Milestone 4

Revised Features List:

- Home Page
    - First page the user sees
    - Greets them when software is opened
    - Clicking on the phrase “Enter Gallery” allows user to enter gallery
- Gallery View:
    - Second page the user sees
    - Displays six preset images the user can select for their starting image
    - Features an upload button the user can use to upload their own starting image
- Upload Button:
    - Ability to upload images in order to allow the users to make mosaics of their own images.
- Mosaic View:
    - Third page the user sees
    - Displays the mosaic in the middle of the page
    - Displays navigation bar at the top of the page
- Mosaic:
    - A large composite of smaller images

- Image Interaction Menu:
    - Displays the following options:
        - Info -> Information about the piece and the artist displayed in a textbox
        - Remake -> Replace current main image with a newly selected image (this is image would be a “template” for the composite)
- Navigation Bar:
    - Displays the following options:
        - Home -> Redirects to homepage
        - Download -> Popups user’s file system so they can download the mosaic
- Zoom Button:
    - Allow user to zoom in on the mosaic

Individual Contributions

Sophie:

- Working on making menu appear when image clicked
- Then I'll work on getting data to transfer to menu

https://github.com/CSCI-3308-CU-Boulder/209-5-FA20/blob/master/All%20project%20code-components/UI/views/mosaic_view.html

Elizabeth:

- Fixing responsiveness of web page so they can adjust to different browser sizes
- Looking into html interaction with flask

https://github.com/CSCI-3308-CU-Boulder/209-5-FA20/commit/2bb82526368af165bc7ae7b31b120780d661f225
https://github.com/CSCI-3308-CU-Boulder/209-5-FA20/blob/master/All%20project%20code-components/UI/views/home_page.html

Tim:

- Zoom feature

https://github.com/CSCI-3308-CU-Boulder/209-5-FA20/tree/master/All%20project%20code-components/Zoom%20Feature

Kevin:

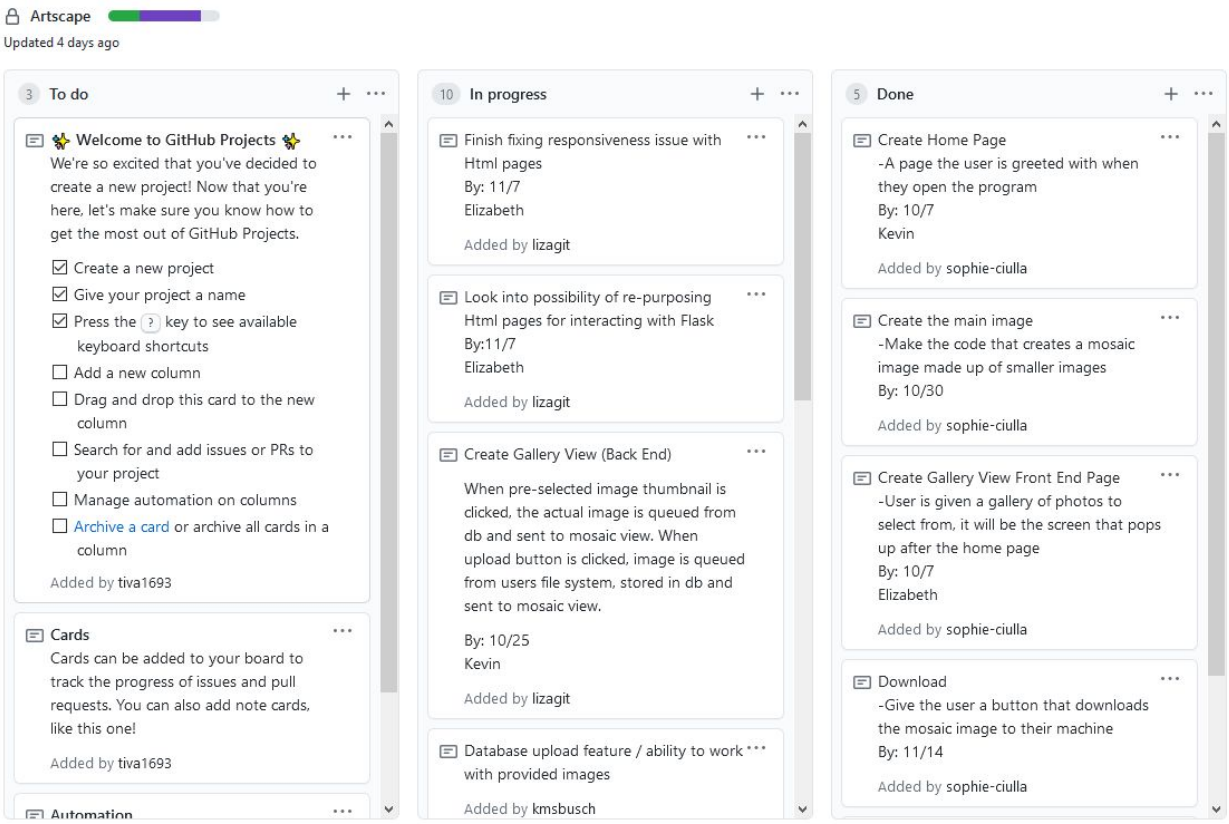
- Making image map for mosaic
- Integrating mosaic function to flask
- Transmitting image file from gallery to mosaic page
- Integrating reshuffle functionality into mosaic page
- FInding use for database / managing database organization

http://kmsbusch.pythonanywhere.com/

(not on Git but there is a working site that demonstrates the functionality of the application)

Sky:

- Main algorithm of the mosaic part
- Export metadata to collaborate with the zoom and re-shuffle part.



