

Team MS^2: Stefan Tan, Michelle Tang, Matthew Ming, Sajed Nahian
SoftDev1 PD6
P00 -- Da Art of Storytelling
2018-10-15

PROJECT: SCENARIO TWO

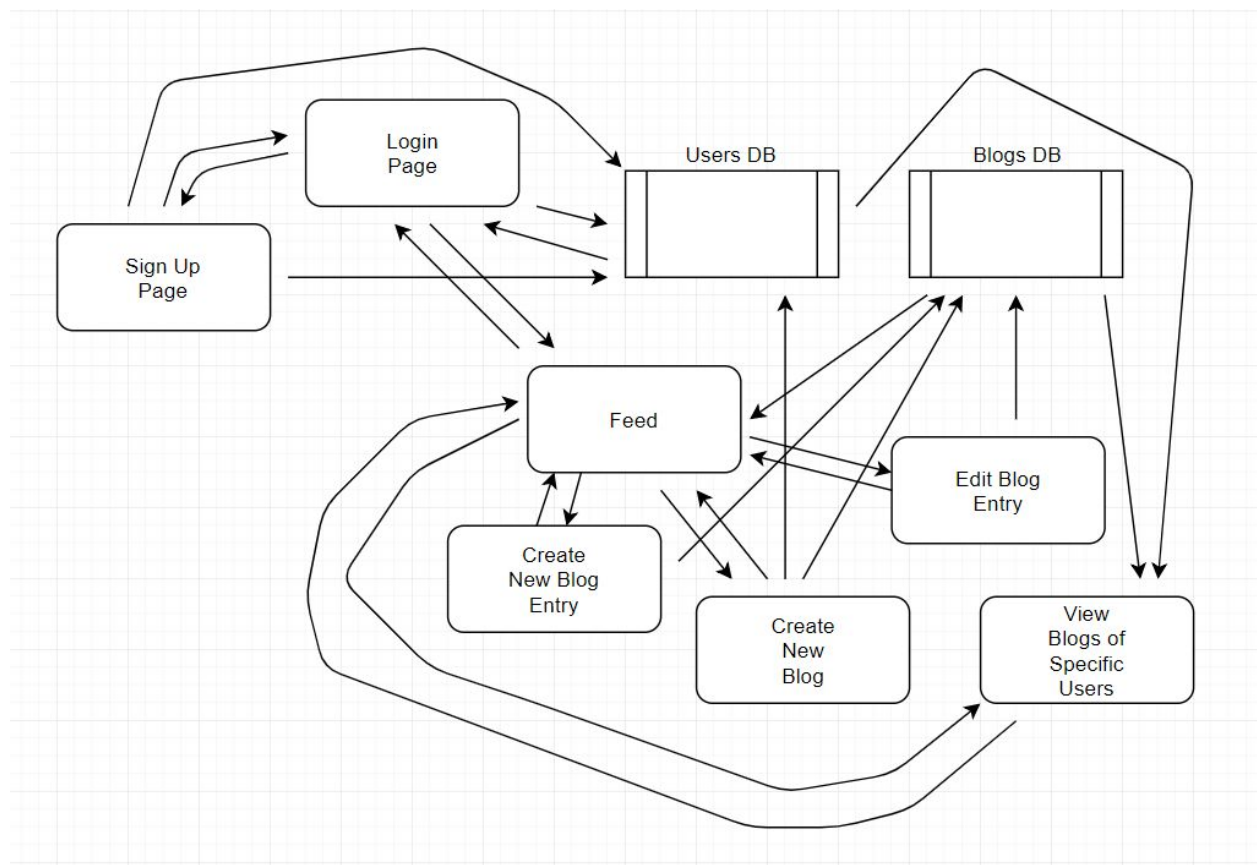
Program Components:

