Verit Li, Brian Yang, Justin Mohabir
Team: Uh?
SoftDev
P00: Storytelling Game/Website Design Doc
2022-10-27
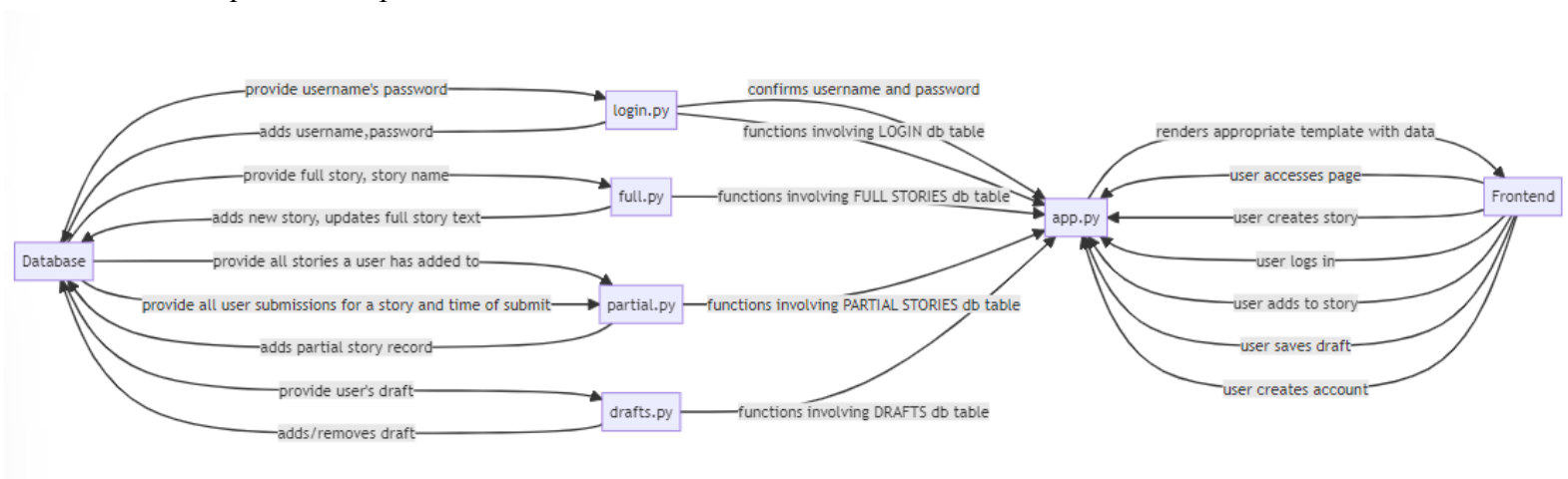Time spent: 5.3 hrs
Target Ship Date: 2022-11-16

Program Components
- __init__.py (To run flask's micro web framework)
  - Renders templates
- login.py - contains functions involving LOGIN table
- story.py - contains functions involving FULL STORIES table and PARTIAL STORIES table
- Html template files (To display html on a webpage)
  - Home page
  - Landing page
  - Login page
  - Full Story page
  - Account creation page
  - New story creation page
  - Edit/Add-to-story page
- Readme (Overview of the Program and how to run it)
- A Sqlite database file for storing info (See Database Organization for more info)
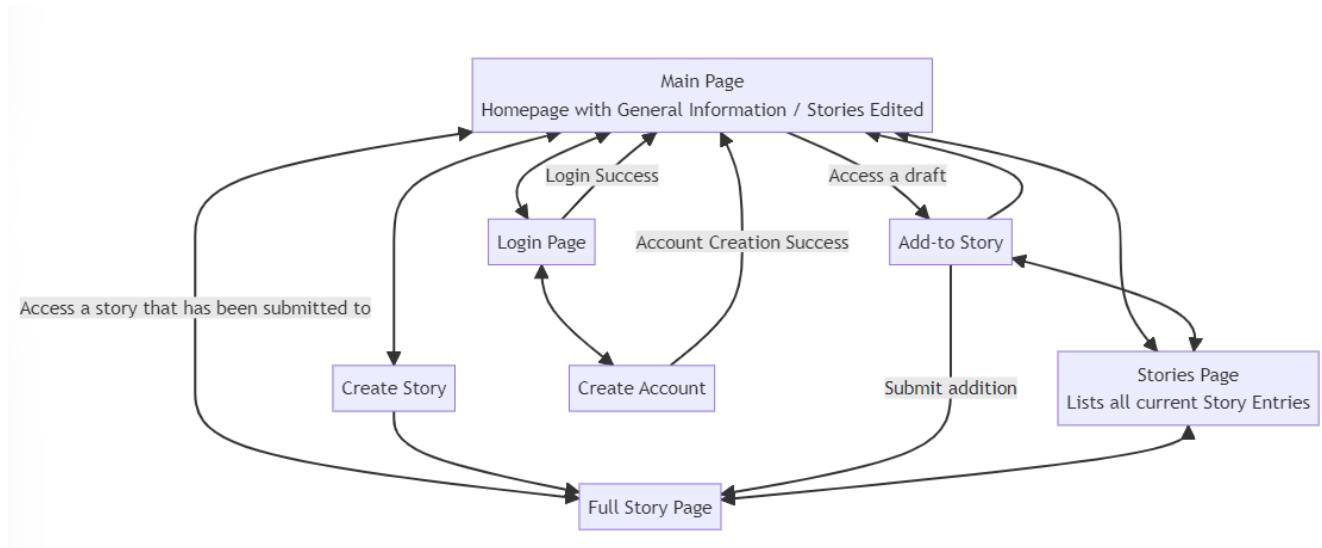- Python file that takes in users changes and creates an html output from it (init.py)

