**Meeting Minutes 4 – Group # 3**

**Project: *SNA4SlackASU***

SER 517: Software Factory I

24th November 2017

**Group # 3**

|  |  |
| --- | --- |
| **Name** | **ASURITE ID** |
| Abhimanyu Dogra | adogra4 |
| Aman Srivastava | asriva26 |
| Ishan Dikshit | idikshit |
| Nikhil Tibrewal | ntibrewa |
| Sanchit Narang | snarang2 |
| Shuchir Inamdar | sinamda2 |

**Attendees**

Sponsor – Prof. Filippo Lanubile

Co-sponsor – Prof. Fabio Calefato

Team members – Abhimanyu Dogra, Aman Srivastava, Nikhil Tibrewal, Sanchit Narang, Shuchir Inamdar, Ishan Dikshit

**Meeting Duration**

Meeting start time: 10:30 AM MST Nov. 24, 2017

Meeting end time: 11:20 AM MST Nov. 24, 2017

**Topics Covered**

1. Progress of the project during last three weeks.
2. Usage of NetworkX (Python Framework).
3. Requirements and deliverables for 4th iteration.
4. Preparation for 5th meeting with the sponsor.
5. Scaling up the scraper to integrate with the web services in real time.
6. VM instances for cassandra cluster and Neo4J at cloud platform.
7. Flask api code with slack\_archive table structure.
8. Kinds of ties used for graph representation, closeness, betweenness and degree centralities.

**Resources and important links**

1. Social Network Analysis concepts

<http://www.di.uniba.it/~reti/collab/SNA.pdf>

1. Waffle: Kanban board for Issue tracking with Github and Slack integration

<https://waffle.io/aman-srivastava/SNA4Slack>

1. Github repository

<https://github.com/aman-srivastava/SNA4Slack/>

1. Slack conversations data source

<https://slackarchive.io/>

1. Data storage

Cassandra: <http://cassandra.apache.org/>

1. QUnit: Testing framework for JavaScript

<https://qunitjs.com>

1. PyUnit: Python Unit Testing Framework

<http://pyunit.sourceforge.net>

1. Wireframe

<https://en.wikipedia.org/wiki/Wireframe>

**Next Steps/Action items for 4th iteration**

1. Setting up of NgXdashboard for the front end.
2. Establishing the link between front end and cloud platform to build a sample angular component.
3. Addition and execution of unit tests for networkX poc implementation
4. Storing the metrices calculated from archives into the cloud database in a web service readable format.

**Requirements and progress discussed**

1. Crawler scaled up in a way where providing team name collects data from each channel of the team and updates to cassandra cluster at cloud platform. Presented the data through logging in to the Bitnami cassandra instance created at cloud.
2. Flask api code shown which is mapped with object relation entity and is integrated with slack spyder which executes the running of web service with provided urls.
3. If cassandra provides latency, then we will used the csv file which the crawler is capable of generating autonomously from slack archive.
4. Discussed the Neo4J prototype implementation which saves the data in graph database performing the metrics computations in real time.
5. Enhanced the initial wireframe fitting in Social Network Analysis metrics.
6. Presented the networkX library implementation which took 3-5 seconds for one team, calculated degree centralities and betweenness, and eventually generated the graph using matplotlib python library.
7. Discussed the ties used for graph representation building a “mention” model based on mentions of various users with an average of 70-80 mentions.

**Next meeting(Tentative)**

Friday, 15th December 2017 (3-week cycle)