1. Authenticate / Sign In
 - a. Consists of a UNIQUE user/password
 - b. If approved:
 - i. Create a new blog
 - ii. Append previous blogs
 - iii. View past entries
 - iv. Edit past entries
 - v. View blogs of other people (through search)
 - c. If not approved, will be given (flash) notification
 - d. Implemented through Python & Flask.
2. View Information for Database
 - a. User must be logged in
 - b. View Own Blogs
 - i. Read access and ability to edit (SEE Program Components #4)
 - c. View Other User's Blog
 - i. Only able to read
 - d. Will be used through HTML (no CSS), Flask (templates), and Jinja
3. *Search
 - a. Will only be able to search if they are logged in
 - b. Given EITHER title of blog or author
 - c. Will be executed using SQLITE3
4. Add/Modify Information in Database
 - a. User must be logged in
 - b. Modify an entry in an already existent blog
 - c. Create an entry in an already existent blog
 - d. Create a new blog
 - i. Will provide a template to establish uniformity
 1. Template: Title area, text area, author area,
 - e. Will be executed using SQLITE3 (mainly)

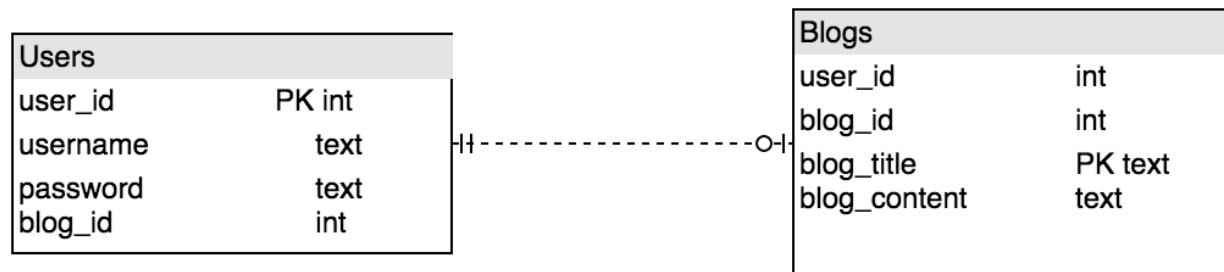
Relation of Components:

Users are first asked to login. If they do not have an account, they can create one which will modify the Users DB. Input from the login page will be verified against information from the Users DB and if correct, will redirect to the feed page. The feed page will take information from the Blogs DB database and display random content from it. From the feed page users can create a new blog entry, create an entirely new blog, view blogs of specific users or edit

previous blog entries. Creating a new blog entry will modify the Blogs DB. Creating a new blog will update both the Users DB and the Blogs DB. Viewing blogs of specific users will take info from both the Users DB and the Blogs Db. Finally editing a blog entry will modify the Blogs DB.



Database Schema:



