

Criterion E: Mr. Hammond's Evaluation

Criterion: Application stores the username and password in a live database that updates along as we register more users. The user can also reset their password

Evaluation: Criterion met. The username and password of the employees are stored in a google sheets user database.

Criterion: Application checks the live database and ensures all of their credentials match. It also updates the database when the password is changed

Evaluation: Criterion met. The program searches through the database for the username and password entered, and the badge. And if they match, then it lets the employee login, otherwise displays an error message.

Criterion: Application allows the user to register a new user.

Evaluation: Criterion met. The program allows new employees to register a username and password into the database that is persistent.

Criterion: Application completely change their password and update the old reference of the user object in the database to contain the new password

Evaluation: Criterion met. The user is able to change their specific password inside the database, while keeping the username and badge the same

Criterion: Application contains a graphical user interface that makes the user experience very easy

Evaluation: Criterion met. The program is a runnable application with a graphical user interface that is simple and easy to use.

Criterion: Application is able to successfully lock and unlock doors; there is also a fail safe method to lock and unlock the door so that cannot be accidentally unlocked

Evaluation: Criterion met. The user can unlock specific doors from specific buildings. There is a fail safe that prevents accidental locking and unlocking. There is a textbox entry system for each building to unlock a specific door entered to avoid accidental locking/unlocking of the doors. Added a button to implement another layer of protection from accidental locking and unlocking. Users must click the button to lock or unlock a valid door entered in the textbox entry.

Criterion: Application has an error system that tells the user what is going on in all cases, the user always knows what is going on

Evaluation: Criterion met. If the user enters the wrong input like a random text, or they enter a door that is not relevant to the building they are currently inside: the error message tells the user that they entered an invalid door. And the door does not exist.

Final Meeting with the Client:

In the final meeting with Mr. Hammond, he expressed that the product meets all the requirements and satisfies all the needs in the park. He said, “The product is outstanding and works great. However, one features that I wish was implemented was a option to login with the username or login with the badge id, since the assigned badge ID by the company is unique to every employee anyways. But I did not ask for that, and you have delivered what was asked for. And it works great. Thank you.”

Recommendations for Future Development:

As stated by Mr. Hammond, having the program that is versatile and compatible with logging in through a username or logging in through an id would be a cool implementation that increases the convince as some people prefer to enter their id instead of username and visa-versa.

I personally would like to implement a searching algorithm that checks if the username being entered is not already instantiated with a password, and if it is then the employee should not be able to create a new account and would either reset their password or create a new account with a new username.

I would also like to implement a MySQL database in place of a google sheets database, as it is more easily accessible and mutable than a google sheets database. It is easier for developers to implement, and it is a lot more secure than a google sheets database that generates an API endpoint.

Word Count: 621