

**Kazimierz Major — Andrew Juang, Eric Guo, Noakai Aronesty, Qina Liu**  
**Softdev**

**P00: Move Slowly and Fix Things**  
**2021-10-27**

### Program Components

- Login for users
- Dashboard to manage own blogs (publish/delete/edit blogs)
- Backend database (table for users, table for entries/blogs)
- Newest blogs (title, author) on index/home page
- User profiles with their blogs (same template as dashboard or home page)
- Render\_template to create each blog page, each blog has the same template
- HTML templates with Jinja2 for dynamic abilities

### Component Relationships

- The login page will be for users to log in, and after you successfully log in, you can then access the website: view blogs and publish, edit or delete blogs of your own.
- The home page will list all blogs, and ability to go to different users' dashboards (including your own)
- The backend data will store all of the information in the database to be retrieved and displayed on various pages, in the case of blogs, or to be retrieved and checked in the case of users. SQL and sqlite queries to facilitate this.
- Flask used to interact with front end and template Jinja2 areas dynamically
- The blogs will be generated via the same template to keep the same design across all blogs.
- Clicking on a hyperlink to a blog will render the blog with that template

### Templates

- Login / error page / register
- Home / Your Dashboard / Other Dashboards
- Edit / Add Entry / Create Blog
- Viewing Blogs

### Database

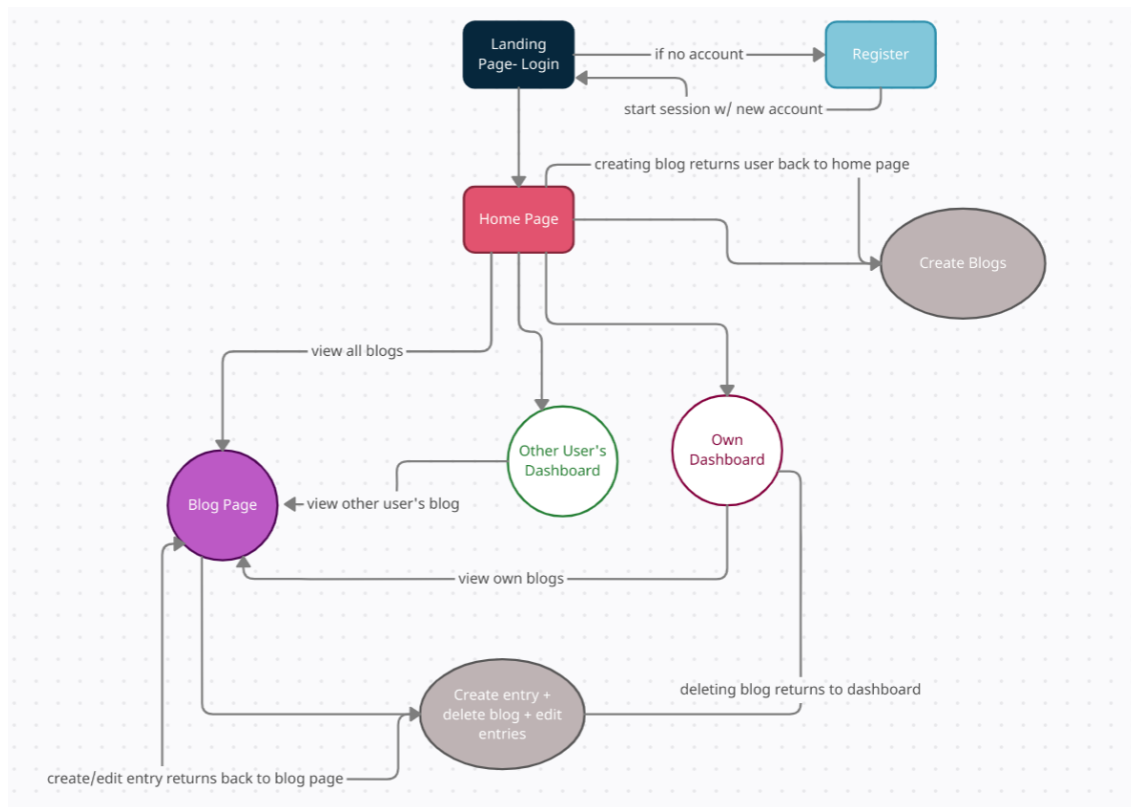
- Username / Password

Username	Password
Bob	turtles123
Bill	epicg4m3rZ!
Bobby	welovefortnite

- Username is unique and case sensitive
- Password is not unique and case sensitive
- Username/Blog
  - Blog Titles can't repeat
  - Entry names can repeat for the same blog

Username of author	Blog Name	Entry Name	Content
Bob	We are farmers	Hi There	I love farming, you love farming, ...
Bill	Hitting Silver	Day 1	It was very difficult, but I did it. I ...
Bill	Creating a team	Yeehaw	Creating a team is ...
Bobby	Weeeeeeee Bingle	First Victory Royale	Fortnite is hard, but we love fortnite, and we just ...

## Sitemap



## Rendered Pages:

/ (when not logged in)	/register
<p>Welcome, login to continue!</p> <p>Username Input Field</p> <p>Password Input Field</p> <p>Corresponding Error Messages</p>	<p>Welcome, register to continue!</p> <p>Username Input Field</p> <p>Password Input Field</p> <p>Confirm Password</p> <p>Corresponding Error Messages</p>
/ (when logged in)	/dashboard/[user]
<p>My dashboard + logout</p> <p>Table of [Blogs] By [Authors]</p>	<p>Homepage</p> <p>logout</p> <p>Table of Blogs</p>
/display/[blogtitle]	/create, /edit
<p>Homepage</p> <p>My dashboard + logout</p> <p>[BLOG TITLE]</p> <p>[Entry name]</p> <p>[text]</p>	<p>Save/discard changes</p> <p>logout</p> <p>[BLOG TITLE]</p> <p>[Blog entry name]</p> <p>[text]</p> <p>UPDATE</p>

## Task Breakdown

- Setup basic app + routes
  - Authentication: Login / Register (**Noakai, Qina**)
  - Index/home page (**Andrew**)
  - Dashboards (**Andrew, Qina**)
  - Edit / Writing / Update blogs(**Andrew, Qina, Eric, Noakai**)
- HTML page templates + CSS

- CSS (**Noakai**)
  - Login / Register / Error template (**Qina**)
  - Index/home/dashboard page template (**Andrew**)
  - Viewing blogs (**Andrew, Qina, Noakai, Eric**)
  - Edit / Writing blogs template (**Eric**)
  - Update blogs template (**Eric**)
- SQLite Database
  - Users (**Eric, Qina, Noakai**)
  - Blogs (**Noakai, Qina, Eric**)