Abstract

Many students nowadays enroll in one or more Massive open online courses (MOOCs) which offers video based lectures. These MOOCs provide discussion forum for each course where students can discuss about the course. Instructors and students rely on these asynchronous forums to engage one another in ways that potentially promote meaningful problem solving and knowledge construction. The need for analyzing forum data can provide valuable insights to instructors that will be helpful to enhance the course if needed and also for preparing next release.

Currently there are no existing systems onsite for analyzing discussion forum. However, there are various tools available for social network analysis and discussion forum analytics. There are no implementations yet for both layers i.e. quantitative and qualitative. The proposed system will perform quantitative and qualitative analysis on data and will provide educators and researchers with meaningful insights.

There will be 3 users of system: course instructor, mooc administrator and mooc researcher. The approach used by us is object oriented. First, interface will accept *bson* file which is then converted to *csv* file that can be used for quantitative analysis using javascript library (d3.js). The dashboard for each user will be different as per their requirements and analysis they want from forum's data.

Table of Contents

List of Figures

Chapter 1:

Introduction

1.1. Objective

In the context of our project, a Learning Management System is a web application for the administration, documentation, tracking, reporting and delivery of educational courses or training programs.

The discussion forum is a platform in the system, where the members of the MOOC can hold conversations in the form of posted messages. It is a key tool in new knowledge building among students in Learning Management Systems. The components of a discussion forum may include Comment Threads (the initialization of a post), the Comments (answers or supporting questions in regard to the post) and Replies (response to the comments).

The huge number of Threads makes it difficult, for instructors and researchers to evaluate the social cohesion among the learners so an automated support for the analysis is needed.

Information overload causes a problem for the instructors to evaluate the positive and negative points of a course, which if facilitated, will help them to improve the course content.

At the completion of this project, we aim to achieve the following things:

- Create a GUI that will analyse the discussion forum data, among the participants and give graphical output. These outputs will be helpful for improving the current or running courses and, to modify the future courses.
- Using this, the instructors can summarize, reinstate, challenge and foster discussions, monitor understanding and take a feedback enabling them to explore ideas, promote valuable arguments and threads while closing off unproductive discussion.
- Create graphical visualizations like network graph, scatter graph, bar graph, pie chart, transition graph and alluvial graph, which will help the instructor to get a better understanding of the students' response to his courses and which week's material does he needs to change or update so that the activity increases in those specific weeks.
- The admin should be able to add a user, assign a specific course to a single or a group of users, add a specific set of courses and upload the data dump in bson format.
- The dashboard should display a summary of the analyzed data dump, where all the number of total and active/non-active users, number of comments, comment-threads, length and popularity of the course.

1.2. Methodology used

Data collection methodology:

It is divided into two categories primary and secondary methods.Secondary data is which has already been published in journals or publications etc whereas primary data is collected through interviews, questionnaires etc.Qualitative analysis is based on secondary data collection methodology

For research we referred various publications (mentioned in next chapter) to explore what ideas and methods could be used in developing such systems. We searched few papers according to the modules of our system for eg. what parameters can be

considered to get active users of discussion or which tools are used for generating network graph (interaction among users). We performed qualitative research with open-ended questions to develop an understanding how system will work (functions), what preconditions and assumptions of application.

Design methodology:

We have used bottom-up design and structured design where we identified inputs and desired outputs to create graphical representation. we identified system modules and designed basic layout of system and backend. After developing individual modules integrated them to form whole system. we developed few diagrams(in chapter 5) to get an overview of system i.e how it will look and overall workflow of system.

Testing methodology:

Testing methods used by us are: unit testing, integration testing, system testing and acceptance testing. The developed modules were individually tested then one by one each module was integrated and tested.Then, the whole system is integrated and tested as a to check if all the modules can work in unison. Finally, acceptance testing was conducted with a few instructors who used the system and provided us with some insights and based on those, basic modifications were made to the design and UI.

1.3. Organization of the report

The main body of report is preceded by title, certificates, approval and abstract of project. This is further followed by 5 chapters which summarize the objective, methodology of system, system requirements and survey.

Chapter 1 explains objective of system, describing the need for system and methodology explaining object oriented approach which is used by us.

Chapter 2 is literature survey where all the 12 journal papers and proceeding papers we referred are summarized.

Chapter 3 is problem statement which defines our system.

Chapter 4 is system requirements comprising of functional requirements that are modules of our system followed by hardware and software requirements stating various software's used and operating system we will be working on.

Chapter 5 gives an overview of design which consists of various diagrams. As we are using object oriented approach we will have class diagram for system and database schema. This also contains system block diagram.

Chapter 6 gives implementation details which has module wise description and assumptions to be considered in each module followed by snapshot of the interface.

Chapter 7 provides various test cases to check all functionalities of system working correctly or not.

Chapter 8 gives what results we came across from analysis we performed on various courses and compared the results among them.

Chapter 9 gives conclusion and future scope followed by references to what all papers and proceedings and git links we referred for this project. All references are in IEEE format.

Chapter 2:

Literature Survey

To get a clear idea of the problem statement, we referred a few publications and proceedings from the IEEE Xplore and ACM Digital Library respectively. We referred different papers to implement different modules.

