

Team wiki_nani

Joan Chirinos, Soojin Choi, Raunak Chowdhury, Anton Danylenko

Core Fxnality

- Add page
- Edit page
- Auth
- Search
- Landing page
- Guest access
- Account Creation

Additional Fxnality

- Remove page
- Roll-back
- Table of contents/navigation utility
- User-submitted comments on pages
- Timestamp of the last edit

Components and Relations

Login box

The login box will appear at the top left corner of every page in the site. If the user is logged in, the box will display a welcome message and a logout button. If the user is not logged in, the box will display a username field, a password field, a login button, and a create account button. The login button will lead to “/login”, which will authenticate the user and store their logged-in status in cookies. The create account button will attempt to create an account using the given username and password.

Search Bar

The user can type in a query into the search bar; the app will then check the database for any valid hits, and will return those hits with links to their pages.

A “valid hit” may be defined as any row in the database whose title contains part or all of the search query.

Results

When the user submits a search query, they will be greeted with a results page. This will include a search bar at the top, pre-filled with their current query, and results of the search. Each of these will be a link to a page, possibly followed with a short excerpt from the page contents.

Individual Pages

When the user clicks on a specific result on a result page, they will be taken to the individual page of that result. The page will include a title, author of the page, the text content, and a button taking you back to the search results.

Add Page

In order to add a page, the user has to be logged in. On the Add Page page, there will be a form with input/textarea fields for the title and the contents of the page and a submit button. When the form is submitted, it will be added to the database.

Edit Page

The user has to be logged-in in order to edit a page. The edit page will first include a dropdown from which the user selects a page to edit; a form will be generated with the input/textarea fields filled in with the current title and content. When that form is submitted, the database will update to reflect the user’s edits.

Database Schema

Entries:

- Title
- Content
- Edited-by (ID) [1-TO-1 RELATION WITH **LOGINS**]

Logins:

- ID
- Username
- Password

Delegations

Joan Chirinos

- Project Manager
 - Delegating
- Displaying wiki pages
- Drawing up diagrams, flowcharts, etc.

Soojin Choi

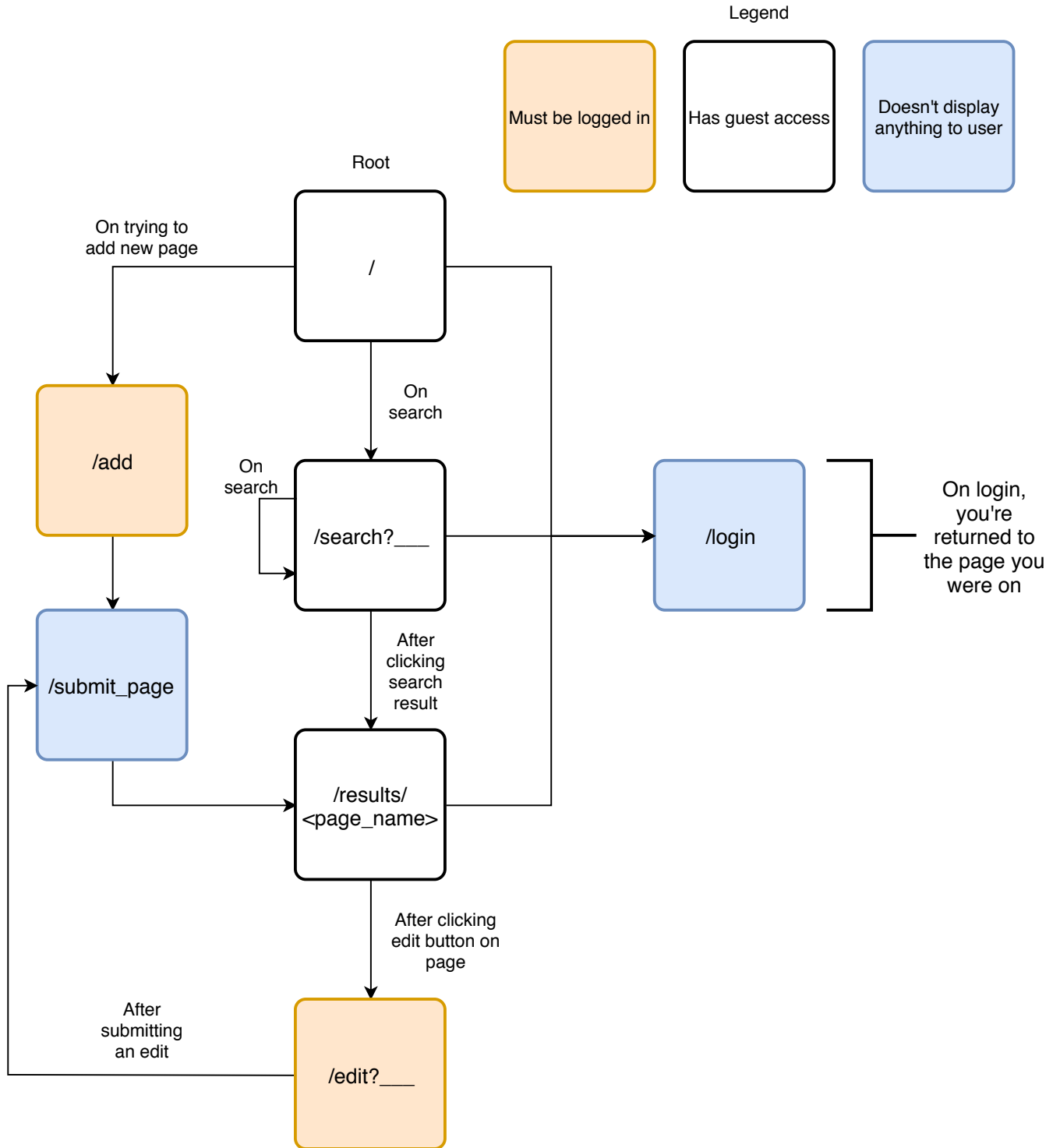
- Login system
 - Add users to the database
 - Use session to determine if someone is logged in or not
 - Authenticate users

