# Tiffany Yang, Tracy Ye, Chloe Wong, Stella Yampolsky
# Team <placeholder>
# SoftDev
# P00
# 2024-10-21
# Time spent: 3
# Target Ship Date: 2024-10- TBA

Program Components:

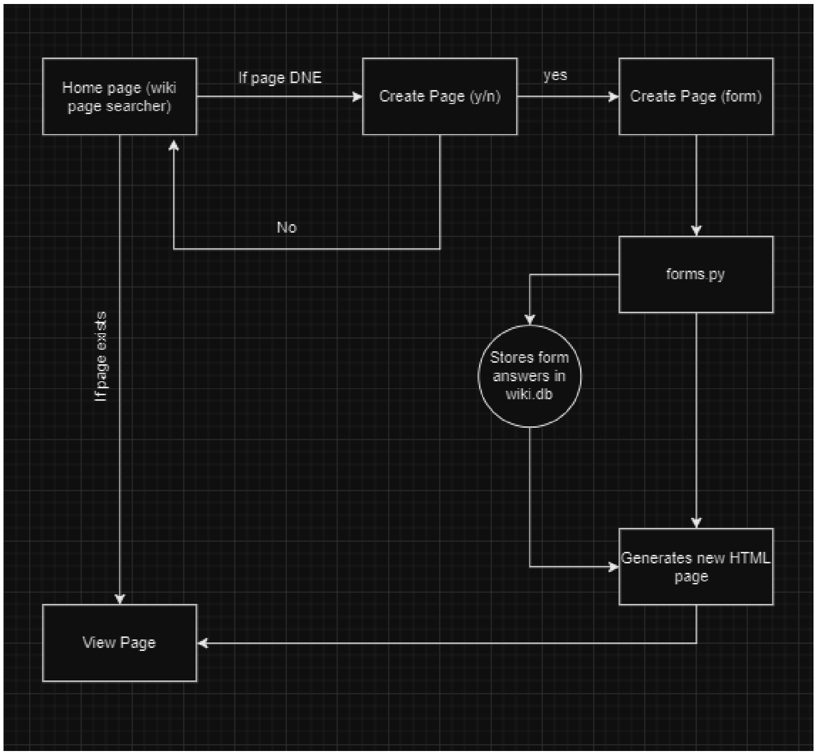
Front end:

- □ 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

{ Database module organizes data for Flask app → Flask app serves data using HTML templates }



Files:

Front end:	Backend:
□ NewPage.html *{ page with forms and runs forms.py upon submission; form answers = wiki.db record and fields }* □ Editpage.html □ [load pages].html (actual pages)	□ wiki.db (stores page data) □ app.py (main script) □ forms.py (generates the new pages)

Break Down :

```
wiki. py:
 (‘/’)
        render home.html
('/search')
        if (response] DNE in DB wiki
                render newpage.htm)
        If [response] exists in DB wiki
                render viewpage.html
('/result')
         if [yes] was selected
                Render [1]
         if [no] was selected
                 Redirect (‘/’)
 ('/load page)
        If [username] in DB banned
                redirect (‘/’)
         if [Username ] good
                render [2]
        if [passkey] != passkey
                redirect (‘/’)
        if [passkey good
                render [1]
(‘/editpage’)
        redirect(‘/’)
```