Those papers have been categorized and summarized below:

1. Most Discussed Topics

[8]This paper contains simple statistical indices inspired by the work in the text analysis field. Interactions in a forum can be: interaction between peers (the students) and interactions between students and instructors (teachers and tutors). Lexicon model is used to identify different keywords and also their frequency in each message as well as the threads of the discussion forum. It can also be used to identify keywords to group the messages in the forum into particular topics and subtopics.

[10]This paper has focused on reducing the instructor's burden of going through the whole discussion forum in order to get a sense of what the students are discussing or what problems the students are facing. The authors of this paper are proposing a solution that includes grouping together similar topic discussion so that the instructor can easily go through the discussion forum. The initial posts were used to create models as they are a very cost saving way to get idea of all the range of topics being discussed in the forum. The authors used a toolkit named MALLET toolkit to develop topic models.

[9]In this research, contents of discussion forums of students were extracted into for mining patterns. They deployed topic detection and data visualization tools to analyse the discussion forum data better to generate intelligence to understand how students are performing and feeling about the course modules they are taking. By using the Forum Graph, academic administrators can perform social network analysis to understand the interaction among students and teachers to identify the frequent contributors and passive observers. The Forum Graph can probably help teachers better understand the participations of their students in the forum discussion.

2. Active / Non-Active Users

[7]This paper discusses about an engine capable of analyzing an online platform to gain multitude of information such as the student's interaction with their peers and teachers, student's use of the academic content available online, etc. It uses popular open-source tools such as R language packages igraph, RMySQL, etc. The visualization clearly identifies the peripheral participants with the help of different coloring. The visualization also helps us to clearly identify the active participants of the network. These metrics about the online forum can give an instructor better means to assess the online participation of the students in the discussions.

[5]This study examined the social presence experienced by students in online forums. The qualitative data was analysed based on Rourke et al. measurement of social presence i.e. interactive response, affective response and cohesive response. By

quantifying the qualitative data, the study found that interactivity in the discussion boards served the purpose to maintain contact among the students.

[3]In this research, discussion forum users who continually and actively participate in the forum discussions throughout the course are identified. They employed different measures for evaluating whether those active users have more influence on overall forum activities. They have conceptualized MOOC forum composed with three types of interactions i.e. threads, posts and comments with that three measures of threads are analysed that is views, replies and duration of a thread. Users' participations in each week are accumulated to study the number of weeks each user has engaged in the MOOC forum. Based the result of analysis, users who constantly participate in forum discussions are identified as statistically more influential users, and these users also produce a positive effect on the discussions.

3. Largest Interaction Group

[1]The authors of this papers have created a discussion forum analysing tool called as iForum which takes as input, the discussion forum content and then offers a set of novel visualization designs for presenting the three interleaving aspects of MOOC forums (i.e., posts, users, and threads) at three different scales. Visual analytics techniques have been proven effective in exploring forum data in an intuitive and interactive way.

[4]The paper gives information about how NetworkSeer visualizes interactions in the forum, including where and when the interactions happen.Along with Statistical analysis of the forum it also provides an insight on Network view in a very informative manner. It is possible to customize the views based on specific parameters like Country, Time period, particular sub-forum. This provides great flexibility to analysts in concentrating on areas they are interested in.

4. Categorization of Course

[6]This paper talks about the problems of the points reward mechanism and presents some suggestions for future MOOC forums. The points reward mechanism can motivate students to write more posts, but students are also likely to write ”spam

posts", only to make themselves look like an active participator to earn the extra points. It analyzes the contents of students' and Instructors' posts in order to understand their behavior in the forum. It tries to find out both the positive and negative impacts of the points reward mechanism in MOOC course forums.

[2]The paper aims at helping the users in searching their query in most accurate & effective way possible from the large data set available. A thread or comment is termed as transaction (T) having a transaction-id (TID). The usefulness of a transaction (T) is quantified based on two measurable indicators – Support & Confidence. The Support value of item A with respect to T is defined as the proportion of transactions in the database which contains the itemset A. The Confidence value of a rule, AàB, with respect to a set of transactions T, is the proportion of the transactions that contains item A which also contains B.

[12]The unsupervised machine-learning approach utilizes each forum post as an utterance for the k-medoids clustering algorithm that takes actual data points as the centre of each cluster. These clusters undergo qualitative evaluation using a text-mining tool called "Topic modelling" to form more interpretable and cohesive models. It is possible to apply an unsupervised modelling framework developed for synchronous conversations to asynchronous discussions with an additional step of topic modelling in order to form more interpretable and cohesive models.

Chapter 3:

Problem Statement

To create a web interface to gather information about social cohesion of participants in a Massive Open Online Course (MOOC). The course platform is based on Open EdX Cypress Edition and cohesion is calculated strictly based on Discussion Forum participation. The system will assist the MOOC Administrator and course Instructor with information such as total registered learners, most active groups and learners, number of threads and comments, interacting groups and MIS report. An instructor can also be a researcher; the researcher can compare different courses.

Chapter 4:

System Requirements

4.1 Functional Requirements

MOOC Administrator login:

MOOC admin shall be able to login to the system using his username and password. He will have access to admin dashboard on the server that will enable him to upload the Discussion Forum data dump, monitor system usage and add a course or instructor.

