Self Replication Technical Specifications

Software Used

Python

Heroku (Web Hosting)

Python Packages

Django (web framework)

pygithub (framework built around GitHub API)

Description

The app itself is simple. It consists of three webpages: a main page, a callback page, and an error page.

**The main page** offers the user a button that they can click to fork the original "Self\_Rep" repository. The button calls the GitHub API and creates an authentication callback code. This code is required to use the GitHub API to make any changes to the logged in user's repositories. This required me to register a new Github app which had a unique client ID associated with it. I use the pygithub module here because it provides a more developer friendly way to interact with the GItHub API.

Once authenticated, the GitHub API redirects to a callback page, which I have mapped to my callback() function in views.py. The callback function checks if the logged in user already has a "Self-Rep" Repository. If so, the user is directed to the **error page**. The error page is a static page that informs the user thusly and allows them to download the Docs for further reading. Otherwise, the callback() function forks the "Self-Rep" repository from my original account. Then, it displays the **callback page** which states that the process was successful.

Hosting

For hosting the app, my goal was to figure out a streamlined process that would allow a user to host their own version of the app without having to download any software packages (it can all be done via web browser). Heroku offers an easy to use web interface for deploying apps from a git repository. One decision I made to streamline the process further is the disabling of static files. Configuring a Heroku app to serve static files can be tricky sometimes, and since the app does not actually need to serve any files.

If the user follows the instructions in my Instructions Doc.pdf, they will create a new heroku app running the same code. In the code, I have my client ID hardcoded as a variable so any replicated apps will still rout through the original Self Replicating app. Making the replicated apps run as standalone apps requires the user to register their app through GitHub and make a push to their repository. This process might be complicated for users who are new to using GitHub, so I decided to make it an optional final step.