Feature - Recover Function

- Programmer: Liam Tiemon
- Date: 04/18/2018

Problem Definition

Users often make changes which they then need to revert. The current problem is that if a user makes a change on the wiki, they don't have an automatic way of going back to previous versions of that specific page.

Adding a recover function will allow users to revert changes without having to remember what was on the wiki page before they appended more information.

Requirement

Previous versions of wiki pages need to be saved in a database so that users can recover specific versions.

User stories

1. As a user, I want my data to be saved so that in case I need to revert some progress, I can go back to a previous version.
2. As a user, I want to automatically go back to a previous version of my wiki page so that I don't have to manually edit a page and remember what used to be on it.

Design

User story 1 & 2

Interface

- API name: `RestorePage.py`.
- Input: the parameter is the name of the wiki and its url path.
- Action: older files from the database are pulled and pushed to the front to be the most recent copy.
- Output: the page is now reverted and the user now has an older copy of the wiki.

Detailed design

- To trigger the restoration of the page, users will select which page they want to revert to in the GUI.

- The markdown file is selected from the url and the commitID.

- Using basic Python file I/O, we write what was stored in the database to a new file and push it up to the top of the database using `ArchivePage.py`'s `store` method.

Implementation

The dependencies used in this feature are:

- `from io import open` - to pull contents of markdown files for storage.
- `ArchivePage` - to store the page after the changes have been reverted.

User story 1 & 2

The Python code for implementation can be found at Github location.

Tests

- This test uses `psycopg2` to use the database.
- The tests for the added features are included in attached `test_web.py`. This test is can be found from our project repository.
- In this test case, two test files are created and stored using `store` from `ArchivePage.py`. Then the test restores the previous stored page.

Checking the requirements

- User story1 - Test results show that versions of pages are being stored for access later.
- User story2 - Test results show that users can revert changes back to a previous page.

Limitations of tests

1. These tests fail to cover the UI portion of the wiki.
2. Unit tests are mostly for backend code.

User documentation

Markdown files are saved in a database everytime edits are made to the wiki page, so that users can revert to an older copy of the wiki if they need to.