Team Jelly Jam Pancakes: Jacob, Jeremy, Prattay
SoftDev Pd 07
P00 Design Doc
November 2022
Target Ship Date: 2022-11-14

PROGRAM COMPONENTS:

Python File-

- app/__init__.py (Website runner):
    - will run the website, serve and retrieve data using Flask
    - render login.html and home.html
    - Get and post request for user logins

- app/db_builder.py (Database builder)
    - will create/edit the database for story/line storing
    - will create/edit a database to store users and passwords
        - this will be checked to determine if a user has already created/added to a story

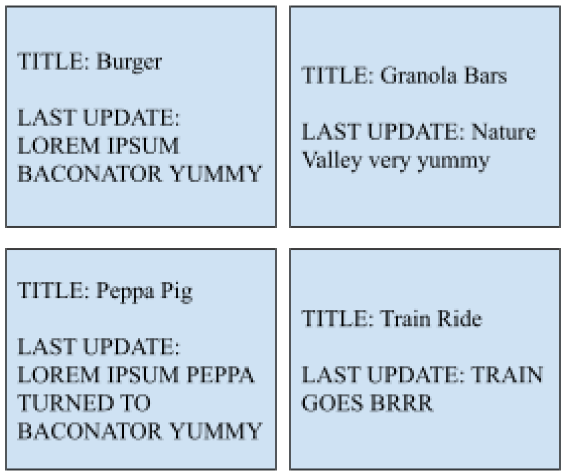
HTML Files-

- /templates/welcome.html
    - Home page for those entering the website
    - Buttons for logins / registering
- /templates/register.html
    - Page for new users to create their accounts
    - Will have UserID form, and PW & retype PW form
    - Will login the user immediately after account creation (?)
- /templates/login.html
    - General login page for registered users
    - Security features TBD
- /templates/feed.html
    - The home page where the user can decide to create a new story, or browse through loaded previews of already created stories
    - Previews may only show the latest update/line added to a story, but will show a user the whole story if the user has already edited the story
        - For a telephone game like story creating process and to follow RULES

- /templates/editing.html

    - Opens an editing window (similar to when you edit a file on github where the user can input new lines to a story

- /templates/stories.html
    - Retrieves data from stories.db and displays the contents of stories onto the webpage

CSS File(s)-
- Display the stories in a social media-like feed, like an Instagram FYP, but instead of posts it is stories and their thumbnails/titles and latest line added.
    - Most upvotes? Most updated/edited? Search feature for genres/titles.
    - 
      TITLE: Burger
LAST UPDATE: LOREM IPSUM BACONATOR YUMMY
TITLE: Granola Bars
LAST UPDATE: Nature Valley very yummy
TITLE: Peppa Pig
LAST UPDATE: LOREM IPSUM PEPPA TURNED TO BACONATOR YUMMY
TITLE: Train Ride
LAST UPDATE: TRAIN GOES BRRR
- Will add more detail in later versions

DB Files-
- Logins.db
    - User ID | Password
    - Stored in text
- Stories.db
    - Story ID | Title | Whole story | User IDs | Latest Line added
    - Stored in text

WEBMAP:

"/"

- Root webpage, displays login.html
- Asks to input username and password, or create a new one

"/home"

- Home page after the user logs in, displays previews of stories using feed.html

"/create"

- Creates a new story, uses edit.html

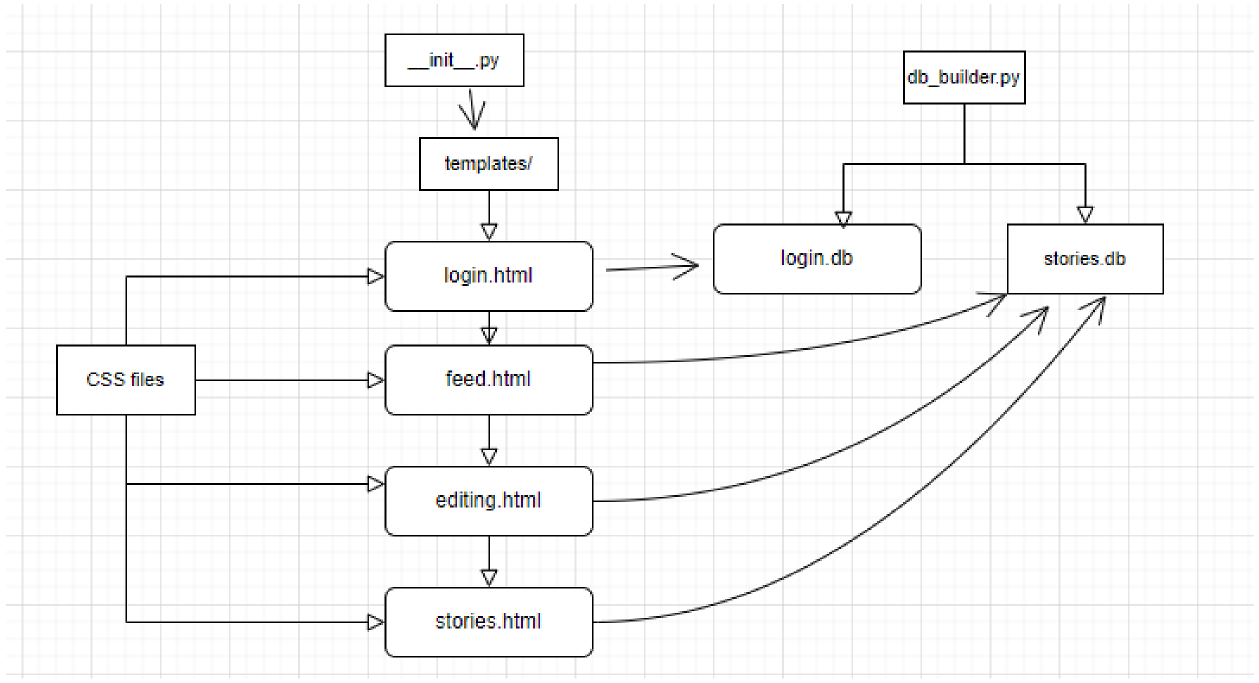
"/<story-title>"

