

Tiffany Yang, Claire Song, Jessica Yu, Jackie Zeng
Team Anything
SoftDev
P00
2024-10-23
Time spent: 1 hour
Target Ship Date: 2024-11-7

Scenario Two: Your team has been contracted to create a web log hosting site, with the following features:

- Users will have to register to use the site.
- A logged-in user will be able to
 - Create a new blog
 - Update their blog by adding a new entry
 - View and edit their own past entries
 - View the blogs of other users

Front end:

- ☐ HTML templates and CSS (if time): create layout of web pages; display tables and information from databases

Back end:

- ☐ Flask app: host web pages
- ☐ Database handling: create and link databases; sort data

Components:

- User accounts
 - User sessions
 - User login/logout system
- Routes
 - Route to submit new blogs
 - Route to edit past blogs
 - Route to view past blogs
- SQLite db for storing blog info
 - Page_id
 - Usernames/logins
 - Password hash

Database: (build_db.py creates this)

blogs	
id	INTEGER
user_id	INTEGER
title	TEXT
content	TEXT
users	
id	INTEGER
username	TEXT
password_hash	TEXT

Database Organization: (DB browser for SQLite): stored data are placeholder values

user Table:

	id	username	password_hash
	Filter	Filter	Filter
1	1	hi	script:...
2	2	yo	script:...
3	3	sup	script:...
4	4	ok	script:...
5	5	aa	script:...

blog Table:

	id	user_id	title	content
	Filter	Filter	Filter	Filter
1	1	1	hi	soliekhdsfjhsdkfhksjhfsdkjfhsdjkfhs...
2	2	2	yo	yosdjhfdskjfhdskfjdshfkdsfhfsdkfjsdf...
3	3	3	sup	suppppppppppppppppppppppppppppppppp...
4	4	4	hi	hi
5	5	5	a	aa
6	6	5	a	a

Task Assignment:

Backend: Tiffany and Jackie

- ☐ Build_db.py (creates the DB)
- ☐ __init__.py

Front end: Claire and Jessica

- ☐ blogPage.html
- ☐ editPage.html
- ☐ editpage.html
- ☐ home.html
- ☐ login.html
- ☐ my_pages.html
- ☐ newPage.html
- ☐ newpage.html
- ☐ signup.html
- ☐ user_pages.html
- ☐ Users.html

CSS: Whole Group

- ☐ blogPage.css
- ☐ editpage.css
- ☐ home.css
- ☐ login.css
- ☐ my_pages.css
- ☐ newpage.css
- ☐ signup.css
- ☐ style.css
- ☐ users.css

Site Map:

Nav bar: (On all pages)

- Outgoing routes:
 - Home (/)
 - Users (/users)
 - My Pages (/mypages)

Home (/)

- Outgoing routes:
 - Login (/login)
 - Sign-Up (/signup)

Login (/login)

- Outgoing routes:
 - Home (/) if successful login

Sign-Up (/signup)

- Outgoing routes:
 - Login (/login) after account is made

View Blog list (/pages/<int:user_id>)

- Outgoing routes:
 - View selected blog of user (/view_blog/<int:page_id>)

Create Page (/create)

- Outgoing routes:
 - View Blog (/view_blog/<int:page_id>) after page is made

Edit Page (/edit/{page_id})

- Outgoing routes:
 - View Blog (/view_blog/<int:page_id>) after editing is done

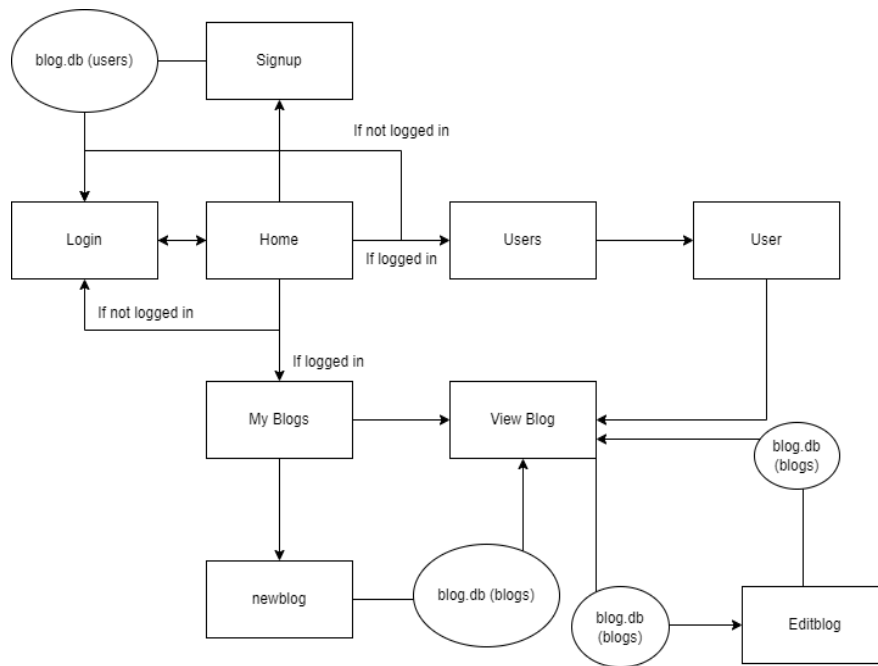
User Blogs (/users)

