BUNNYTRUFFLES // p00 // "BunnyBlog"
AHMED SULTAN, JEFF LIN and LEIA PARK / pd9

PROJECT MANAGER: AHMED SULTAN

ASSIGNMENT: SCENARIO 2 :: WEB LOG HOSTING SITE
- Users will have to register to use the site.
- Logged-in users will be able to
    - Create a new blog
    - Update their blog by adding a new entry
    - View and edit their own past entries
- Not logged-in users will be only able to
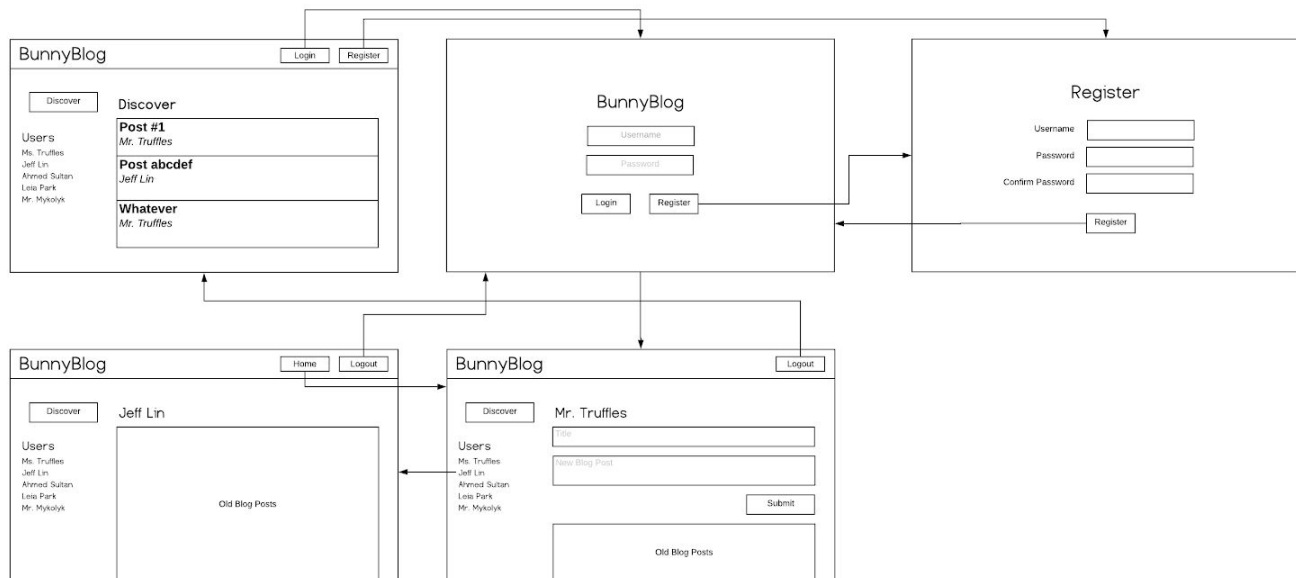   - View the blogs of other users

Front-End Component Map
Sitemap
Database diagram
Database diagram 2 (page 2)

COMPONENTS MAP

FEATURES LIST
- Accounts
    - REGISTRATION (account creation)
        - User should be able to provide a unique username, and password to create an account on the site
        - Creating an account grants the following privileges:
            - Personal blog
            - Write access to personal blog
    - LOGIN (account access)