Instructor login:

Course instructor shall be able to login to the system using his username and password. He will have access to instructor dashboard that will provide him with all the analytical information concerning from his/her course.

Dashboard:

Dashboard shall act as an interface between the user and the server. Based on type of user login, dashboard will provide them with list of functionalities. The user can select any functionality provided on the dashboard, depending on the type of request, server will respond by displaying the analysis result in graphical / tabular format.

4.2 Hardware and Software Requirements

Client Requirements:

Hardware:

- 2 GB Ram
- Core 2 Duo or above
- Server Access

Software

- No Special Requirements.

Server requirements:

Hardware:

- Ubuntu 16.04
- 2 GB Ram
- 50 GB Memory

Software:

- Python 3.6.1
- Mongo dB 3.4
- Anaconda 5.0
- Xampp 7.0

Chapter 5:

Project Design

5.1 Class Diagram

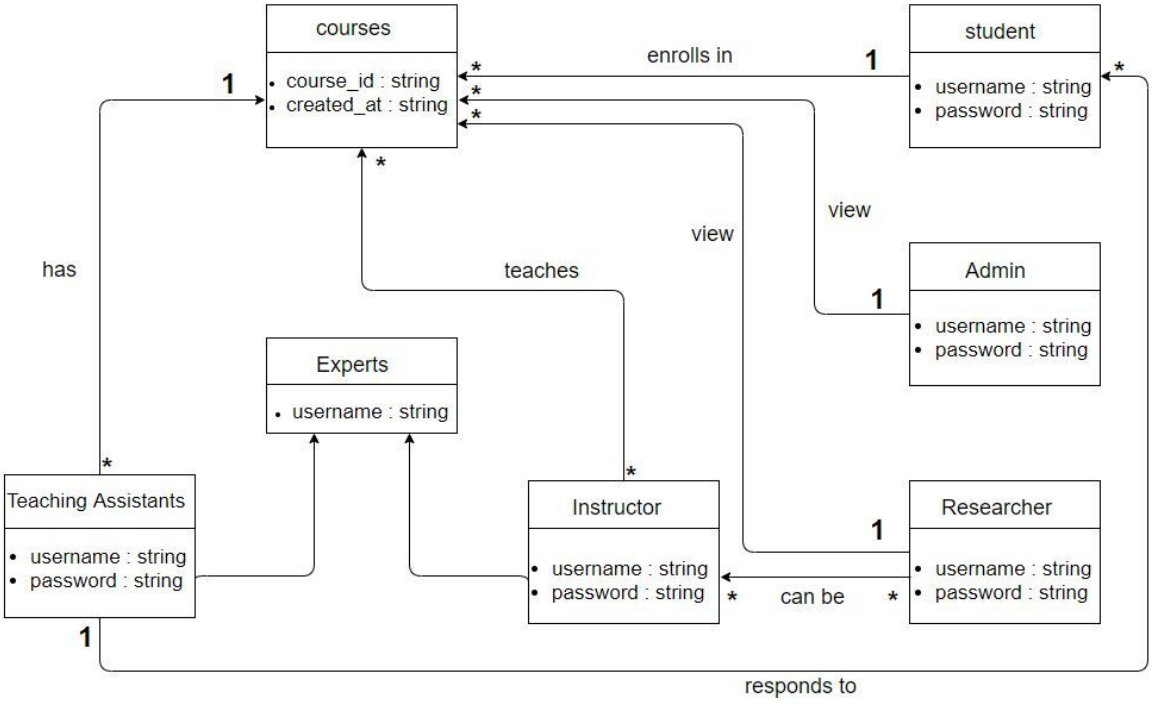


Fig 5.1

Classes:

1. **Course:** Courses offered in MOOCs and each course has its own discussion forum.

2. **Student:** Student can enroll in many courses and can participate in forum to interact with instructors and other students.

3. **Experts:** Experts are instructors and teaching assistants for a particular course.

4. **Instructors:** Instructors are the one who provides students with quality course and compelling learning experiences. They also participate in forums to interact and clear doubts of students.

5. **Teaching Assistants:** TA helps instructors with course materials and responds to student's questions. They are also expert in that course and can provide accurate answers to questions.

6. **Admin:** Admins can view all courses offered by MOOC. Admin would be interested to know how many registrations for each course and percentage of active users for that course.

7. **Researcher:** Researcher may or may not be an instructor for many courses and can compare cohesion of two courses. Researcher will be interested in active groups and their activity in forum.