- Outgoing routes:
 - Selected User's Pages (/pages/<int:user_id>)

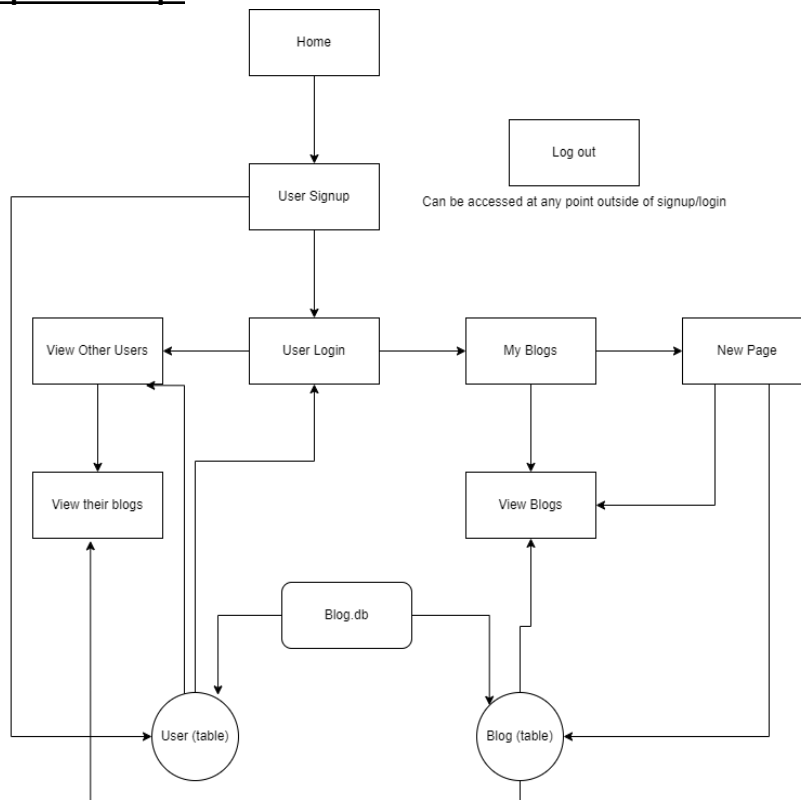
My Blogs (/mypages)

- Outgoing routes:
 - New blogs (/create) to make new ones
 - Any blogs made by user (/pages/<int:user_id>)

Site Map:



Component Map:



Page Draft

<div><h3><u>Home (Page name)</u></h3><hr/><div>HomeUsersMy PagesLog In/Out</div><hr/><div><div>Login</div><div>Sign-up</div></div></div>	<div><h3><u>Users</u></h3><hr/><div>HomeUsersMy PagesLog In/Out</div><hr/><div>User1 User2 User3</div></div>
<div><h3><u>My Pages/User Pages</u></h3><hr/><div>HomeUsersMy PagesLog In/Out</div><hr/><div>[New page] Page1 Page2 Page3</div></div>	<div><h3><u>Page1 [Title]</u></h3><hr/><div>HomeUsersMy PagesLog In/Out</div><hr/><div>Page Details here [if own page, edit button here] [if editing, pre-filled textbox with previously saved details here]</div></div>

<p><u>__init__.py:</u></p> <p>(/) Template: home.html Description: Shows a button to</p> <p>(/signup) Template: signup.html Description: Provides a form for new users to create an account</p> <p>(/login) Template: login.html Description: Provides a login form for existing users</p> <p>(/logout) Template: None Description: Clears the session to log the user out and redirects to the home page</p> <p>(/create) Template: newPage.html Description: Displays a form to create a new blog post</p>	<p>(/view_blog/<int:page_id>) Template: blogPage.html Description: Shows a specific blog post</p> <p>(/<int:page_id>/edit) Template: editPage.html Description: Provides an editing form for the blog post, restricted to the author</p> <p>(/users) Template: users.html Description: Displays a list of all users in the system</p> <p>(/pages/<int:user_id>) Template: user_pages.html Description: Shows all blog posts created by a specific user</p> <p>(/mypages) Template: my_pages.html Description: Lists all blog posts created by the logged-in user</p>
--	--