

BAA TIME (Alejandro Alonso, Theodore Fahey, Ivan Mijacika, Emma Buller)

SoftDev

P00 -- <Move Slowly and Fix Things/Project/Team has been contracted for one of two scenarios requiring a user to register, update information pertaining to them, and view other's information.

2021-11-09

Time Spent: 30 Hrs

Scenario Chosen: **Scenario Two**

HTML Framework:

Create New Blog	Edit Existing Blog	See Other Blogs
	Blog 1	User Bob
	Blog 2	User lowndown
	<h1>WELCOME</h1> <div>Logout</div>	

- ☐ form and flask based login page
 - ☐ Gives option to create a new account if no login.
 - ☐ Old logins will be stored in db
 - ☐ If login does not exist = redirect to error page
 - ☐ Cookies to track login.
 - ☐ Can multiple users login at the same time???
- ☐ show three option (home page): create new blog, edit existing blog (with dropdown menu to choose which blog), see other blogs (with dropdown menu to choose which blog)
 - ☐ Only page that has logout page
 - ☐ Redirects you to homepage, gets rid of cookie
 - ☐ See other blogs → display selected blog
 - ☐ See other blogs tab -> display all blogs -> select blog -> display posts
 - ☐ Edit existing blog → display selected blog with option to create new post, and option to edit on each existing post

- ☐ Edit existing blog tab -> blog entry
- ☐ Dropdown menu will only have YOUR blogs
- ☐ Create new post
- ☐ Edit old post
- ☐ Create new blog → display create new post, by extension adds blog (like github repo folders)
 - ☐ Submit button for making blog will redirect you to homepage and add to usernames and create new database for blog entry

Python Framework:

- ☐ Create a route for each user's page.
- ☐ File for sqlite various sqlite methods
 - ☐ Add column or edit table value (for adding new blog to user row)
 - ☐ Check all column (check if login is correct/if username exists when creating new account)
 - ☐ Also used for the see other blogs tab
 - ☐ Check if database already exists

Sqlite3 Framework:

- ☐ Multiple databases (see below)
 - ☐ Users database
 - ☐ username
 - ☐ password
 - ☐ user_id
 - ☐ Num_blogs
 - ☐ Blogs database containing all of the blogs of all users
 - ☐ Each row will include blog_title, user_id, num_blogs, blog_id, and last_date_edited.
 - ☐ New row created when user makes new blog
 - ☐ Posts database containing all of the posts from all blogs
 - ☐ Each row will include post_title, post_text, blog_id, user_id, and last_date_edited.
 - ☐ New row created when user adds a new post to their blog
- ☐ How we are using sqlite
 - ☐ Edit post = UPDATE
 - ☐ Add post = INSERT

Users Database

username	password	user_id	num_blogs
“joe”	“1234”	1	1

All of the blogs are stored within one separate database, differentiated by blog_title, user_id, and blog_id:

“Blogs”

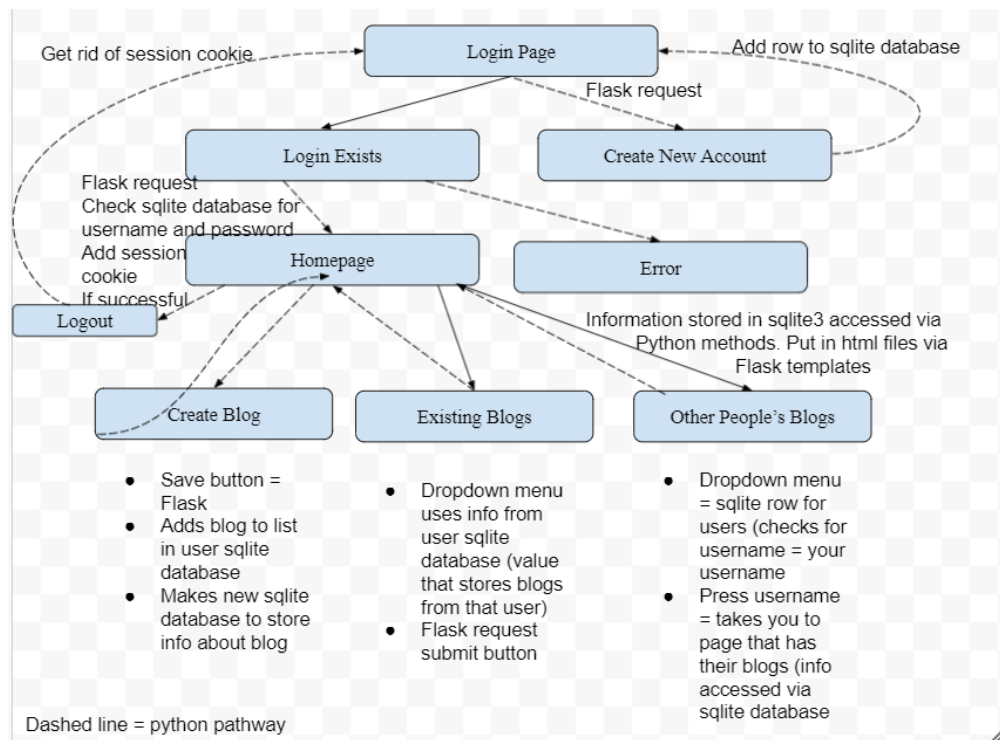
blog_title	user_id	num_blogs	blog_id	last_date_edited
“blogName”	“MyPost”	1	“5”	2021-11-9 12:34:56

posts

post_title	post_text	blog_id	user_id	post_id	last_date_edited
“MyPost”	“blahblahblah”	“5”	1	“10”	2021-11-9 12:34:56

Roles & Responsibilities:

- ❖ Alejandro Alonso → PM
 - blogsdb.py, auth.py, main.py
- ❖ Theo Fahey → Devo
 - auth.py, html templates
- ❖ Ivan Mijacika → Devo
 - blogsdb.py, html templates
- ❖ Emma Buller → Devo
 - auth.py, main.py, html templates



This is slightly outdated