- User should be able to access their account, provided they have already created one, using the same unique username and password
- Logging in to an account grants the following privileges:
  - Write access to personal blog
- Viewing blogs
  - Regardless of logged-in status, all users of the site will be able to view all blogs on the site
- Writing blogs
  - Only users who have created accounts and logged in will be given their own personal blog
- 'Discover' page/home page
  - When a user visits the home page of the site, they will be met with a variety of blogs from different users
    - Clicking any one of these elements takes the user to that blog's location on the site
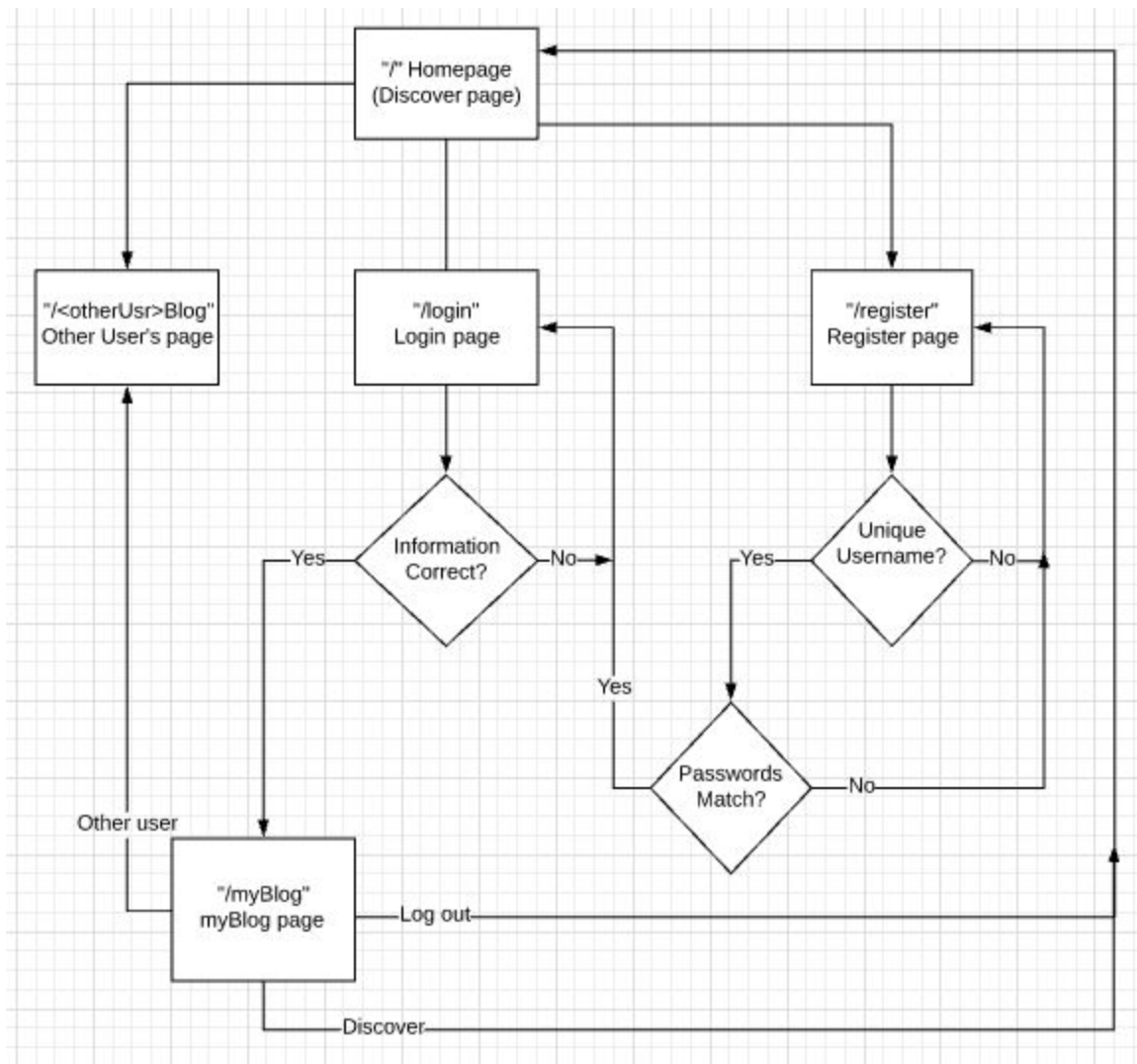
PAGES for SITE
** parent page contains:
- **header "BunnyBlog"**
- **Buttons: login, register, discover**
- Homepage
  - Login button - route to login page
  - Register button - route to register page
  - Discover button - route to discover page
  - View of all the blogs
    - Clicking blog routes to blog's location on site
- Login page
  - Enter username and password
    - IF new user logins THEN "this user does not exist"
    - IF existent user logins THEN routes to own blog
    - Routes to own blog page
- Register page
  - Enter username and password twice
    - IF new user registers THEN routes to own blog
    - IF existent user registers THEN "this username is taken"
    - IF passwords do not match THEN "passwords do not match"
    - Routes to own blog
- myBlog page
  - Username
  - Entries
    - Each has an edit button
    - Entries are pulled from entry database (only displays entries with the same user id)
  - New entry - title box, text box w submit button
- Other user's blog
  - Uneditable entries
    - Entries are pulled from entry database (only displays entries with the same user id)
- Discover Page

