Picopy

Jett Crowson, Justin Floyd, Benton Haensel, Jacob Dewild, Sarah Bordiuk, Will Huynh

Summary

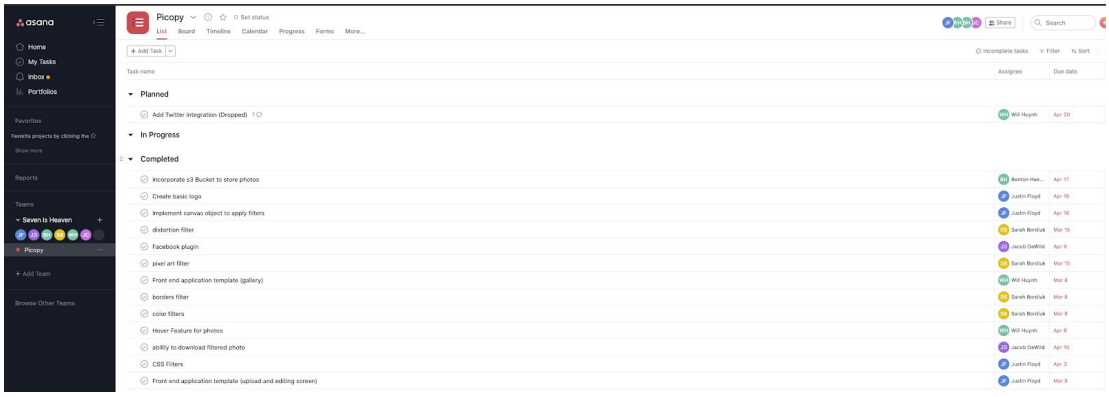
The app is built on Node.js, and more specifically, the express web framework. When a user (who has registered and logged in) uploads a photo, it is converted into a canvas, where filters (both CSS and custom from a javascript library) can be applied. The canvas is then converted into a base64 format where it can be POSTed to the server. The server writes the data to a local png file and then uploads this file to the Amazon S3 Bucket, where all photos are stored. Concurrently, the server sends an INSERT query into the database to store the new photo with its list of filters, url, and the user who created it.

When the user navigates to the gallery, queries are sent to the database to receive the photos and load them up. Picopy only sends the initial query to the database to maximize speed (the app does not need to reload all photos on every sort or filter request). From the gallery page, the user has the option to filter their photos by user, by filter, by their own photos, or view all photos. They can also (in addition to the filtering), sort the photos by new or old.

Project Tracking Tool (Asana):

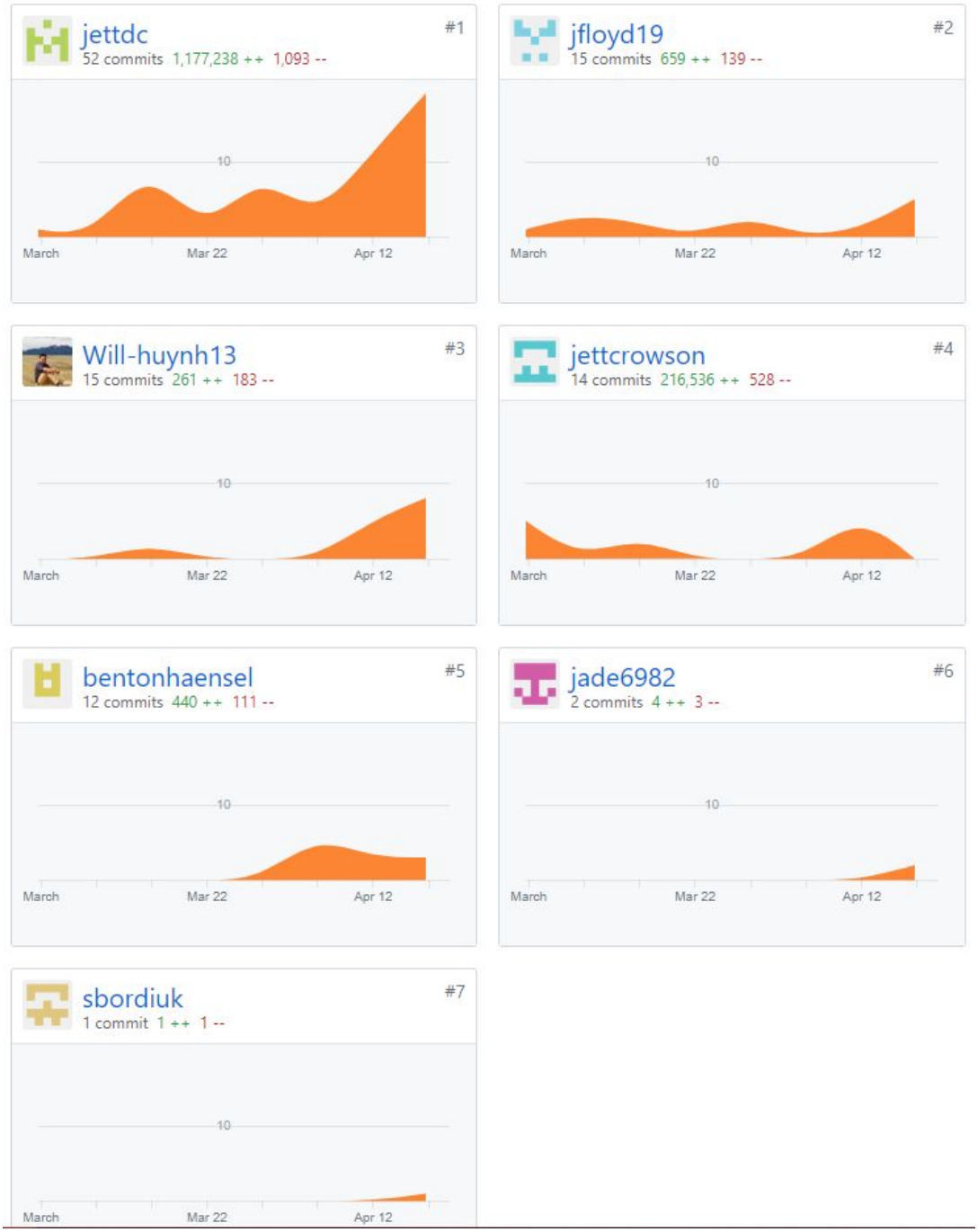
https://app.asana.com/0/1161745936726670/list

Join Link (must be using @colorado.edu email):
https://app.asana.com/share/colorado/picopy/221372384534556/ded04b06a1c76dfe574d7997460f6d45



Version Control (GitHub):

https://github.com/jfloyd19/Components



Brief of Each Members Contributions:

Jett Crowson

- Set up node and express, designed and implemented gallery layout, designed home page, designed and implemented the database, deployed and integrated the database, designed and implemented entire user register, login, and cookie session system, implemented all of gallery sorting and filtering sections, implemented private photos section of gallery, implemented download buttons, assist on Amazon S3, implemented all advanced javascript filters from external library, implemented canvas functionality, assisted with hover boxes in gallery, drafted test cases.

Justin Floyd

- Implemented initial editing screen and its functionality, implemented css filters, implemented canvas object and its conversion to an image, completed facebook share button, drafted test cases, created and published basic picopy logo, pushed all project milestones

Benton Haensel

- Implemented file uploading feature from user input image with applied canvas. Adding photo data to database from canvas image.

Will Huynh

- Added all CSS for hover feature, debugged, assisted gallery page layout, added basic facebook share button.

Sarah Bordiuk

- Changed height of an html element

Jacob Dewild

- Assisted on facebook button

Heroku Deployment:

http://picopyimages.herokuapp.com/