Final Report

McKay Bonham

My proposal was to create an online photo gallery, where users could upload photos and then view them. I succeeded in doing this on a basic level.

The big difference between what I planned on at the time of the proposal and what I finally created is that I planned originally on having different users have different accounts, all with their own individual photo collections. Eric commented on my proposal and told me that authentication of user accounts would be difficult, and was beyond the scope of this course. Based on that, I eventually ended up doing away with separate accounts at all. Now all users see and contribute to the same photo gallery. This is an unfortunate limitation, but the implementation turned out to be pretty hard for me even with this simplification.

Otherwise my photo gallery website turned out pretty much as I imagined it, except for some minor formatting differences in how the final product would look. (I pictured more borders around different elements of the site, for example, but didn’t end up liking the way they looked.)

Other than the default Django template, I didn’t end up using much code from other sources. This wasn’t for lack of trying — originally I copied and pasted many lines of code from many sources, trying to make the images display correctly and the upload form work. For example, I tried to make the upload work using a Form Python object. But it didn’t work. These failed incorporations of existing code were the reason I eventually started over on my project, wiping the code clean so I could make it in a more simple fashion. (Hence having 2 GitHub repositories corresponding to the project.)

Obviously my project is much less ambitious than Imgur or Photobucket or other similar websites. The only way it can really compete with them is in its simplicity.

I do feel like I learned a lot from struggling with this project (and I worry that the end result seems too simple to the user to represent this struggle!). I had to brush up on my JavaScript, even though its application on the website is minor (making the “Previous” and “Next” buttons point to the right pages). I had to study in depth about the format in which uploaded files arrive in the website’s clutches. I learned about the importance of putting database components into folders that Django recognizes as part of the app, so that its permissions will allow referencing that information correctly. I re-learned the importance of keeping track of data types, as I got stuck for quite a while on a bug where I needed to convert a string into an int before using it. I learned that websites take a lot of work, and that adding complex features to fill them out (like separate accounts for separate users, or being able to view thumbnails of pictures, or re-order or delete pictures without using the admin GUI) needs to be done gradually.