◦ Loads ALL entries from database
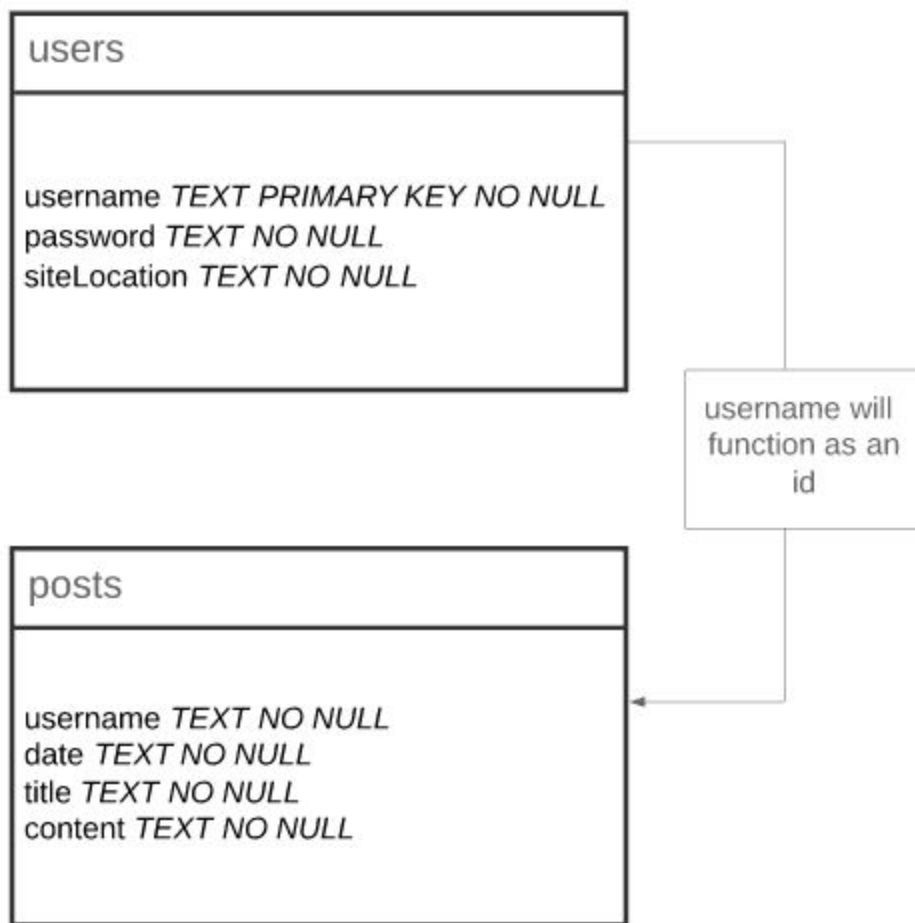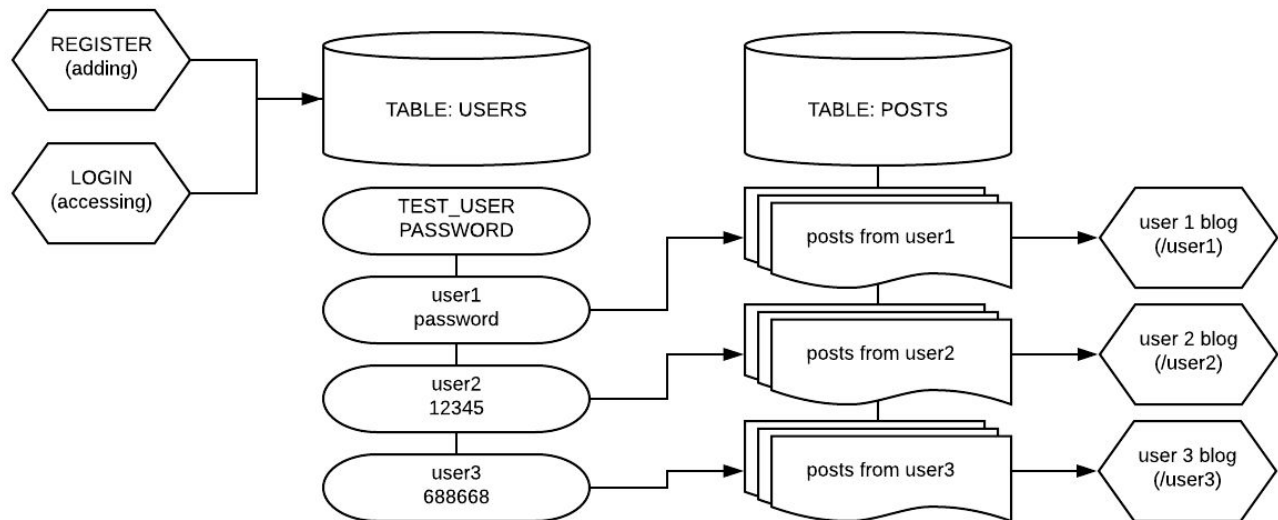
SITE MAP

