

CS35L Project Proposal - Shrine

Team Members

- Alan Yu azy7@g.ucla.edu
- Chentyi Zhang chriszhang09@g.ucla.edu
- Randy Gu rgu855@163.com
- Jin Zhang jinzhang2001@g.ucla.edu
- Hanry Xu hanryxu@ucla.edu

Overall Idea:

The idea of the project for our team is to implement a password vault/manager. It will facilitate users to store and organize their passwords in a more convenient and accessible way.

Motivation:

Major password managers list TOTP passcode as a paid-feature (here we exclude Authenticator apps e.g. Google Authenticator from password managers), and none implement HOTP passcode as a feature. With this project, users may copy and fill one time passcode directly from their browser, without the trouble of going to their mobile app, read and remember the one-time passcode, and submit to the website.

Features:

- Upload data from Client to Back-end
Username and passwords provided by users will be saved in the server database.
- Display Dynamic Data to the User
Added username and passwords will be displayed on the website, available for viewing, copying, and editing.
- Meaningfully search through server-side data
Users may search for website name or username to retrieve certain passwords.
- Unique Feature #1
Aside from the basic website name, username, and password, store two factor authentication (TOTP and HOTP) keys and dynamically generate dynamic codes.
- Unique Feature #2
Provide a random password generator to create stronger and safer passwords.
- Unique Feature #3
Notify users of repeated and unsafe (breached) passwords.

Schedule:

- Week 4 10.18 -10.24
 - initiate git repository
 - research on possible technical implementations of the following
 - Programming language and framework to use
 - Suitable database to use
 - Algorithms:
 - random password generator
 - Two factor authentication generator
 - Encryption implementations between transmissions (front-end to back-end, back-end to database)
- Week 5 10.25-10.31
 - create database structure/framework
 - front-end design draft
 - back-end work on standalone features i.e. random password generator
- Week 6 11.01-11.07
 - implement CRUD functionality
 - implement two-factor authentication functionality
- Week 7 11.08-11.14
 - Implement search function with database
 - implement password check function
- Week 8 11.15-11.21
 - Implement back-end search function
 - implement sign-in functionality
 - finish front-end implementation
- Week 9 11.22-11.28
 - integrate front-end with back-end
 - final revisions and improvements
- 12.01 Source code deadline
- 12.03 Report