5.2 Use case Diagram

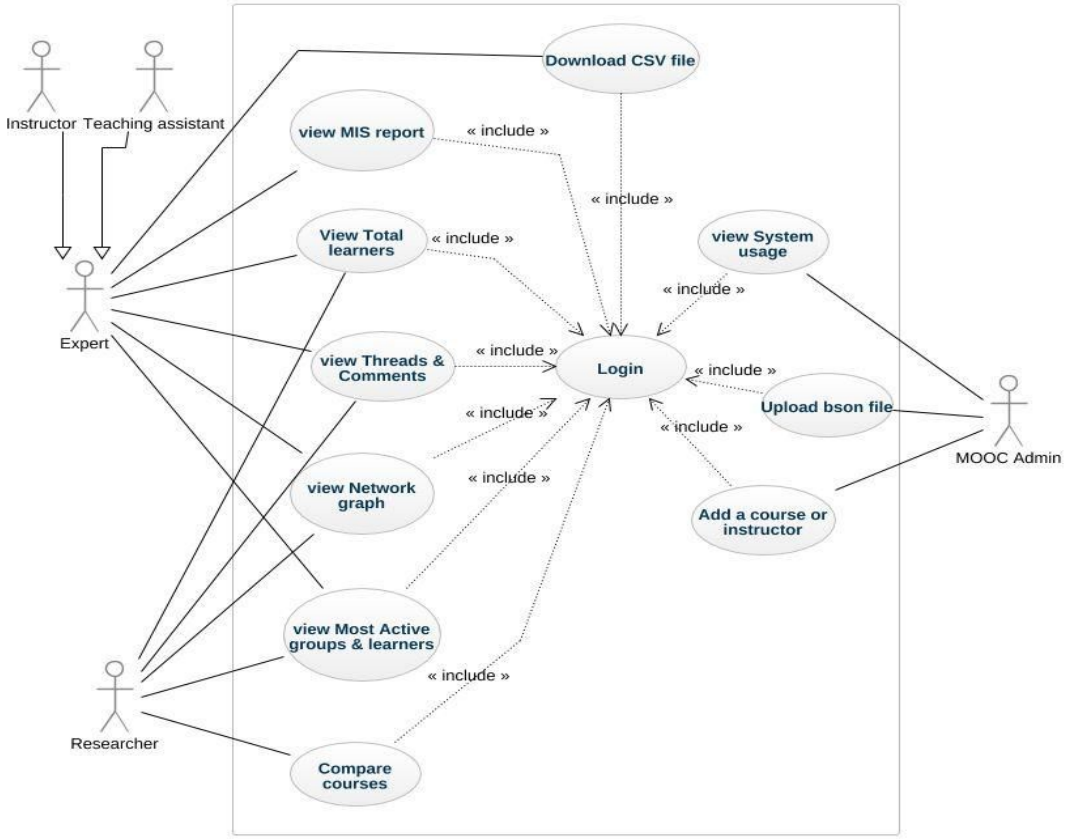


Fig 5.2

Actors	Use cases
Expert	1. View MIS report 2. View Threads & comments 3. View Network graph 4. View Active group & learners 5. View Total learners 6. Download CSV file
Researcher	1. Compare Courses 2. View Threads & comments 3. View Network graph 4. View Active group & learners