- Displays contents of a story

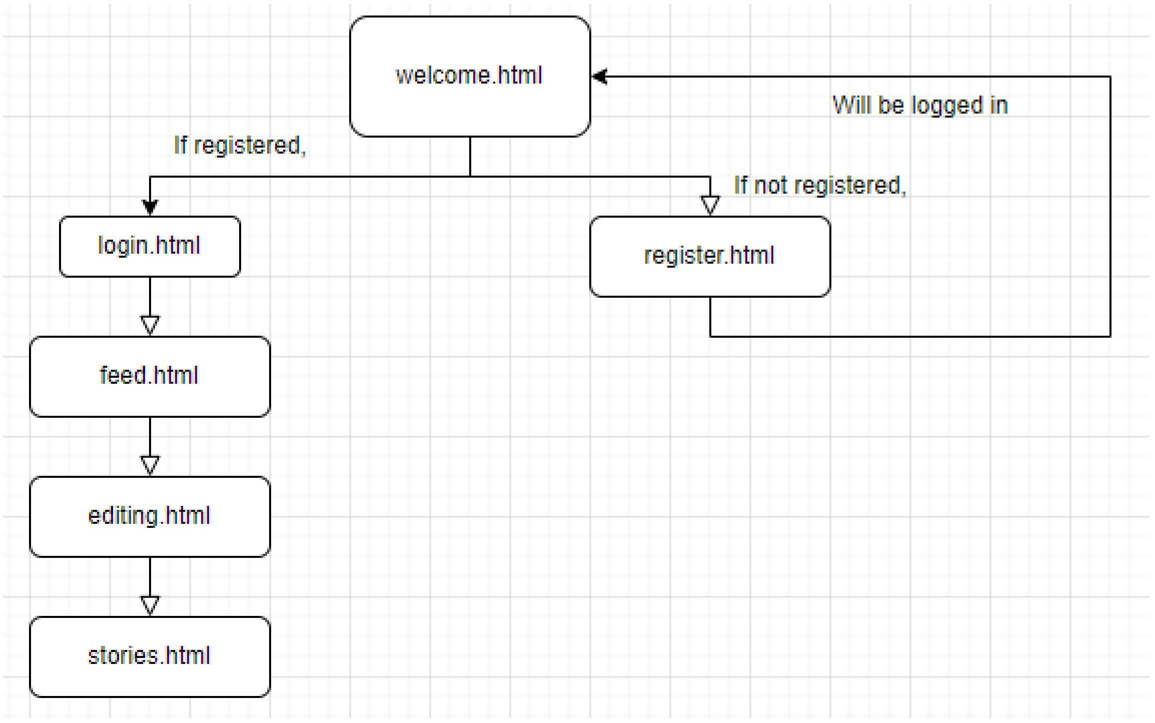
"/<story-title>/edit"

- When a user chooses to edit a specific story that already exists, uses edit.html

COMPONENT MAP:



SITE MAP FOR FRONT END:


welcome.html
Will be logged in
If registered,
If not registered,
login.html
register.html
feed.html
editing.html
stories.html

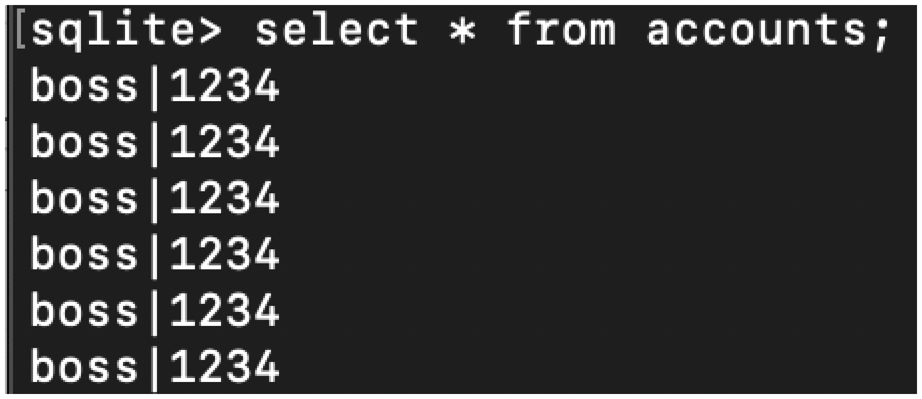
TASK BREAKDOWNS:
Jerm: PM, will work on SQL and db organization (db_builder
Prattay: will work on Flask and __init__
Jacob: will work and create on the HTML and CSS files

Components of Website:

Register:

1. register.html allows new user to input new username
    a. If username is already found in the "accounts" table, then an error message pops up and they pick another name
    b. If username is not taken, then they input a password
2. SQLite stores the new credentials in accounts.table by adding the new entry to the bottom of the "accounts" table
3. Once the user has successfully registered, take them back to the landing page

    Errors:
    - **FIXED**: The SQL database keeps storing the same user/password combination as a new entry each time



```
[sqlite> select * from accounts;
boss|1234
boss|1234
boss|1234
boss|1234
boss|1234
boss|1234
```

Log in:

1. login.html allows user to input a username and password combination
    a. If username not found, then error message pops up saying "username not found"
    b. If username found, then check password
        i. If the corresponding password is found, then log in
        ii. If the corresponding password not found, then error message "incorrect password"
2. Login page should have a registration button for new users