- Will consist of one database with two tables:
- MATCHES: user_id, blog_id

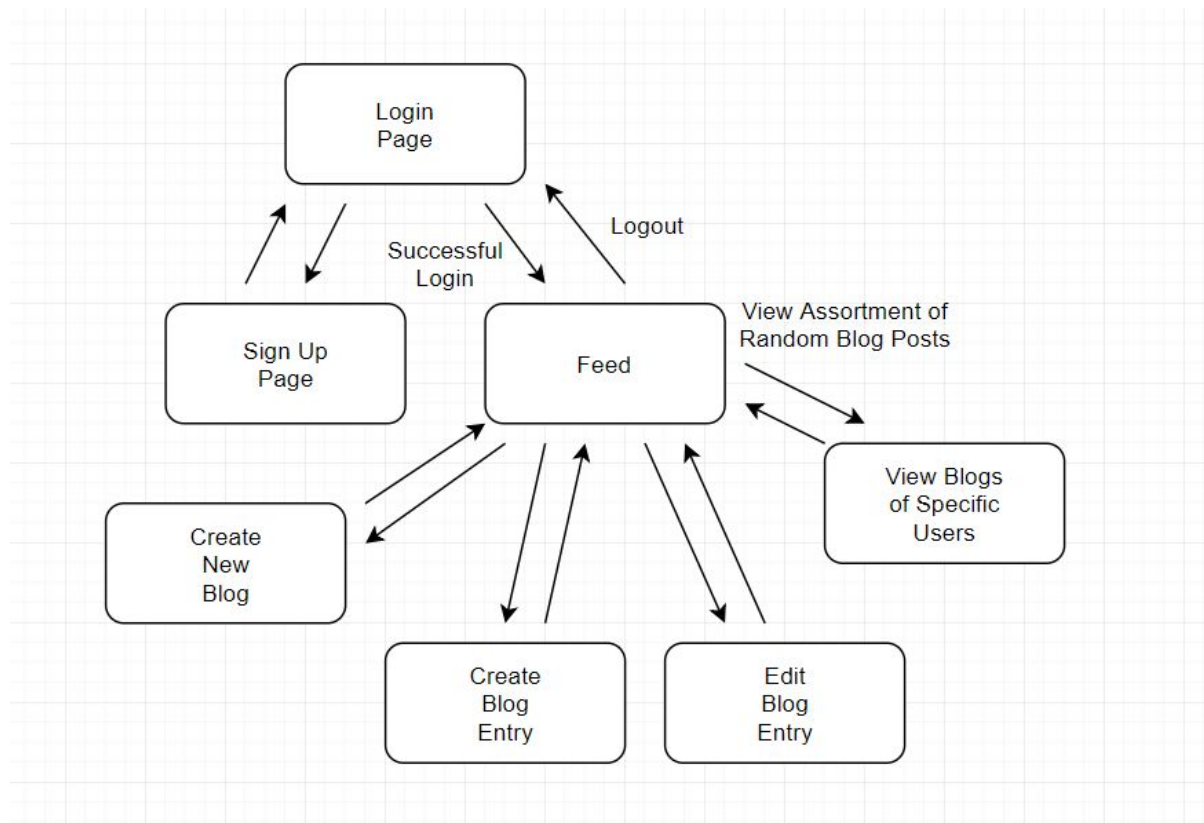
USERS :

- Two TEXT fields: username and password.
- User_id is unique for each user.
- Blog_id is unique for each blog created.
 - The user id will be a primary key
- Will update whenever a new user is created
- Headers: user_id | username | password | blog_id
- An additional TEXT field containing contributed entries*

BLOGS:

- Two TEXT fields: one for the title of the blog and another one for the text content within the blog.
 - The title will be a primary key.
 - Will update whenever a verified user adds an entry or creates a new blog
- An additional TEXT field containing the author

Site Map:



- **Welcome/Login page**
 - All users are greeted to the login page. Registered users are permitted to login while first time users can sign up their new accounts in the sign up page.
- **Sign up page**
 - Users must create a username and a password which will be saved in the database. The username must be unique.
 - After signing up, the user is sent back to the login page.

Login Page

Username

Password

Sign up

Username

Password

Confirm Password

- **Feed**

- Users can access their feed, which gives them access to search for other blogs or create new ones.
- In the feed, there will be a list of the blogs they have created. Within each blog, there will be an option to add an entry.
- In the feed, there will also be the option to create a new blog
- There will also be a search bar that will take an author/blog and will lead to the search page.

- **Search Page**

- Users are able to browse through other people's blogs.
- Results will only be displayed if there is a blog name match or an author name match.
- Each blog entry can be clicked on which will lead to a blog page.

- **Blogs:**

- Contains past entries from blog
- Able to be edited
- Contains link to logout if user is already logged in
- Has option to add more entries

Task Breakdown:

- Jinja templating - Matthew Ming
 - Sign In
 - Sign Up
 - Search
 - View Blog Entry
 - Modify Blog Entry
- Database creation and code for modification, creation or retrieval of entries. (For both tables within the database) - Michelle Tang

- Functions needed:
 - addUserToDatabase (username, password)
 - checkUserInDatabase(username, password)
 - getBlog(blog_title)
 - modifyBlog(blog_title, blog_content)
 - createBlog(blog_title, blog_content)
- Routing and sign in code - Sajed Nahian
 - Connection to Jinja templates
 - Connection to Database functions
- Project Manager / Tester - Stefan Tan
 - Make sure idea is accurately reflected in final website
 - Make sure all components of project are done in timely manner
 - Test various edge cases on the website to ensure proper implementation
 - Maintain effective communication with team members and Mr. Brown