Team Sharp Cat: Victoria Gao, Alvin Wu, Jonathan Lee, Madelyn Mao
SoftDev
P0: Quick! Quick! - Design Document
2020-12-17

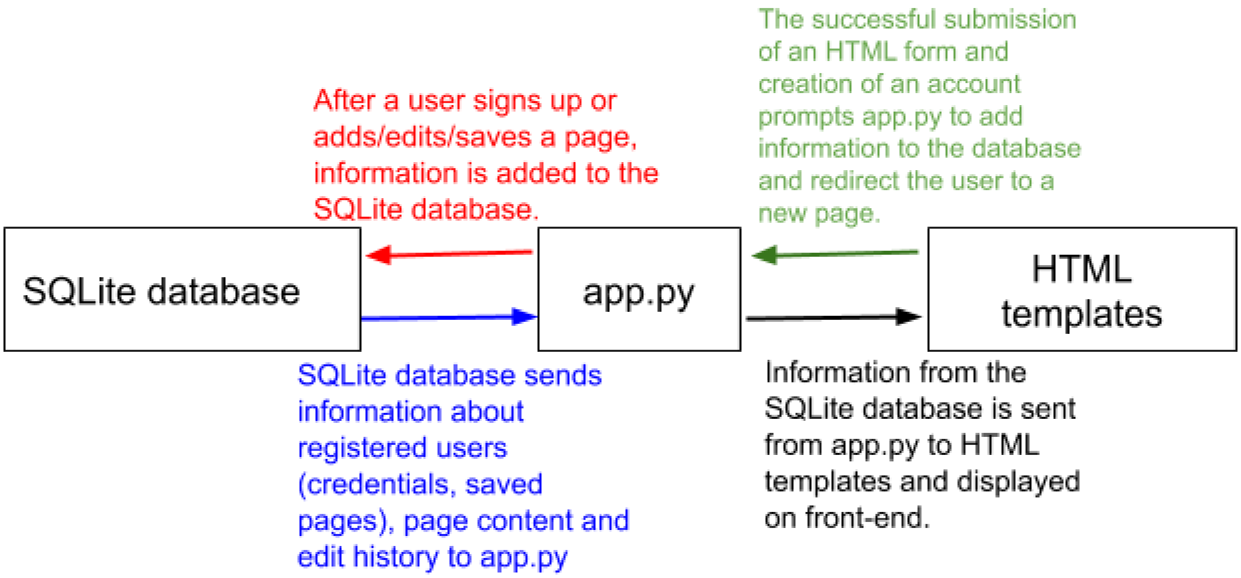
Components

- app.py: connects backend files (i.e. database) to frontend files (i.e. html templates). This Python script will redirect users to another webpage based on the buttons they click and their input in HTML forms.
- SQLite database: contains four tables:
    - Login: Each row of this table contains a user's username and password.
    - Content: Each row of this table contains a unique wiki page name, category, URL to the page, URL of an image, and text in the body of the page.
    - History: Each row of this table contains a timestamp, page name, username, text on the page before edit, URL of image on the page before edit, text of page after edit, and URL of image on the page after the edit. On the frontend of the edit history webpage of a wiki page, we will display up to two previous edits of this wiki page.
    - Saved: Each row of this table contains the URL to a saved page and a username.
- HTML templates:
    - index.html: home page that lists a few categories of pages the user can click on and sign up/login buttons.
    - login.html: contains an HTML form that allows the user to enter username and password. When the user clicks the "Submit" button, app.py will check if the credentials match an entry of the Login table in the SQLite database. Users will be redirected to response.html or error.html depending on whether they logged in successfully.
    - signup.html: contains an HTML form that allows users to create a username and password for their account. app.py will check if the usernames chosen by new users already exist in the Login table and verify if they fulfill other requirements (i.e. password length). Users will be redirected to response.html or error.html depending on whether their account was created successfully.
    - error.html: If users fail to login or create an account successfully, they will be directed to this page and be told what went wrong. Below the error message, there is a "Sign Up" button that redirects the user to signup.html and a "Login" button that redirects the user to login.html
    - response.html: If users login in or create an account successfully, they will be directed to this page. This page will have a navigation bar with options like "Add Page","Saved Pages", and "Logout". There will also be a few categories of pages that the user can choose from. If users click the "Add Page" option, they will be

redirected to addpage.html. If users click the “Logout” option, they will be redirected to the home page index.html.

- savedpages.html: contains links to the pages that a user saved. There will be a button that allows a user to delete links to saved pages.
- page.html: contains text and images about a topic that is received from the database when the user clicks on a page name. There will be an “Edit” button at the top of the page that redirects the user to edit.html when it is clicked. There will also be an “Edit History” button that redirects the user to history.html
- history.html: lists up to 2 most recent changes to a page.
- edit.html: contains an HTML form with fields that allow the user to add text and images to an existing page. It has two buttons: “Save Changes” will change the content of a wiki page and “Discard Changes” will redirect users to the page they were viewing before clicking the “Edit” button.
- addpage.html: contains an HTML form with fields that allow the user to create a page name and add text and images to the new page. The user will select a category that their page belongs to from a hard-coded dropdown menu.
- logout.html: displays a message showing that the user logged out successfully. The user can go back to the home page index.html by clicking a button.

Component Map


The successful submission of an HTML form and creation of an account prompts app.py to add information to the database and redirect the user to a new page.
After a user signs up or adds/edits/saves a page, information is added to the SQLite database.
SQLite database
app.py
HTML templates
SQLite database sends information about registered users (credentials, saved pages), page content and edit history to app.py
Information from the SQLite database is sent from app.py to HTML templates and displayed on front-end.

