Spooky Ghosts: Jun Hong Wang, James Yu, Thomas Zhang
softdev pd7
p00--design doc
2022-10-27

Python file(s):
 - contains code to start flask server
 - code for rendering the main webpage template
 - maybe a python file to deal with the sql stuff separately
 - maybe another one for the push requests? (but can be in app file)
HTML File(s):
 - main html file loaded by Python file will have text, with a form on the side requesting login
credentials (username and password)
 - if the login is successful, a page with editable text will be displayed (maybe using html
contenteditable attribute, but we're not sure if this will allow permanent changes)
   - if the user edits the text, it should be permanently changed? (but previous versions could still
be recoverable like a real wiki?)
 - if login not successful, nothing changes, stays on main page
SQL file:
 - we won't use it for text storage, since we didn't think it made sense for this purpose
 - instead, we will have a table with user credentials, like username, password, and account type
  - these credentials will be matched with the information that is submitted through the form
  - 2 types of accounts, admin and user
  - only admin can edit, user can only view

Sitemap:
 - when the python file is run and link is opened, the main page will load, displaying uneditable
text
  - will be on localhost root (127.0.0.1)
 - if logged in admin account, then user can view another page with editable text
   - editable text on 127.0.0.1/edit (to achieve, maybe use push request in the form so user submitted
information doesn't appear here)

Priorities:
 - make a template that displays uneditable text for mainpage
 - add a button that links to another page with editable text
 - make the python app that renders the html files
 - make the sql tables (maybe in a python file) to compare user credentials
  - for now, user credentials will be hard-coded, so accounts will be automatically set up

Python (run in terminal) -> Flask (creates server, serves html) -> HTML (has text)
HTML -> python (login or change file) -> sql table (for comparison of user credentials)