Jay Hou, Liam Mullen, Helen Luo, Alex Yan

Motivation

- Generate high-res, novel images for the end user without the need for learning any knowledge of photo-editing skills; there is quite a learning curve associated with using image manipulation softwares.
- Potential techniques include mosaic generation, edge detection, greyscale, inversion, convolution, Gaussian blur.

Tech Stack

- Backend/Server: **Django** (Python) for server simplicity, scalability, and default securities such as SQL protection and user authentication.
- Frontend: **React.js**
- Database: **PostgreSQL** to manage user accounts and their previously processed images.

Milestone Goals:

- Sprint 1:
    - Basis of web server functionality, interact with Database to transfer and store image data.
    - Develop an organizational framework for the development process that emphasizes modularity and scalability of our product.
- Sprint 2:
    - Propagate the database with stock royalty free images to generate new images from.
    - Implement the core features of our web application with the image mosaic being the focus.
- Sprint 3:
    - Add additional features such as sharing the images to other social media platforms. The most recent images are up for display to other users. Show image creation history from the user.
- Sprint 4:
    - Finish up with a more dynamic user interface.

End Goals:

- Have an edited image produced for the user after the user uploads or enter keywords of what they want edited.
- Store images created for the user in a database for the user to look back at upon request using PostgreSQL.
- Allow users to download images generated so they can use them for their own purposes (ex. Posting on social media)
- Allow for users to preview and view the latest past ten images in the home page.