**03/07/2017 Meeting Minutes**

**Project Name:** Capstone 3

**Meeting Time/Location:** 5:30p / Room 279

**Meeting Chair:** Nikitha

**Meeting Secretary:** Tulasi

**ATTENDENCE:** (A=Absent; X=Attended)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Anson | X | Anusha | X | Kavya | X | Hrishikesh | X |
| Manjiri | X | Nikitha | X | Pavitra | X | Apoorva | X |
| Mathias | X | Elaine | X | Tulasi | X |  |  |
| Dr.Pawaskar | X | Praveen K. | X |  |  |  |  |

**AGENDA:**

1. Demo and Presentation - 60 min
2. DB queries to the Client- 30 min
3. One on One/ Sprint Planning - 60 min
4. Data Model queries and discussions- 15 min
5. Retrospective  - 30 min
6. Assigning tasks and team discussions – Remaining Time

**ACTION ITEMS:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Task** | **Name** | **Date** |
| **1** | Update hours in YODIZ | Everyone | Regularly |
| **2** | Implementing edit and save peer groups | Elaine | ASAP |
| **3** | Identifying the dependencies and prioritizing, splitting the user stories accordingly | Mathias, Manjiri | 3/9/2017 |
| **4** | Template and guidelines for the documentation | Mathias | 3/9/2017 |
| **5** | Analysis on D3.js and how it can be implemented in Laravel | Apoorva, Tulasi, Kavya | 3/9/2017 |
| **6** | Test User management and create Laravel dusk scripts for it | Anson, Hrishikesh | ASAP |
| **7** | Update Data model, create final CSVs for multiple years and create migration scripts for the new tables | DB team | 3/9/2017 |
| **8** | Determine how to implement the computation on columns | Development and DB team | ASAP |
| **9** | Check with the Client what should be displayed on default scatterplot | Elaine | Next client meeting |
| **10** | Go through the new data model and provide suggestions and inputs to the DB team | Everyone | ASAP |
| **11** | Rename the columns appropriately | DB team | ASAP |

**SCRUM STANDUP MEETING TASKS:**

* No Scrum meeting on Mar 7th

**CHAIR AND SECRETARY SCHEDULE:**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Chair | Secretary | Complete |
| 1/10/17 | Elaine | Mathias | Y |
| 1/17/17 | Elaine (2) | Mathias (2) | Y |
| 1/24/17 | Pavitra | Nikitha | Y |
| 1/31/17 | Apoorva | Elaine | Y |
| 2/7/17 | Anson | Hrishikesh | Y |
| 2/14/17 | Hrishikesh | Anson | Y |
| 2/21/17 | Mathias | Anusha | Y |
| 2/28/17 | Manjiri | Pavithra | Y |
| 3/7/17 | Nikitha | Tulasi | Y |
| 3/14/17 | Kavya | Aproova |  |
| 3/21/17 | Pavitra (2) | Manjiri |  |
| 3/28/17 | Praveen | Kavya |  |
| 4/4/17 | Anusha | Nikitha (2) |  |
| 4/11/17 | Mathias (2) | Pavitra (2) |  |
| 4/18/17 | Tulasi | Praveen |  |
| 4/25/17 | Manjiri (2) | Mathias (3) |  |
| 5/2/17 | n/a | n/a |  |

**Key Items Discussed:**

1. **Discussions with Client on Demo and presentation:**

* + Show date/time in military 24-hr format for all the Jobs
  + Sort by institution alphabetically while displaying schools for peer groups
  + Additional filtering for peer groups if Bandwidth available
  + Few additional variables are added to the calculations and variables sheet
  + Handle the scenario where Admin uploads the wrong files in to the database by mistake and after files have been processed, Admin uploads the correct files

**Q.** With the current design, is every user forced to create peer groups?

**Ans:** No, Guest user can use the default (public) peer groups which are created by admin and accessible for every user on the website. Guest user do not have access to create the peer groups

**Q**: Are we adding more filters?

**Ans**: If there is a bandwidth available after completing all the features then team will look into adding additional filters else no

1. **Database discussion with Client:**
   * For schools csv file, entire csv is loaded into the database without removing any columns
   * For other csv files, only variables provided in the computations and variables sheet are loaded into the database
   * Default rate- Display default rates only for the years where there is data and for other years display the values as N/A in the UI
   * No calculation on Loan default rate column
   * Lindsey will get back to the team if there are any other columns to be loaded in the Finance table. As of now only one column is being considered which is from Finance non-profit private csv.
   * Admissions table is not included in the new data mart
   * Employees will be replaced with Instructional staff
   * Admin salary is the total of all the salaries of all categories
   * In the calculations and variables sheet, first variable is numerator and the second is denominator used when deriving computation variables
   * In the UI, Degree per thousand students will be replaced with corresponding columns – more than one column
   * Client will check and get back to the team if it is required to maintain data for all the years in Carnegie
2. **Learning’s:**
   * Scrum master to have a check point for every task and get the updates from the team regularly
   * Update the code on GitHub daily even if it is not complete
   * If there is any question that the presenter can’t answer during Demo, ask the team to answer instead of providing a vague explanation to the client
   * Client is not able to follow the presentation. So, give a brief Introduction on the features while starting the presentation. For example- user should be able upload files
   * Understand and analyze the data properly before moving on to the database design
   * No need to normalize the table to the degree as it is done in case of transactional data
   * Organize the presentation, anticipate the questions from the client and be prepared with the explanations
   * Giving a dry run to the team might avoid the possible mistakes done during the Demos
3. **Retrospective discussions at consolidated level:**

**What went well?**

Team is improving in each sprint

Demo was good in spite of all the implementation issues

Good team work and resolving the implementation issues at the last minute

**What did not go well?**

Communication gaps between the teams

Updating Yodiz daily basis

Did not take time to design the functionality before coding

**Challenges:**

Documenting everything is difficult but team can try implementing it

**Suggestions:**

Stand ups should be done properly and no hesitations even if you have issues

Give update to the team on Thursday about the task and if the task can be done or not

Comment the code so that the other person can track back

Include another scrum to discuss what are the decisions made for resolving the impediments

Minutes of meeting should be completed right after the meeting

Document the discussions- why is the team approaching a particular way and what is followed by a particular team to achieve a task – can be kept in Yodiz

1. **Team discussions:**

Scrum reports are sent to Professor on Tuesday morning and after meetings on Thursdays and Saturdays.

Move the tasks to done as soon as they are completed

DB team – figure out how to do computations- may be stored procedures

Data model should have only the tables which are final

Can client do the computation outside the scope of the project?

1. **Heroku:**

* The code will be deployed in the next sprint when Dr. Pawaskar is comfortable with the code
* Dr. Pawaskar suggest to use PostgreSQL on Heroku

1. **Presenters for the Demo 3&4:**

Apoorva, Elaine and Anusha