**Re=Search Aggregator** *(Tentative Name)*

*Soham Sahare, Shreya Patel, Sai Srinivas S*

We are developing a web application for users which displays a list of projects/research work being done/completed and the names of professors and other collaborators linked to those projects. The user can then use this data to finalize which school is the best fit for their graduate/ postgraduate studies and research projects. As mentioned before, our scope is limited to ASU’s CIDSE department for the moment.

We have developed the basic application workflow loop consisting of a web server, app-tier, Google Pub/Sub topics and subscription to provide asynchronous execution of jobs and Google Cloud storage for storing common files that are used by multiple tiers. The components pending to be integrated are Persistent Database and the NLP model (tested independently).

The HTTP server uses Flask and Flask-socketio (python) to provide a web-socket connection and a pair of GET and POST API endpoints to host a static website and take user input and display it on the front end. We publish these requests to Google Cloud Pub/Sub Topic which serves jobs to the app tier. The web server also subscribes to another Google Cloud Pub/Sub Subscription to read app-tier processing results. The app tier is also written in python and it is responsible for web scraping of the input URL and selecting relevant data using Regex and returning to the web server relevant results using Google Cloud Pub/Sub. Both of our web server and app-tier load useful constants such as topic/subscription names from Google Cloud Storage. This makes updating constant variables easier and avoids multiple deployments on the cloud.

Our next priority tasks are to implement a database (Google Cloud Datastore) to store intermittent results, topic classification using NLP model and improving on the performance of web-scraping logic. We are yet to test the application’s auto scaling capabilities and performance on GAE. It’s scheduled to be completed in the coming week and half.