Reo, Jonathan, Nick, Jason
Software Development - Period 6, Mr. DW

Story Site Design

Project Component + Interactivity

login.html

- Allows user to login to an existing account or create a new one

home.html

- Allows user to create a new story, view new stories that he/she may contribute to, and view stories that he/she contributed to or authored
- Story creation section is an html form with two text fields (one for title, and another for story itself) and a 'create' button; the form calls upon *storyCreate.py* to carry out the appopriate action
- The to-contribute section displays x amount of stories based upon an html form: drop-down button (10, 20, all, etc.)
- The displaying of stories the user has and hasn't contributed/created is dealt with by *homeDisp.py*
- 'Logout' button at the top of the page

storyPost.html

- Used to display stories which the user hasn't added to; shows only the latest update and a textbox with 'add' button
- Title, author, and creation time of story are shown at the top
- The username and upload time of last update are shown next to the text
- Button at the bottom allows user to return to homepage
- If an addition is submitted, the page reloads the story using storyView.html

storyView.html

- Used to display stories which the user has already contributed to; shows the entire story but doesn't allow for further additions
- Title, author, and creation time of story are shown at the top
- The username and upload time for each update are shown next to the corresponding text
- Button at the bottom allows user to return to homepage

app.py

- The all-seeing, all-doing master file
- Calls upon the util python files to carry out the various functions of the site

auth.py

- Takes care of account creation, and logging in/out
- When registering, validates that the username of the new account is unique by going through the *users* table in the database file
-

homeDisp.py

- Shows the titles, text, and timestamp of posts contributed to by the logged in users
- Differentiates what stories have and haven't been contributed to by going through the *stories* table in the database file and then parsing each individual story to see if the user has/hasn't written something

storyCreate.py

- Checks the master story table to see if the proposed story title is unique; creates new story if it is
- Adds story title to the master story table and creates a new table with the story's author, original text, and creation timestamp in the first record

storyDisp.py

- Checks if the user has already commented on the story and displays a different html file depending on the result
- If the user has already commented, returns storyPost.html
- If the user has not commented, returns storyView.html

bd.db

- Contains tables pertaining to user info and story data (refer to database schema below)

User Interface/Experience (UI/UX)

- To access, update, and create stories, an account is required
- The login page is standalone, in that you can't login from the homepage (as you would with a site like Blogspot)
- Once logged in, the user is taken to the homepage
- The homepage contains three main sections: create story, view stories that you haven't added to, and view stories that you created *and/or* added to
- Users can logout from the homepage
- The top of the homepage contains the story creation form
- The second section of the homepage will display randomly selected titles of stories the logged in user has yet to contribute to. The user will be able to control how many titles are displayed.

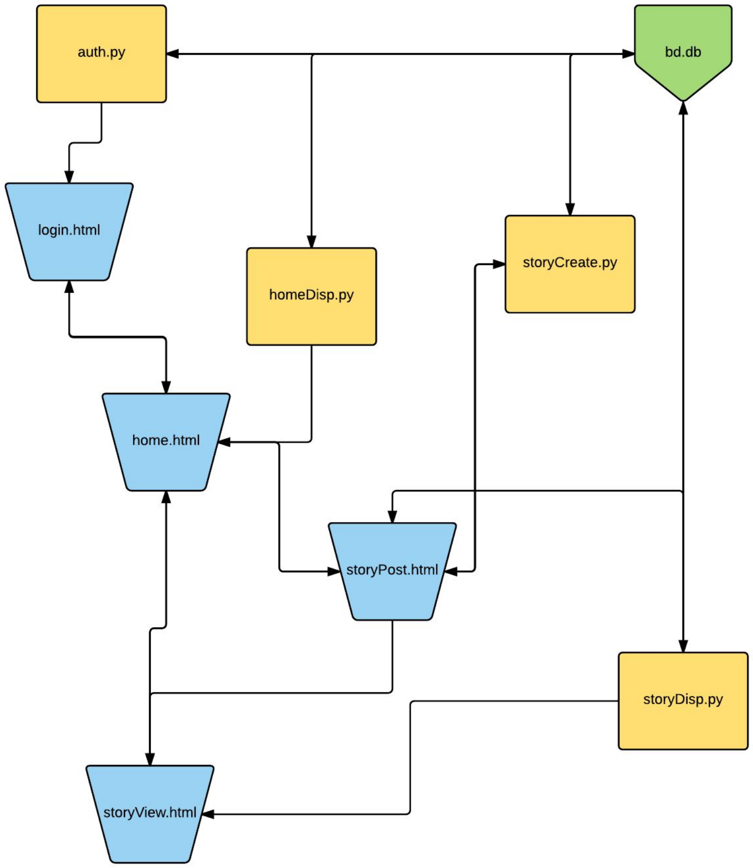
- The bottom of the homepage contains *all* the stories the user has created as well as those he/she has commented on
- In addition to the title, every story displayed includes the author and creation date
- Each story, regardless of whether it has been contributed to, will display the last line that was added to the story (ex. mayo_reo wrote: *“I ate a mayo!”* at October 29, 3:30 pm)
- Though users can’t delete any comments, they can delete stories that they created

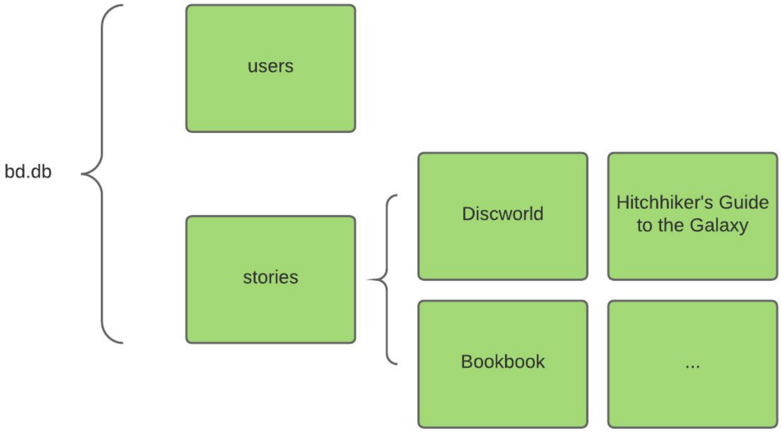
Database Schema

- There will be a single SQLite file that contains all the tables necessary for the site to function properly
- There will be three types of tables: account data, story data, and individual story data
- The *users* table stores each username and its corresponding password (hashed, of course)
- The *stories* table stores each story: has a single column for titles
- Each story will receive its own table; each record has fields for username, text content, and update time
- The first record in an individual story table represents the first update (what was written when the story was created)

Task Distribution

Nick: Project Manager
Jonathan: Posting mechanism
Jason: HTML-Python interaction
Reo: Database management


auth.py
bd.db
login.html
homeDisp.py
storyCreate.py
home.html
storyPost.html
storyDisp.py
storyView.html


bd.db
users
stories
Discworld
Hitchhiker's Guide to the Galaxy
Bookbook
...