Bug Documentation

Whenever you are reporting a bug, please fill out the following sections:

Issue Type:

Exception Name (if applicable):

Bug Description:

Reported By:

Date Reported:

Date Resolved:

Replication (with steps):

The Fix/Solution:

Root Cause:

Takeaways:

For issue type, I think the naming convention should be something like this:
*Object*_Bug#(number)

For example, if there was a bug with the access code, the name should be AccessCode_Bug#1. If you found another bug with the access code, that bug would be called AccessCode_Bug#2.If there was a bug with the database, the name would be Database_Bug#1.

For the exception name section, if there was an actual exception on the program, then list it. If not, then skip this.

For the bug description, please describe how the bug happened. If you are able to provide an exception and its details, that would be good to put here too.

For the replication section, have a numbered list for the steps to replicate the bug.

For the fix/solution for the bug, please describe how you found the fix for the bug. Also, write down if someone else has confirmed if the fix is working (Might make this its own section).

Preferably, whenever multiple bugs are found, please leave some space in between the other bugs on this document.

Issue type: Database_Bug#1

Exception Name (if applicable): SQLite Exception

Bug Description: When running the BucHunt program from our team's fork of the repo, attempting to log in using the provided admin credentials will throw the following exception: SqliteException: SQLite Error 1: 'no such column: a.AccessCodeId'.

Reported By: Keillor Cogswell

Date reported: 03/25/2024

Date resolved: 03/27/2024

Replication (with steps):

1. Clone the team repo to user's local development environment.

2. Go into the Package Manager Console, then run the command Update-Database

3. The database will be constructed, with an error about there being no "AccessCode" table.

4. Run the BucHunt program without debugging. This will launch BucHunt in the user's default browser (e.g., Firefox).

5. Enter the username as waltonca@etsu.edu and the password as YMXH@9J!72kM6Em, then press the "Log In" button.

6. The application will display the aforementioned SqliteException error.

The Fix/Solution:

Testing a clone of the BucHunt repo from GitHub user adverseengineer, at the behest of SE II member Dan, the program was tested without performing the Update-Database command. Upon the program starting, admin credentials were entered into the login page, and the login process was successful, also allowing access to the Admin Portal.

Logan switched the team's fork of the repo over to the one by adverseengineer, also testing it on his local development environment and confirming that the admin credentials worked successfully.

Yu found the missing db file that was causing the error and tried to implement it. There was a problem with transferring the db file across branches

Tyler discovered that the git ignore file had ignored the db files, and was able to delete the content for it in git ignore. Deleting the git ignore allowed the files to move from one source control branch to another. After adding the db file to the branch, you can add back the git ignore.

Root Cause: It was discovered by Yu that the team's fork of the repo was missing the ScavengeRUS database, which had not been added until the day after this issue was initially reported.

Takeaways: When messing with files, looking at the git ignore file can be an issue when moving files around in each version.

Issue Type: UserController_Bug#1

Exception Name (if applicable): Argument Exception

Bug Description: In the CreateUsers method, there is a bug that crashes the program. The method itself is to create a bunch of users at once. No documentation was given.

Reported By: Grayson Burke

Date Reported: 04/3/24

Date Resolved: 04/8/24

Replication (with steps):

1. Log into the Admin Account (You need to have gmail.txt in the ScavengeRUS folder before you can do anything)
2. Go to the Admin Portal
3. Click on Batch Create Users
4. This should pull information from Users.csv then crash the application stating that there was an Argument Exception.

The Fix/Solution: In Users.csv, some of the carrier names were "Tmobile" instead of "TMobile". Changing this allowed for the admin to create a batch of new users.

Root Cause: The Users.csv file misspelled a carrier name, and threw an error because of it.

Takeaways: Make sure that the parameters in Users.csv are correct, for example entering the carrier correctly (you can see all the carriers in Functions.cs).