

Indigo (Stanley Hoo, Jacob Lukose, Naomi Lai, Colyi Chen)

SoftDev

P00: Move Slowly and Fix Things

TARGET SHIP DATE: 11/4/24

## Scenario 1

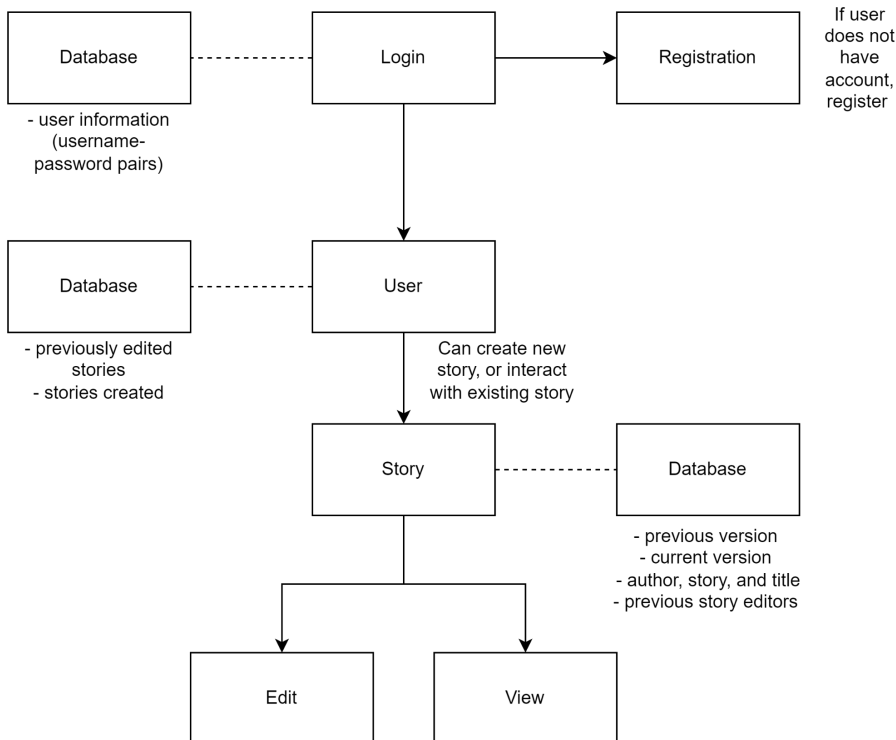
### Components

- User accounts
  - login/logout
- Route for users to create stories and add to our page.
- Route for users to add to stories
- Route to view stories once you've edited it
- Route to view completed stories
- SQLite Database with a unique id for each story and user id for all edits
- A page where those not signed up can sign up, and if already signed up, you can log in.
- A homepage that everyone has access to when logged in.
- Templates for:
  - Homepage
  - Contributions page
  - Stories in progress page
  - Completed stories page
  - Edit Story Page
  - Create story page
  - Signup page

### Component Relations

- User accounts are linked to specific data in the database
- Routes for users to create stories – users must be logged in to create stories
- Route for users to add to stories – only users can add to stories. We will also check the list of users who already edited the story, to make sure that they do not edit it again. Stories must also not be completed, once they are completed by an editor, it will permanently be locked and viewable to everyone
- Route for users to view – displays full story if logged in and contributed to story, otherwise last update. Different views per webpage (contributions will only shows stories you've contributed to, completed stories will only show completed stories, in progress will only shows stories in progress, home will show all)
- SQLite database will store all of the story information, past edit history, story IDs, and user info. Separate ones for users and story attributes/ history.
- Route to view story will retrieve info from databases
- Sign up/login page will send and retrieve data to databases
- Homepage will retrieve data from the databases (stories)
- Templates allow us to use the same html code for multiple pages easily while changing some variables

## Component Map:



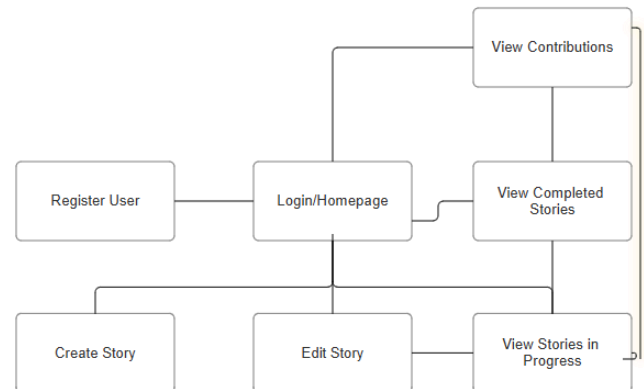
All users can view the page, whether or not they are logged in. If not logged in, they will only be able to view the last edit of the stories or completed stories, if they try to edit will prompt them to login. If they are logged in, will have 4 separate pages: one that has full stories that they've already edited, one that only has stories still in progress, one that has stories the user has contributed to, and one that has all stories. All of this will be managed by sending and retrieving data from the database.

## Database Organization:

- 3 tables:
  - users
    - Saves username, all usernames must be unique
    - Saves hashed password for security
    - We will assign a unique ID to each user to make it easier to access later
  - stories
    - Here will we track the story info
    - Each story will receive a unique ID (AUTO INCREMENT)
    - Store the story title
    - Store the status of the story (completed/in progress)
  - story\_edits
    - Here we will store the reference info for each edit, including unique edit ID (AUTO INCREMENT so it is chronological), story ID, user\_id of editor, and the actual edit

## Site Map

- /home: Homepage where user can log in with their existing username+password, login container in top right
- /register: creates username password; username cannot already exist
- /create\_story: users can create a title, edit story text, and submit the story
- /edit: edit existing story
- /in-progress: display stories in progress
- /completed: display completed stories
- /contributions: display stories the user has contributed to
- Note: No login route needed, we will create a login container on the homepage where users can login



## Tasks:

Storing and retrieving info from the databases, linking everything together, css/styling, user logins, some html templates – Stanley

Css/styling, html pages – Jacob/Naomi

Nothing - Colyi (disappeared)