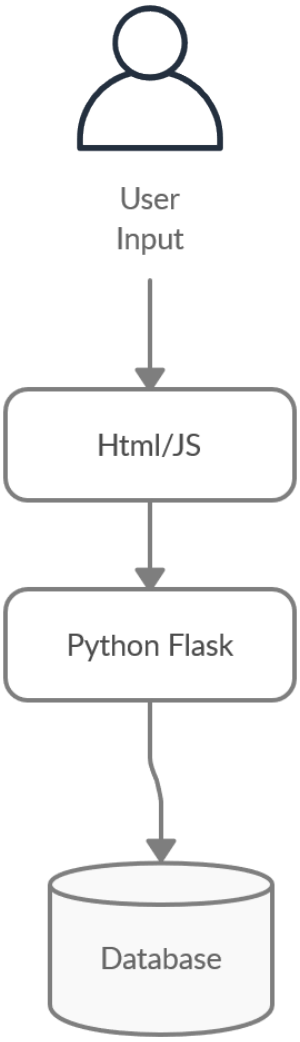
Commit Links

Sophie's:

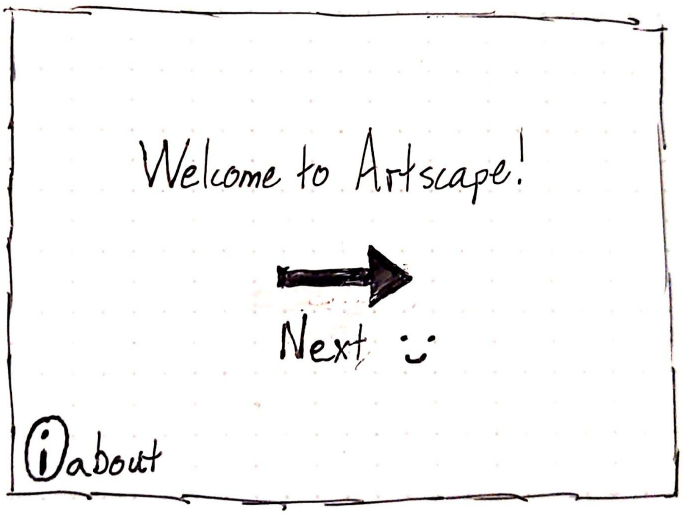
https://github.com/CSCI-3308-CU-Boulder/209-5-FA20/blob/master/All%20project%20code-components/UI/views/mosaic_view.html

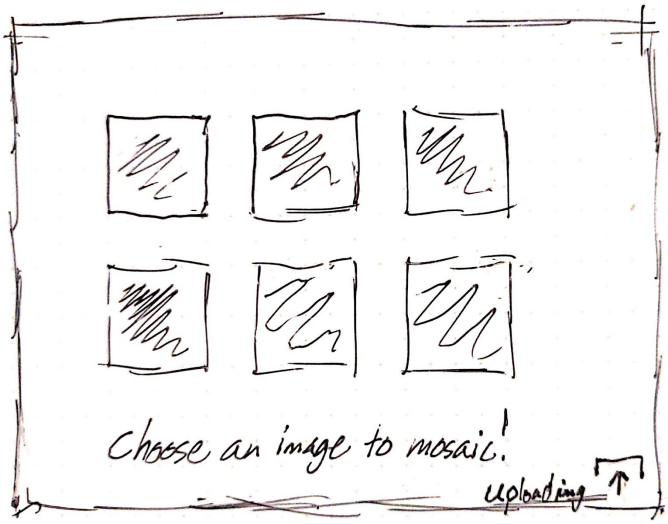
Architecture Diagram

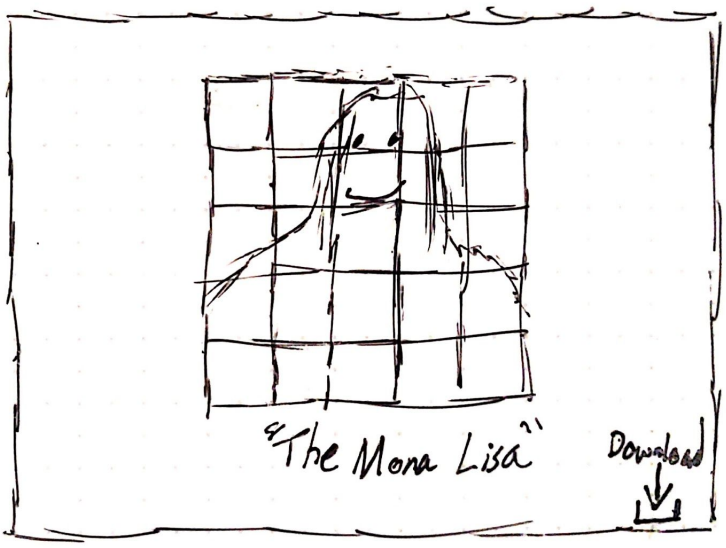
*Html/JS interacts with Flask via REST API