Components Map

Site Map
- Main page (First page to show when opening the webpage)
  - Checks if user is logged-in
    - If not:
      - text is displayed that prompts user to log-in
      - Buttons to Login/Create Account
    - If is:
      - Shows list of all the stories user has created, added to, and has drafts for
      - Buttons to edit a draft, create a story, view all stories in the Story Page
      - Login/Create Account button changed to Logout button
- Login Page
  - Opened by pressing login on Main Page
  - Html form to enter username and password
  - Redirect to Main Page when successfully logged in
  - Button that leads to Create Account Page
- Create Account Page
  - Form to enter a username
    - Checks databases if there is a matching username. If there isn't, prompt the user to create a password
      - If there is, prompt user to enter a different username
    - Checks if password is at least 8 characters long
      - If not, prompt user that password must be at least 8 characters long
    - Save username and password together in LOGIN table in Sqlite database
- Stories Page
  - Displays all Story Entries that you can add to
- Add-to-Story Page
  - Displays story title and id
  - Displays most recent addition to the story
  - Provides a textbox to be submitted with user's addition
    - User is limited to 250 words and is unable to submit if over
    - Dealing with "merge conflict" when submitting:
      - Take the sequence id and story id of the partial story currently being displayed to the user and check if there is a record in PARTIAL STORIES table with the same story id and a sequence id of +1
        - If such record exists, give the user a merge conflict error
        - If such record doesn't exist, create this record using user submission data

- ■ Upon successful submission, brought to appropriate Full Story Page
- ● Create Story Page
  - ○ Requests a story title
  - ○ Textbox for start of story that is to be submitted
    - ■ No word cap
    - ■ Upon submission, brought to appropriate Full Story Page

Database Organization
- ● A FULL STORIES table containing the story's title, full story, most recent contribution, story id
  - ○ Story id generated when new story is created
    - ■ New id is <last_created_story's_id> + 1, starting at 0
- ● A LOGIN table containing usernames, passwords, user id
  - ○ User id generated upon account creation
    - ■ New id is <last_new_user's_id> + 1, starting at 0
- ● A PARTIAL STORIES table containing user id, story id, user's addition, sequence id
  - ○ Sequence id generated when new addition is made/upon creation
    - ■ New id is <last_addition's_sequence_id> + 1, starting at 0

Stretch (*Did not reach*)
- ● A DRAFTS table containing user id, story id, user's draft text, sequence id of partial story that the draft is continuing

Task Breakdown

- Creating html templates (Brian)
- Ability for user add stories (Mohabir)
- Ability for user to add to stories (Mohabir)
- Preventing people who are not logged in from editing (Verit)
- Preventing people who are logged in from editing the same stories again (Verit)
- Allowing users to see all the pages they have edited on the main page (Brian)
- Ability to login/signup (Brian)
- Ability to view full stories (Mohabir)