Database Organization

- Login table:
  - There will be a row for each user.
  - There will be 3 columns: ID, Username, Password

ID	Username	Password
0	wk1	t0g2ka
<non-negative integer>	<username>	<password>

- Content table:
  - There will be a row for each page.
  - There will be 6 columns: ID, Name, URL, Category, Image, Text

ID	Name	URL	Category	Image	Text
0	Siamese Cat	cats/Siamese-Cat	cats	https://cattime.com/assets/uploads/2011/12/file_2667_siamese-cat-breed.jpg	The Siamese cat is one of the first distinctly recognized breeds of Asian cat ...
<non-negative integer>	<page_name>	<category>/<page-name>	<category>	<url_to_image>	<displayed_text>

- History table:
  - There will be a row for each edit to a page.
  - There will be 8 columns: ID, Time, Name, User, TextB4, ImageB4, TextA, ImageA

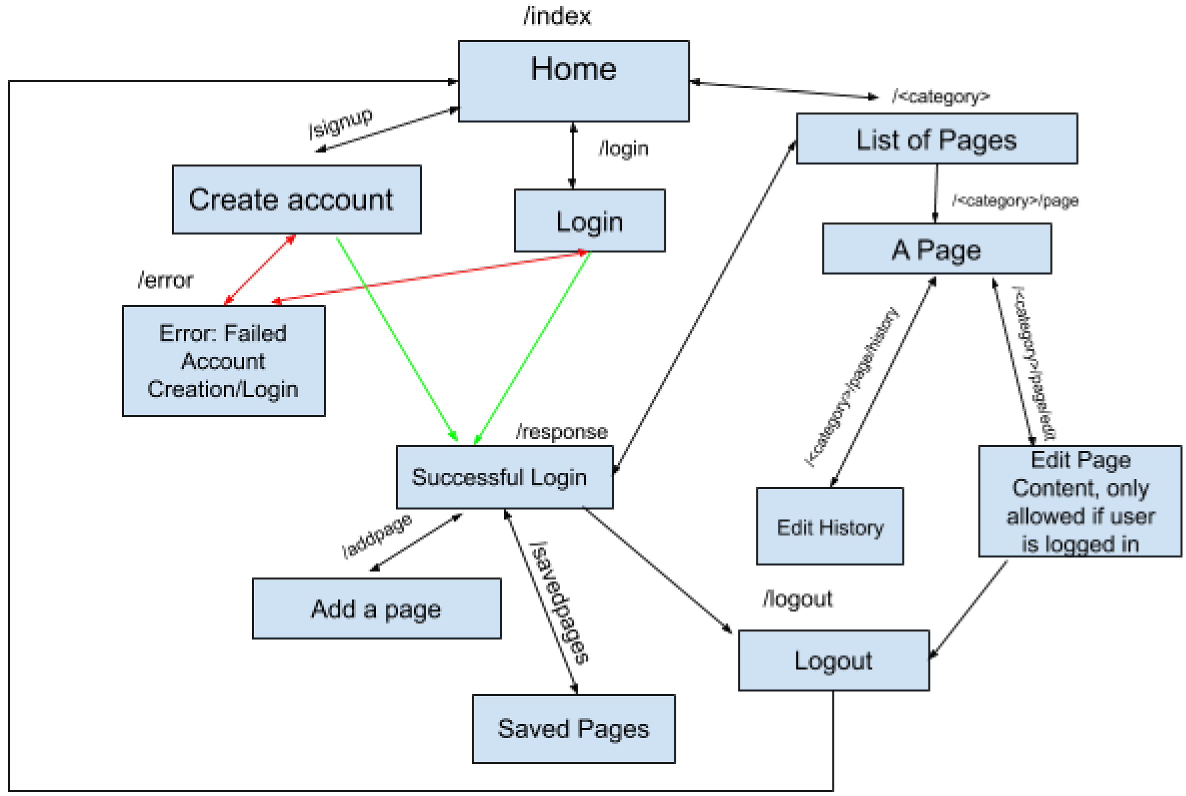
ID	Time	Name	User	TextB4	ImageB4	TextA	ImageA
0	2020-12-20-17-00	Siamese Cat	wk1	The Siamese cat is one of the first distinctly recognized breeds of Asian	https://cattime.com/assets/uploads/2011/12/file_2667_siamese-cat-breed.jpg	The ideal Siamese is a medium-sized, svelte, refined cat with long tapering	https://d17fnq9dkz9hgj.cloudfront.net/breed-uploads/2018/08/siamese-detail.jpg?bust=153556

				cat ...		lines, very lithe but muscular ...	7164&width=355
<non-negative integer>	<date wiki page was last updated, year-month-day-hour-minute>	<pg_name>	<username>	<text_before_edit>	<url_to_img_before_edit>	<text_after_edit>	<url_to_img_after_edit>

- Saved table:
  - There will be a row for each saved page.
  - There will be 3 columns: ID, URL, User

ID	URL	User
0	cats/Siamese-Cat	wk1
<positive_integer>	<category>/<page-name>	<username>

Site Map for Frontend



Tasks

- app.py and Python files (as necessary for “middleware” modules): Alvin Wu
- Login & Saved tables in SQLite database: Jonathan Lee
- Content & History tables in SQLite database: Victoria Gao
- HTML/CSS: Madelyn Mao