

Document Updater Test Report

After two days of testing the Document Updater application, I have produced the following report to summarise my findings.

In its current state, the application has a lot of potential, but requires significant change in order to be used in production. The application generally functions well on a small scale, with files tending to be placed in the correct locations after execution.

However, my testing has found that issues arise when:

- A file name contains an accented/special character
- A user attempts to update more than 64 files
- Two people in our dataset have the same surname
- The allowlist is empty

These are significant issues because:

- A lot of people have names with accented characters or special characters
- The application cannot scale if there are issues when updating more than 64 files at a time
- A lot of people have common surnames

Bug reports

- “Storing two people with the same surname causes loss and corruption of details” - <https://github.com/liamquinn14/generating-test-data-challenge/issues/2>
- “File names with accented/special chars renamed after running program” - <https://github.com/liamquinn14/generating-test-data-challenge/issues/5>
- “Impossible to add more than 64 files to 'finals' when either 'originals' or 'updates' are empty” - <https://github.com/liamquinn14/generating-test-data-challenge/issues/4>
- “Empty allowlist crashes program with IndexError” - <https://github.com/liamquinn14/generating-test-data-challenge/issues/1>

Enhancements

Due to well-documented issues when the application tries to handle people with common surnames, I have submitted the following enhancement request, as I believe that it proposes a solution to several of the application’s bugs and limitations.

- “Enhancement: Store details by ID rather than surname” - <https://github.com/liamquinn14/generating-test-data-challenge/issues/3>

Summary

The Document Updater application shows promise but requires attention to address the identified issues. If the application was deployed to production in its current state, and used real data, a significant proportion of our data would become corrupted or lost. In order to handle the edge cases involved in a real dataset, we need to restructure the application to use unique IDs as file names, so that common surnames do not lead to a loss of data.