DATABASE ORGANIZATION
- Table: users
    ◦ username TEXT PRIMARY KEY NO NULL
        ▪ *Note: username will function as an id*
    ◦ password TEXT NO NULL
    ◦ siteLocation TEXT NO NULL
- Table: posts
    ◦ username TEXT NO NULL
    ◦ date TEXT NO NULL
    ◦ title TEXT NO NULL
    ◦ content TEXT NO NULL

DATABASE LAYOUT DIAGRAMS

Flow Diagram

```
REGISTER          ┌──────────────┐      ┌──────────────┐
(adding)          │ TABLE: USERS │      │ TABLE: POSTS │
                  └──────────────┘      └──────────────┘
LOGIN
(accessing)
```

REGISTER (adding)

LOGIN (accessing)

TABLE: USERS

TABLE: POSTS

TEST_USER
PASSWORD

user1
password → posts from user1 → user 1 blog (/user1)

user2
12345 → posts from user2 → user 2 blog (/user2)

user3
688668 → posts from user3 → user 3 blog (/user3)

users

username *TEXT PRIMARY KEY NO NULL*
password *TEXT NO NULL*
siteLocation *TEXT NO NULL*

username will function as an id

posts

username *TEXT NO NULL*
date *TEXT NO NULL*
title *TEXT NO NULL*
content *TEXT NO NULL*

TASKS
- Login & register pages
  - Front-end html appearance
  - Back-end management
    - User database - create, update, store from login & register
    - Checks & matches input from login/register
    - Redirect to myBlog page

Flash errors

TASKS/ASSIGNMENTS
- Login/register pages *(Leia Park)*
  - Front-end HTML appearance and UI
  - Back-end management
    - Users database
      - Creation of database
      - Updating of database
      - Accessing information from database (in order to login and register)
- Writing/editing blog posts *(Ahmed Sultan)*
  - Front-end HTML appearance and UI
  - Back-end management
    - Posts database
      - Creation of database
      - Managing of database
      - Updating of database
- Viewing blog posts/individual pages/Discover page *(Jeff Lin)*
  - Front-end HTML appearance, UI
    - Back-end management
      - Users and posts databases
        - Management and updating of database
      - Filter system
        - Seeing one person's posts VS. discover page (seeing most recent posts? top posts?)
      - Stretch goal: query system
        - Looks for keyword? date?

* While these sections have been specifically assigned, we'll each be helping out with each others sections while actually doing the project *