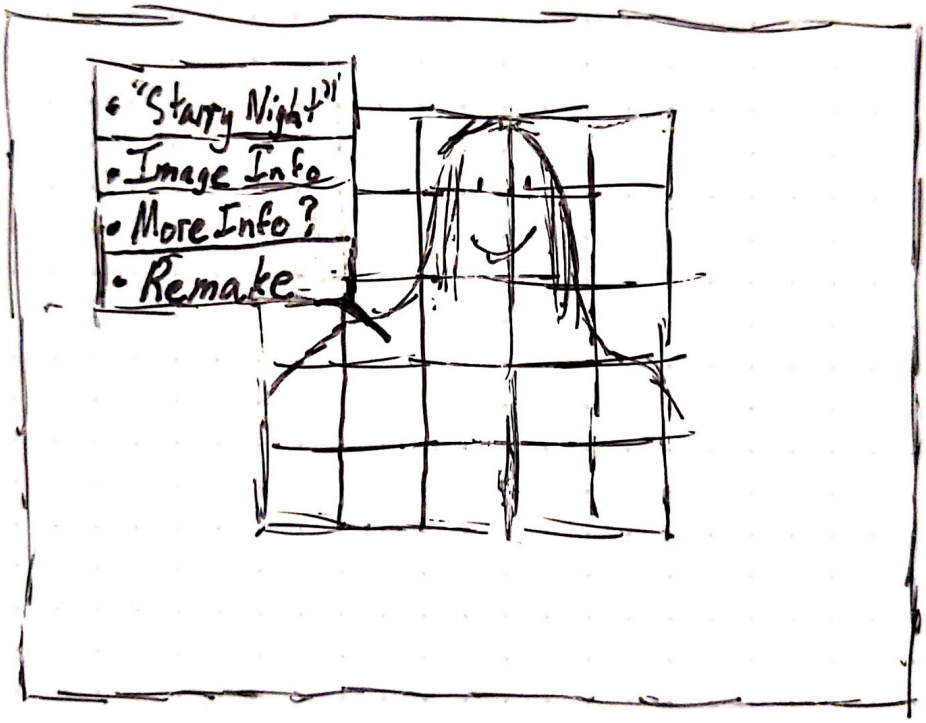


Front End Design

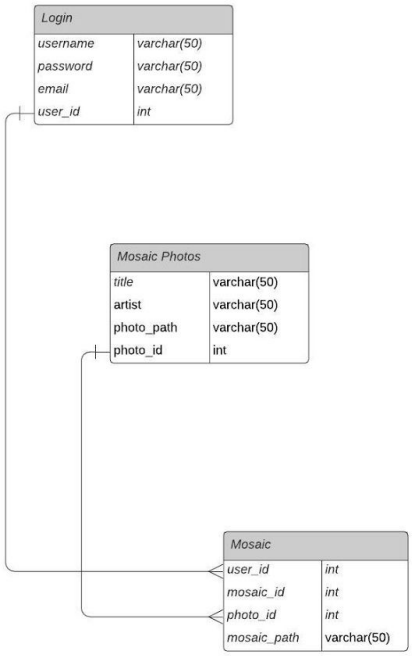

Welcome to Artscape!
Next :)
about


Choose an image to mosaic!
uploading





Database Design



We plan on storing photos for the mosaic, the artist, title and dimensions of the photos, the mosaic itself and login information in our database. The DBMS technology we plan on using is PostgreSQL.

Backup

Our backup for the “Remake” feature is applying a filter to the current image (i.e. black and white, sepia etc.)

Our backup for the “Info” feature is populating the info box with image-independent information such as the size/dimension and number of images used for the mosaic

Challenges

1. To generate the mosaic, a grid is applied to the main image and filled with smaller images. In order for the individual images to be clicked on to trigger the Image Interaction Menu, they need to be recognized separately in the mosaic. This ability will likely require a reapplication of the grid which contains a method of tagging the images that belong in each space of said grid.
2. Generating the image-dependent information displayed in the popup window triggered by selecting the “Info” option. By image-dependent, this refers to information such as the photo name, artist and maybe a blurb about image.
3. The “Remake” option that takes the smaller image that was selected, converts it into the new main image that the mosaic is generated from.
4. Passing information from the database to the backend. Examples of this include images for the mosaic and populating the info popup from selecting the “Info” option from the Image Interaction Menu.