	5. View Total learners
MOOC admin	1. Upload discussion dump 2. Add a course or instructor 3. Monitor system usage

table 5.2

5.3 Database Schema

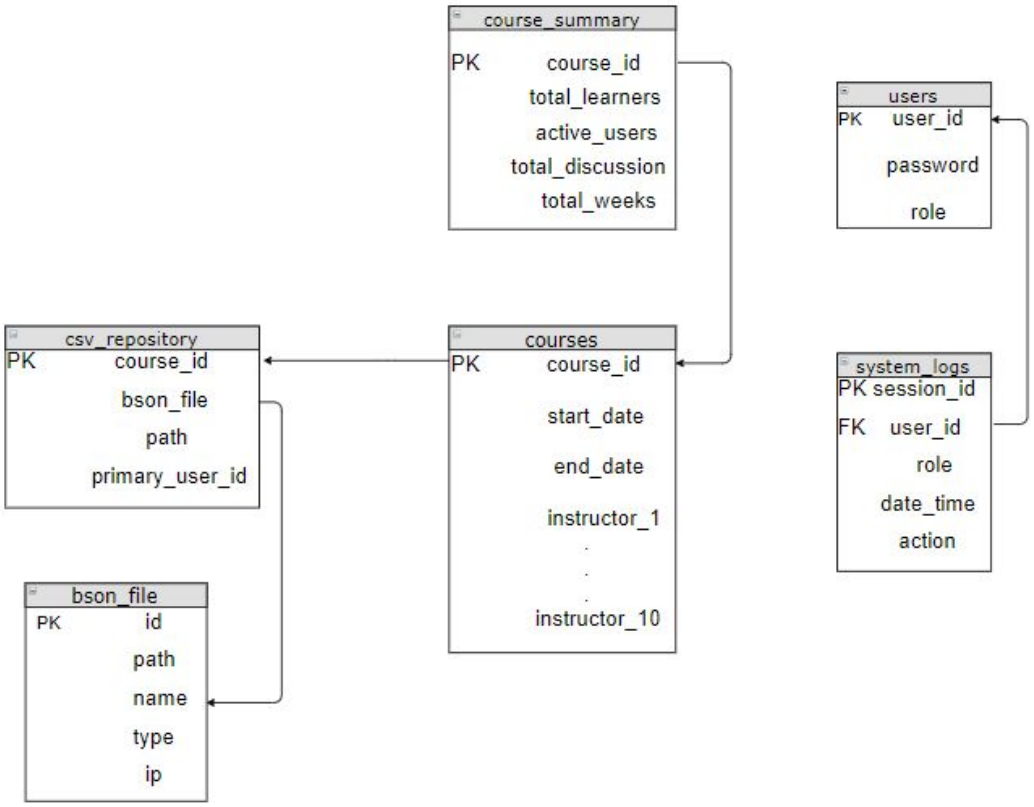


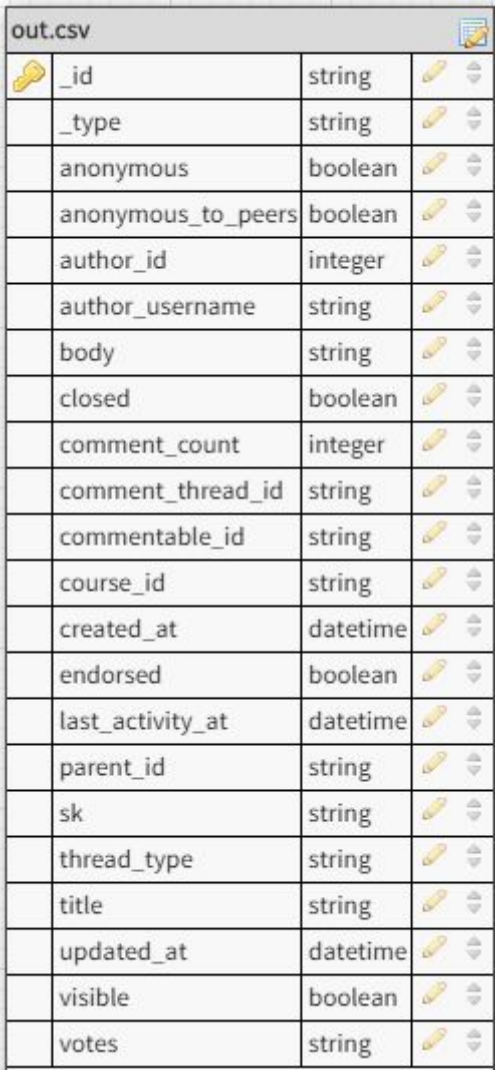
Fig 5.3

Components:

- **course_id:** The full course id of the course for which instructor is responsible.
- **user_id:** Unique user id of the instructor
- **role:** Role of user whether instructor or admin.
- **session_id:** Unique Session id.
- **date_time:** Date and time of generated csv file.

- **action:** Activity performed by user will be recorded.
- **start_date:** Start of course.
- **end_date:** End of course.
- **path:** Path of folder where file is stored.

5.4 Discussion Forum Schema:



out.csv		
_id	string	
_type	string	
anonymous	boolean	
anonymous_to_peers	boolean	
author_id	integer	
author_username	string	
body	string	
closed	boolean	
comment_count	integer	
comment_thread_id	string	
commentable_id	string	
course_id	string	
created_at	datetime	
endorsed	boolean	
last_activity_at	datetime	
parent_id	string	
sk	string	
thread_type	string	
title	string	
updated_at	datetime	
visible	boolean	
votes	string	

Fig. 5.4

Components:

- **_id**

  The 12-byte MongoDB unique ID for this collection. Like all MongoDB IDs, the IDs are monotonically increasing and the first four bytes are a timestamp.

- **_type**

  CommentThread or Comment depending on the type of object.
- **anonymous**

  If true, this CommentThread or Comment displays in the user interface as written by "anonymous", even to course team members and discussion team members.
- **anonymous_to_peers**

  If true, this CommentThread or Comment displays in the user interface as written by "anonymous" to students, but members of the course team and the discussion team can see the author's username.
- **author_id**

  Identifies the user who wrote this. Corresponds to the user IDs stored in the MongoDB database as author_id.
- **author_username**

  The username of the person who wrote the discussion post or comment.
- **body**

  Text of the comment in Markdown.
- **closed**

  If true, this thread was closed by a discussion forum moderator or admin.
- **comment_count**

  The number of comment replies in this thread. This includes all responses and replies, but does not include the original post that started the thread. In this example, the comment_count for the initial CommentThread is **36**.
- **comment_thread_id**

  Specifies the id of CommentThread to which a specific Comment belongs.
- **commentable_id**

  A course team can attach a discussion to any piece of content in the course, or to top level categories like "General" and "Troubleshooting". When the discussion is a top level category it is specified in the course's policy file, and the commentable_id uses the format i4x-{org}-{course}-{run}-{name}. When the discussion is a specific component in the course, the commentable_id identifies that component; for example, "i4x-IITBombayX-IITBombayX-course-2015_2016"

- **course_id**

  The full course_id of the course that this comment was made in, including org and run. This value can be seen in the URL when browsing the courseware section. Example: IITBombayX/IITBombayX/2015_2016
- **created_at**

  Timestamp in UTC. Example: ISODate("2015-01-14T19:47:25.931+05:30")
- **endorsed**

  Boolean value. True if a forum moderator has marked this response to a CommentThread with a Thread_type of "discussion" as a valuable contribution, or if a forum moderator or the originator of a CommentThread with a Thread_type of "question" has marked this response as the correct answer.
- **last_activity_at**