{ edit pages through a form block

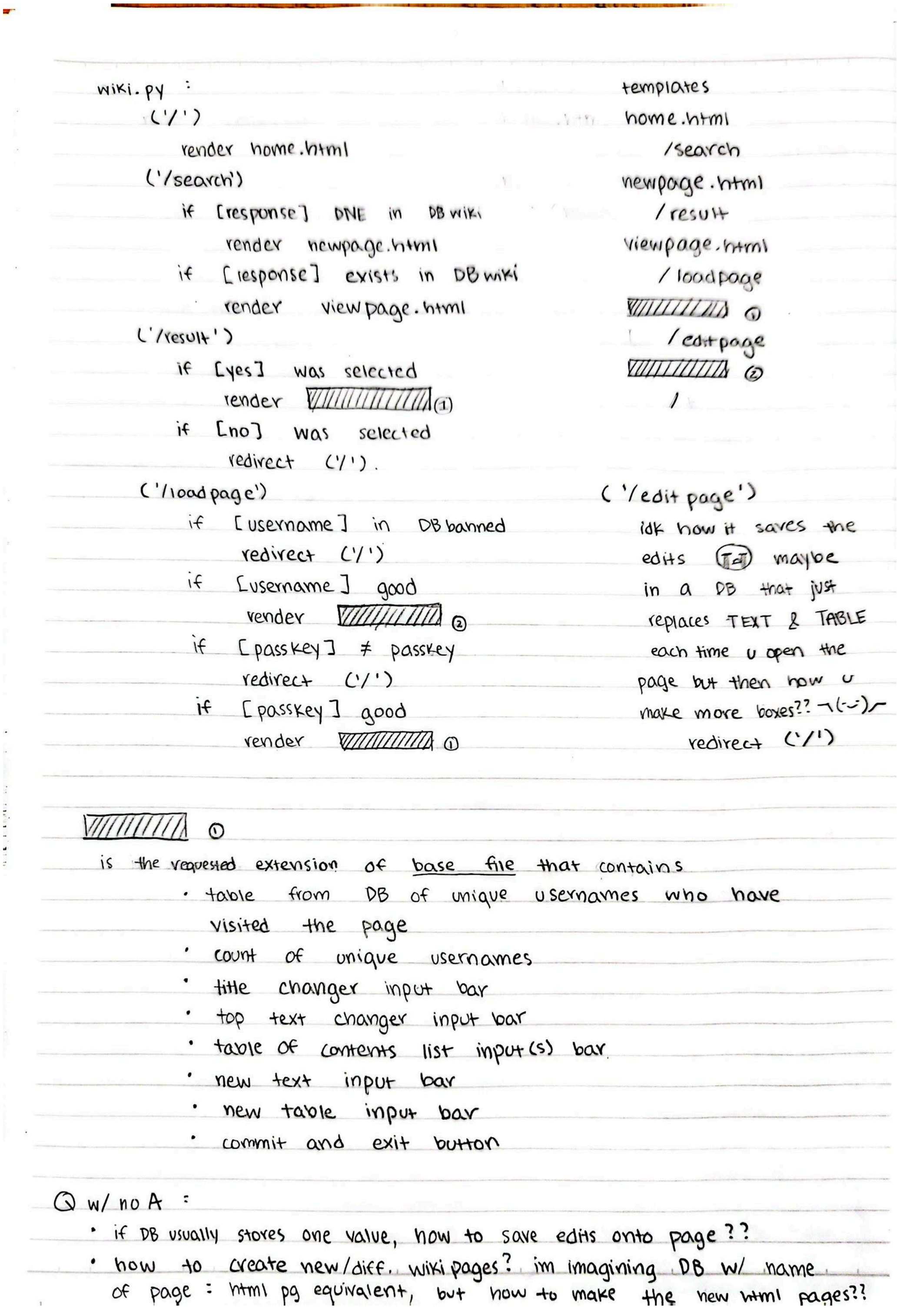
"edit," it loads the page with a pre-filled-in form block with the pre-existing text, and then whenever you save the edits, the database is updated }

```
Templates
Home.html
        /search
Newpage.html
        /result
Viewpage.html
        /loadpage
[render 1]
        /edit page
[render 2]
        /
```

{ render 1 is the editable page for the requested wiki topic }

{ render 2 is the viewable page for the requested wiki topic }

uty if you want to print the below.



```
wiki.py :
    ('/')
        render home.html
    ('/search')
        if [response] DNE in DB wiki
            render newpage.html
        if [response] exists in DB wiki
            render viewpage.html
    ('/result')
        if [yes] was selected
            render ▨ ①
        if [no] was selected
            redirect ('/').
    ('/loadpage')
        if [username] in DB banned
            redirect ('/')
        if [username] good
            render ▨ ②
        if [passkey] ≠ passkey
            redirect ('/')
        if [passkey] good
            render ▨ ①
```

```
templates
home.html
    /search
newpage.html
    /result
viewpage.html
    /loadpage
▨ ①
    /editpage
▨ ②
    /
```

('/edit page')

idk how it saves the edits (T_T) maybe in a DB that just replaces TEXT & TABLE each time u open the page but then how u make more boxes?? ¬(-‿-)⌐

redirect ('/')

▨ ①

is the requested extension of base file that contains

- table from DB of unique usernames who have visited the page
- count of unique usernames
- title changer input bar
- top text changer input bar
- table of contents list input(s) bar
- new text input bar
- new table input bar
- commit and exit button

Q w/ no A :

- if DB usually stores one value, how to save edits onto page??
- how to create new/diff. wiki pages? im imagining DB w/ name of page : html pg equivalent, but how to make the new html pages??

