

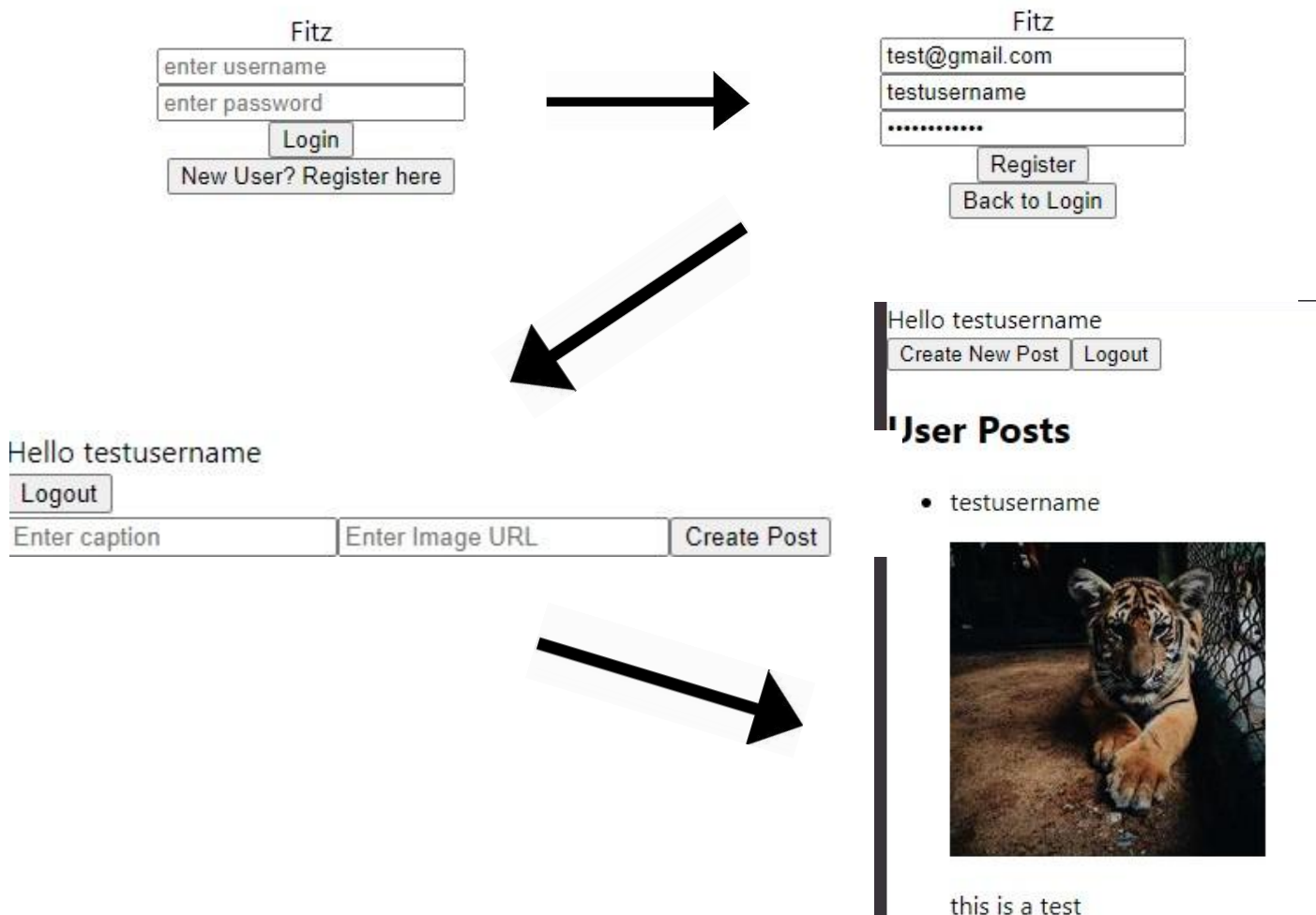
Fitz: Phase 1 Report

Creators: Wesley Vance Blair, Katarina Marsteller

Implementations

We were able to implement one significant area of our Phase 1 plan. That being the user account creation functionality. A user is able to so far sign up for an account - using an email and password - and sign into the website. Upon signing in, a successful login message appears on the screen.

As can be seen in the images below (pictures were cropped for visual purposes), when taken to our home page, the user has the option to login with their username and password, or register with their email, if they don't already have an account. After logging in, the user has the option to create a new post or logout. Once a new post is created, it will appear on their profile. This will allow them to choose which of their posts to enter in the contest (Phase 2). Upon logging out, the user is taken to the initial home page. As can be seen, the application still lacks visual appeal (which will be utilized through css before final project submission).



The backend for the account functionality is split into three main categories: models, services, and API. There is a model for both the posts and the users which defines the schema for the database. The users include an email (in correct email format), username, and password. Posts will entail the creator's username, a caption, an image (URL), and a comment section. The services section is responsible for operating on the database itself. There you will find creation, deletion, and display of user accounts and posts. The API allows for the frontend and backend to communicate. This allows the user to indirectly modify data in the database and the database to send out the data.

Unfortunately, we had to reconfigure our original Phase 1 plan. Upon starting development of the voting system, we realized that the Contest functionality had to be created in order to properly configure the voting system into it. Therefore, we chose to start with the Contest functionality first.

Despite the setbacks, we were also able to organize the files on our GitHub repository to accurately reflect both sections of the project. Backend components are held within the *Backend* folder while frontend components are held in the *Frontend* folder. With this organization in place it will be easy to distinguish files and easily create necessary paths.

As we began to work on our project, we realized some features that we left out of our plan that we believed should be implemented, such as the ability to logout and the specifics of what the display would look like upon logging in. Additionally, not all components mentioned in the original plan will be actually a part of the final product, as new ones will be added.