  Timestamp in UTC. Example: ISODate("2015-08-10T11:28:15.876+05:30")
- **parent_id**

  Applies only to comments made to a response.Theparent_id is the _id of the response-level Comment that this Comment is a reply to. Note that this field is only present in a Comment that is a reply to another Comment; it does not appear in a Comment that is a reply to a CommentThread.
- **parents_ids**

  The parent_ids field appears in all Comment objects, and contains the _id of all ancestor comments. Since the UI now prevents comments from being nested more than one layer deep, it will only ever have at most one element in it. If a Comment has no parent, it is an empty list.
- **sk**

  If null, the _type is CommentThread, else type is Comment
- **thread_type**

  Specifies the type of the CommentThread viz. discussion or a question.
- **title**

  Specifies the title of the CommentThread.
- **updated_at**

  Timestamp in UTC. Example: ISODate("2015-01-14T19:47:25.931+05:30")

- **visible**

  If true, then the CommentThread or Comment is visible.

- **votes**

  In the user interface, students can vote for posts (CommentThread objects) and for responses, but not for the third-level comments made on responses. All Comment objects still have this attribute, even though there is no way to actually vote on the comment-level items in the UI.

5.5 System Design

5.5.1 System Block Diagram

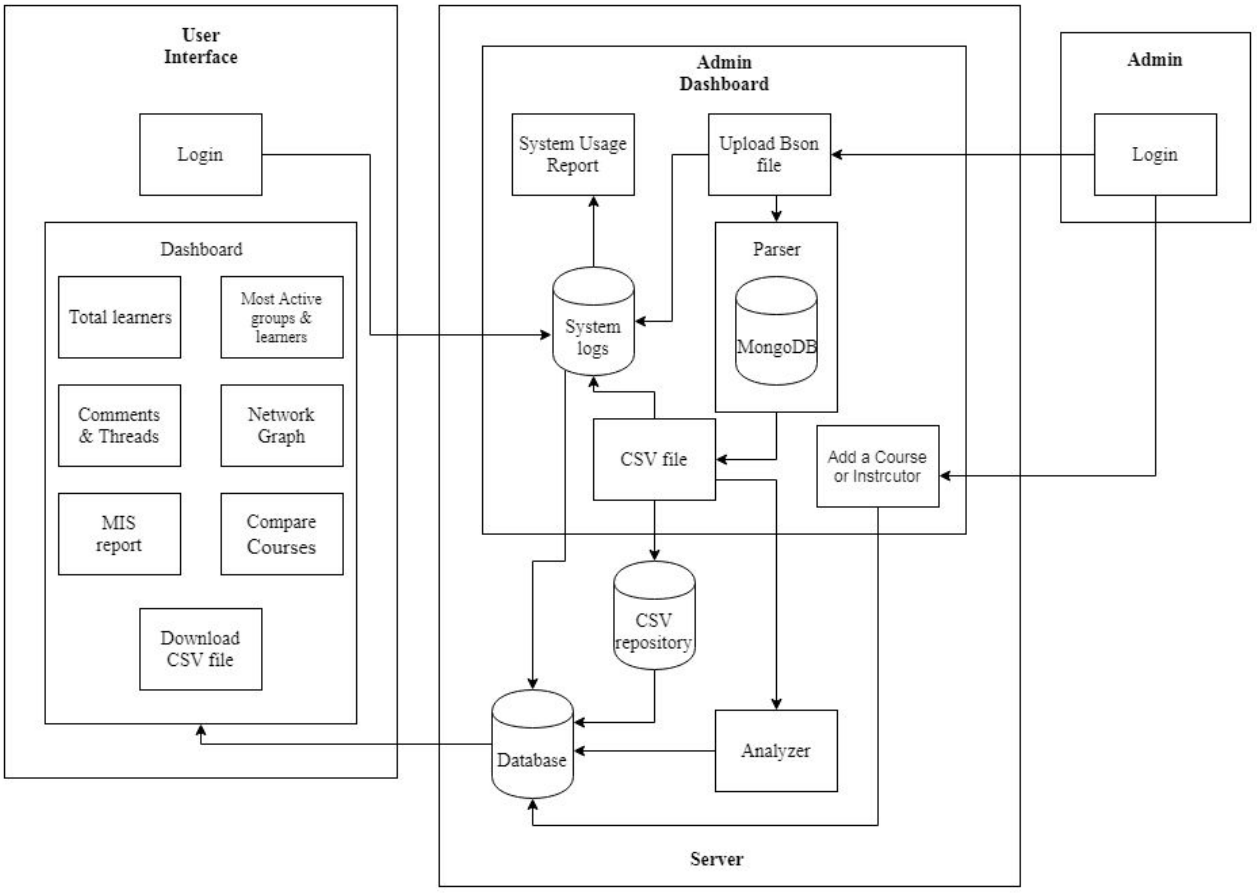


Fig 5.4

The MOOC admin will login to the system and upload the Discussion dump (bson file) onto the server and the bson dump will be converted into various CSV file depending upon their

course_id. These files will be stored in a csv repository database and all the activities will be reflected in the System logs. Admin can monitor overall system usage through system logs.
Database: It will map author_id with the course_id of which the logged in user is instructor of and display analytical results of those courses onto the dashboard of respective Instructor.
CSV repository: It will contain all csv files mapped with their course_id and available for instructors / researchers for download.

