
- ☑ ~~Non-algorithm buttons work in < 0.2 seconds.~~
- ☑ ~~Interface is consistent and ergonomically sound.~~
- ☐ All other applicable design principles adhered to.

The program achieves most outcomes at the level specified. Not all success criteria have been met though. The product satisfies all basic functionality requirements but leaves extra features out.