- As you grow your sql-fu, always start with Devo Knowledge Base (QAF, n&c) and primary documentation. (...and if none of these have the answer you seek, and you find a SO post that is actually useful, or any other diamond in the rough of the Intertrash, add it to Devo KB)
- Get creative, think boldly, get your hands dirty as you experiment via the sqlite3 shell and see what works/sticks...

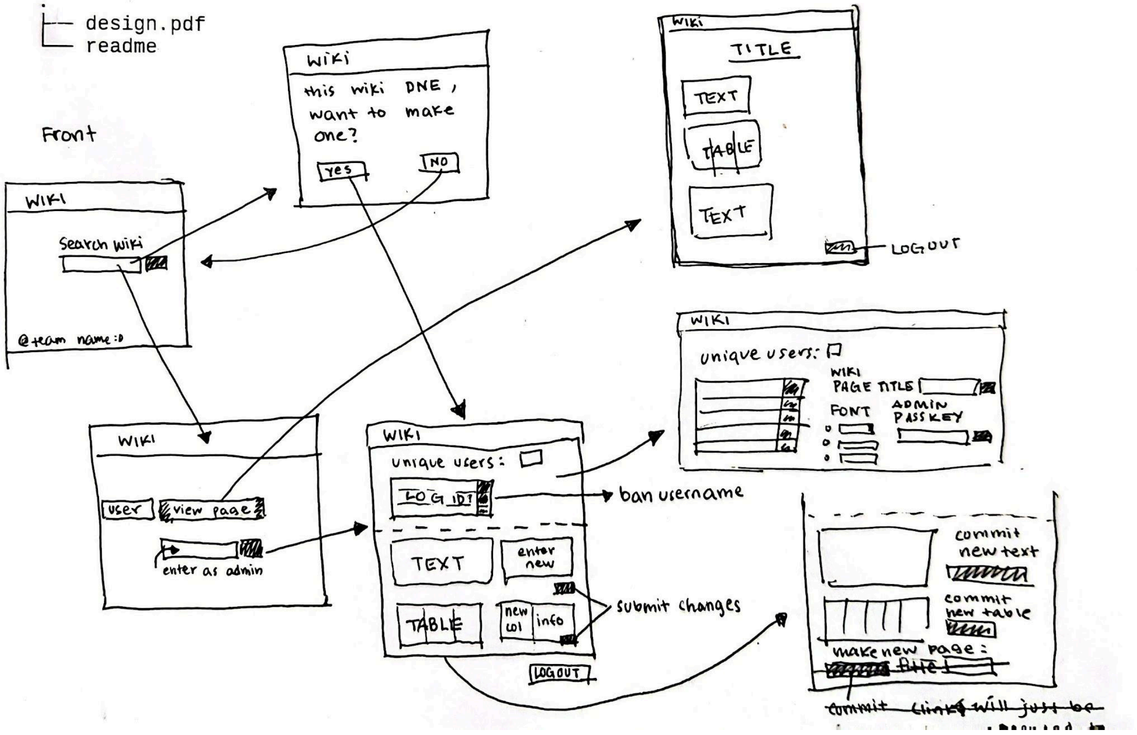
- Multiple Python files should be used, as necessary, for application layer. (*a.k.a.* "middleware" modules, etc.)
- CSS: If your team decides to implement CSS, your site must deliver full functionality, as intended, with CSS is stripped away.
- Use Q&A forum liberally. *"A rising tide lifts all boats."*
- *Reminder:* include heading as comment in all source files.
- Platinum Rule: **THOUST APP SHALT NOT FAIL.**

Your website will incorporate a few distinct components, so it is imperative that your team develop a design and agree upon roles ***before*** you move to implementation.

DELIVERABLES:

- PDF in appropriate location (identical for each teammate)
- Bring hardcopy to class. (3x/team)

```
path/to/myworkshop$ tree p00_wiki
.
├── design.pdf
└── readme
```



[render 1]

Requested extension of base file that contains

- Table from DB of unique usernames who have visited the page
- Count of unique usernames
- Title changer input bar
- Top text changer input bar
- Table of contents list input(s) bar
- New table input bar
- Commit and exit button

Q with no A
If DB usually stores one value, how to save edits onto the page?
How to create new/diff wiki pages? I'm imagining DB w/ name of page: html pg equivalent but how to make the new html ones??