5.5.2 Workflow Diagram

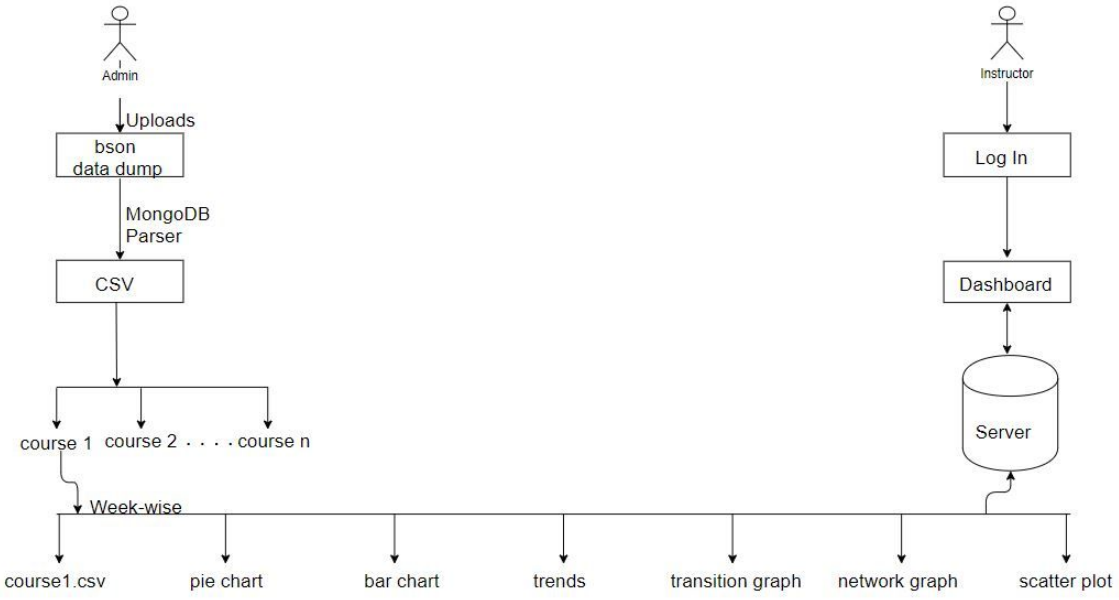


fig 5.5

This is the workflow of system i.e the sequence of activities performed by each user. As you can see, admin when logins will upload a bson-dump that is converted to csv files and separated into course-wise and week-wise. The week wise files for a particular course will provide various visualisations. This will be stored to server so when instructor logins and selects one course all visualisation will be displayed.

5.6 Algorithm

5.6.1 File Separator. (Converting the main csv file into course-wise csv files)

Step 1 : Import the required python libraries.

Step 2 : Receive the file name and path from upload.php

Step 3 : Read the main csv file into pandas dataframe.

Step 4 : Identify the list of unique courses.

Step 5 : Initialize course_len to length of course list.

Step 6 : FOR i=0 to course_len

Step 7 : If course[i] is equal to course_id of data frame then:

Step 8 : Add the frame into another data frame and create it's csv.

Step 9 : Insert the course_id, bson_file and path of file in csv_repository table..

Step 10 : END FOR

5.6.2 Active Non-active users. (Quantitative Analysis)

Step 1 : FOR i=0 to course_len

Step 2 : Read course[i] in a dataframe (eg. df)

Step 3 : Inserting the columns '_type', 'author_id' and'created_at' in df and sorting the data frame in ascending order of 'created_at' (dates)

Step 4 : Storing the start date of course[i] in a datetime variable 'dtv'.

Step 5 : Calculating the difference of each tuple from dtv and assigning it a week in data frame df['week'].

Step 6 : Initialize uw_len to length of list of unique weeks (uw) and create list of author_id from df.

Step 7 : FOR j=0 to length of df

Step 8 : IF _type is equal to 'Comment' then: append aid to 'c' list (comment list)
ELSE IF _type is equal to 'Comment Thread' then: append aid to 'ct' list (comment thread list).

Step 9 : END FOR

Step 10 : Create a list 'ctc' that has intersection of aid in 'c' and 'ct' list (Qualifies as Active User)

Step 11 : Create two list ‘a’ and ‘na’. (that serves as counter for active and non-active users) respectively.

Step 12 : FOR j=0 to length of uw_len

Step 13 : FOR i=0 to length of df

Step 14 : IF df[‘week’].iloc[i] is equal to uw[j]

Step 15 : IF aid[i] in ctc THEN: a[j]=a[j]+1

Step 16 : ELSE na[j]=na[j]+1

Step 17 : END IF. END FOR. END FOR.

Step 18 : Create a data frame and assign unique weeks, a & na as columns and convert it to csv.

Step 19: END FOR.

Chapter 6:

Implementation Details

6.1 Module and Description

Admin Dashboard:

MIS report: Includes system logs and user activity of all the MOOC courses. This report is exclusively available to the admin as he has the only access to all the courses' details.

