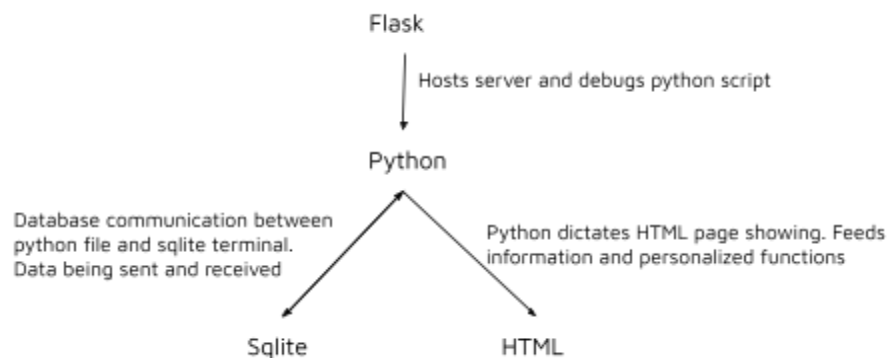


Design Components

- Flask - Hosts website and communicates with front and back end
- Python - Manipulates data received and controls log in/out function.
- Sqlite - Creates and maintains database. Will track user information and story board edits
- HTML (and CSS?) - Specifies homepage, login page, storyboard edit page, storyboard viewing page. We can use a CSS framework (like Bulma) to easily make things look nice

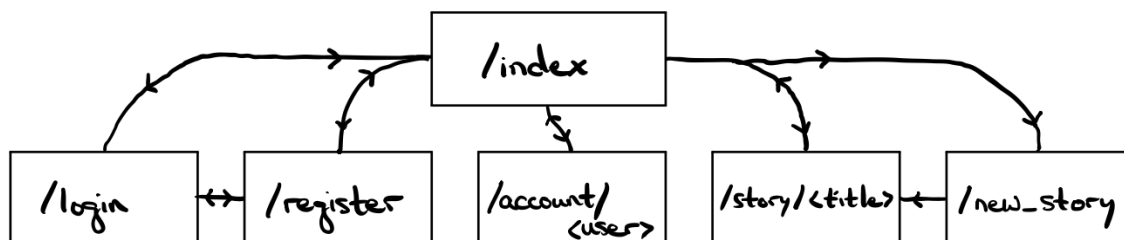
Component Map:



Database Organization

- Users: username (key, unique), email (unique), password
- Stories: title (key, unique), first_block_id, last_block_id
- Blocks: block_id (key, unique), creator_id, block_text, next_block

Sitemap



/index

Welcome!
Please [login](#) to continue.
New user? [Register](#)

/login

Log In
Username:
Password:
New user? [Register](#)

/register

Register
Username:
Password:
Confirm:
Existing user? [Login](#)

/index (logged in)

Hello, [username]!

+ New Story
Stories You've Contributed To
Alfred in Wonderland [first 150 chars...]
Another Story

Add to a Story
3 Little Devos [latest sec.]
Another...

account settings

/story/<title> (contrib.)

Alfred in Wonderland
© Once upon a time...
© Some other part...

/story/<title> (not contrib.)

3 Little Devos
© Not by the hairs on my chinny chin chon!

Add to the story...

Task Breakdown

- Setup basic Flask routes
 - Index – Yusuf
 - Login – Yusuf
 - Register – Yusuf
 - Account/<user> – Chris
 - Edit account details – Chris
 - Story/<title> – Chris
 - New_story – Chris
- Create HTML templates for pages
 - Index (not logged in) – Naomi
 - Index (logged in) – Naomi

- Login - **Deven**
- Register- **Deven**
- Account details
- Story (for contributors)- **Deven**
- Story (for people who haven't contributed yet) – **Naomi**
 - Only last block
- New story – **Naomi**
- Add CSS – **Chris**
- Setup SQLite database
 - Users - **Deven**
 - Stories- **Deven**
 - Blocks – **Yusuf**