Raunak Chowdhury

- Add/edit pages (possibly remove pages later)
 - Specifications included above in the Add and Edit Page sections

Anton Danylenko

- Searching
 - Search for the keyword in titles of pages (maybe later implement searching through the page contents themselves)
 - Split across whitespace
 - Iterate through the resulting list while:
 - Using string.find() method to find the closest match in the database
 - Adding the match to another list
 - Return said list



login
box

1

Title

Search

Submit

/search? query = <query>

Add Page

2

/search? query = <query>

login
box

1

Title

<query>

Submit

/search? query = <query>

Results

result_1

/results? page_name = <name>

result_2

..

result_3

..

...

result_n

..

/results? page_name = <name>

login
box ¹

Title

by ~~original-user~~ →

Edit page ²

/edit? page_name = <name>

Content

Contributors

<c1>

<c2>

...

<cn>

/add

Title

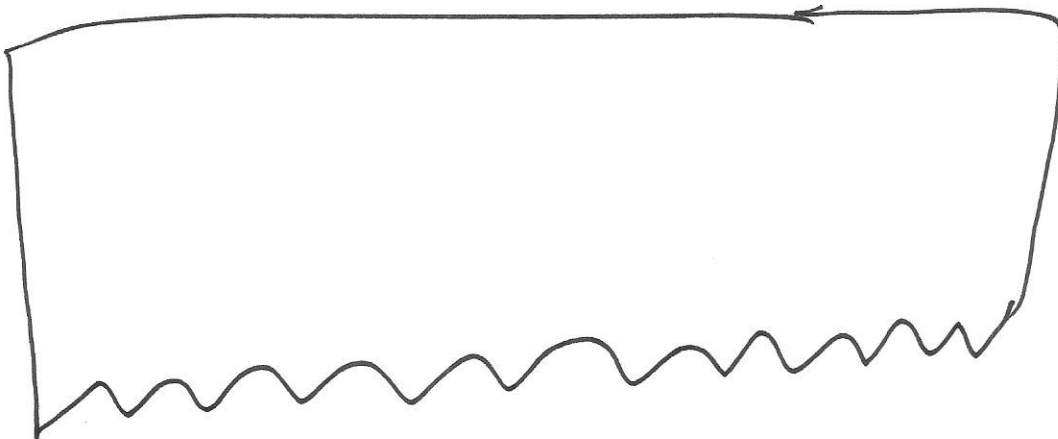
Content

method: POST

/submit-page

/edit? editing = <name>

Title & name > ← not editable
Content



Submit

Method: POST

/submit-page →

/login

- Doesn't display anything to user
- If login button clicked
 - attempts a login
 - ↳ if failure, flashes error
- If create acc. button clicked
 - attempts to create account
 - ↳ if failure, flashes error
- Redirects to previous page

/submit - page

- Takes POST data from /add or /edit
- If adding, adds title and content to DB
- If editing, replaces old content w/ new one in DB
- Redirects to /result page

1

If user not logged in:

A hand-drawn login form with two input fields labeled 'Username' and 'Password'. Below the fields are two buttons: 'Log in' and 'Create acc.'.

login is input w/ ~~name~~ value
= "login"

create acc. is input w/ value
= "create"

both name = "button"

If user logged in:

A hand-drawn user interface for a logged-in user. It shows a message 'Welcome <user>!' and a button labeled 'log out'.

2 Only appears if user is logged in