**Project Team Name and Number** : Team 14: GradScout

**TA Name :**  David Randall

**Date Submitted** : March 4th, 2017

**I. Summary of Progress/Milestones Reached (since last reporting period)**

**Milestones Reached**

* Data Curation: Top 25 universities (65 Programs)
* User Surveys Data Analysis

**Summary of Progress**

* **Data curation:** The top 50 universities in the United States with around 165 graduate programs were curated, and the top 25 universities with a total 65 programs in Information and Computer Science domains were shortlisted. The final rankings have been normalized based on four ranking systems - US News, Forbes, Times and CWUR.
* **User research:** User survey and interview data has been collected and analyzed.
* **ERD and DB implementation:** Initial ERD Design has been designed and will be implemented in Django during the development stage.
* **Environment setup:** All the required software, environment, GitHub repo have been set up on the working laptops.
* **Design process:** User research data has been collected from the survey, the qualitative data collected was used to devise two user personas - who represent the potential users of our application. Initial paper prototype for the application has been created.
* **Development: Data Collation**
  + Current work is focused on data scraping for Research opportunities and Program Curriculum from the top 25 universities in the list.
  + Implemented web scrapers in order to retrieve the course information from the individual program website. The information retrieved is in the form of Course Number, Course Name and Description. The data needs to be cleaned and is currently stored in the memory as the python dictionary.
  + Apart from the curriculum data, information for the preferences such as University ranking, test scores, location, length of the program, program fees, school categorization, etc. has been manually collected.

**II. Deliverable/Milestone Status (since last reporting period)**

* Data Curation: Top 25 universities (65 Programs) - Green
* User Surveys Data Analysis- Green
* Capstone Deliverable I – Project Description - Green
* Capstone Deliverable II – Project Proposal - Green
* Capstone Deliverable III – Project Charter - Green
* Capstone Deliverable IV – Project Schedule/Plan - Green
* Capstone Deliverable V – Project Status Report - Green

**III. Resource Status (since last reporting period)**

* Green – All required resources are available to continue progress.

**IV. Sponsor Status (since last reporting period)**

* Green – Sponsor has followed communication plan and they have expressed satisfaction with progress.

**Additional Inputs from the Project Sponsor:**

**1. Regarding the Database Schema:**

The sponsor has advised us to understand and keep note of how the different factors (preference factors for universities) are related among themselves, so that we can understand patterns and interdependencies among these factors.

Example 1:

Rankings can be related to the acceptance rate and test score. Higher ranking implies higher GRE score requirements and lower acceptance rate. So identifying these dependencies and explanatory relationships in order to get weightage for the preference matching part of the problem is important. Statistical analysis can be done to get to know about these relationships and how strongly they are correlated to each other. Factor analysis can be one option for that.

Example 2:

Why are a majority of the universities are located in California, Washington and New York region? Possible reasons - Job scenarios, Type of Industries, Business Schools on East Coast and Technical Schools on West Coast...etc. How do these factors affect the weightage given to the student's preferences? For example: When it comes to Business related program preferences, Location preference weightage for the East Coast universities might be higher compared to the West Coast, etc.

**2. Maintaining a Central Database for all the information.**

Our project sponsor suggested that once data has been obtained from web scrapers, the ideal approach is cleaning and storing all the information in database tables like SQL. One thought was to repeatedly run all the web scrapers on locally stored web pages, query it directly from the Django application whenever we need to perform matching. But that approach is better suited for constantly changing data points.

As the data retrieved from the scrappers for this project will be stored in memory, the sponsor suggested to look for the ways to store the data from our variables into the databases. For example, is it possible to store data into the database table by directly accessing it from Python's Dictionary?

Since, the university related information is not going to change drastically over the period of this project, it makes more sense to get all the data from scrappers and keep storing it in the structured or unstructured format in a database and query that data from the web application depending on our preference parameters to provide final recommendations. The sponsor suggested to check for database connectors from Django to SQL, SQLite, Postgres, etc. so that they can be scaled depending on project needs.

**V. Other Status Points**

N/A