File upload: The admin uploads the data dump in the form of a bson file, which will then be converted into a csv file. This main csv file is then split into multiple csv files based on the course names. Then, each course's csv file is split into week-wise csv files which is used to analyze the MOOC discussion forum data.

Add course: The Admin can add a new course whenever it is introduced in the system. He can even assign the existing or new instructor/instructors to the specified course.

Add instructor: The Admin can also add a new instructor whenever he/she is assigned a specific course. He can add the courses assigned to a specific instructor or assign multiple instructors to a specific course (with the limit of 10 instructors for any specific course).

Instructor Dashboard

Whenever an instructor selects his specific course from the list of his available courses, he is presented with a number of graphical representation options of the analyzed data like:

Download csv files: The instructor is provided with an option to download the csv file for the courses that he instructs. This will provide him with a readable format of all the data because the bson file is not in a readable format.

Display the list of courses: Whenever the instructor logs in through the instructor login screen, he will be shown the list of courses that he instructs so that he can select any course and view the analyzed details of that specific course.

Number of Comments, replies and comment threads: The instructor can view the quantitative figures like number of Comments, Comment Threads and Replies in the form of a pie chart for his specific courses.

Active & Non-Active users: This is a bar chart representation of the active and non-active users. This is in a double bar format and shows the Active as well as Non-Active users for all the weeks of a specific course between the start and end date of the course.

Trends: This is a bar chart representation of the number of Active and Non-Active users as well. This is in a single bar format and contains one bar for each week of the course between the start and end date. The main difference between this and the previous bar chart is that in this, each bar is divided into 4 layers (n1, n2, n3 and n4). The n1 consists of non-active users that have 0 contributions (Comments or Replies or Comment Threads) in that specific week. The n2 consists of semi-active users who have contributions between 1 to 4 in that specific week. The n3 consists of most-active users who have contributions between 5 to 10 in that specific week. The n4 consists of discussion drivers who have contributions greater than 10 in that specific week.

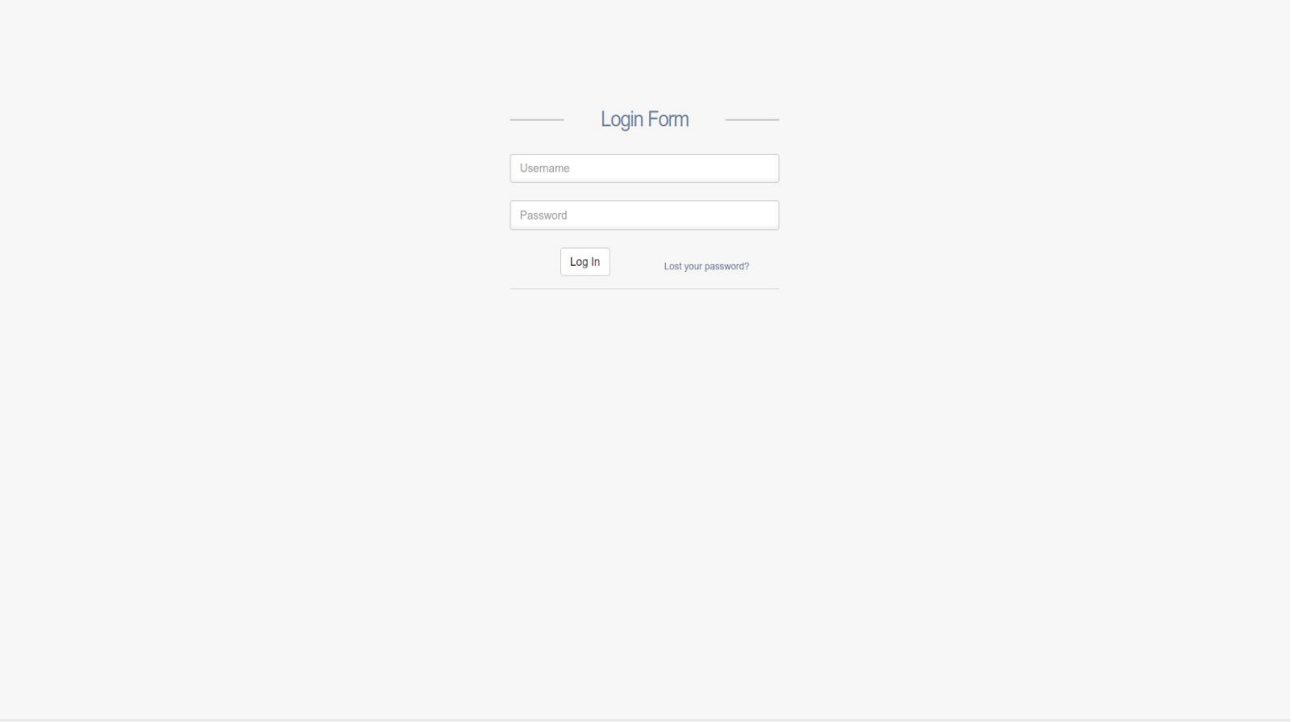
Transition graph: This graph shows transition of each author throughout the course week-wise.Each transition represents the contribution of that author in that particular week.The graph shows transitions for all the authors at once, however the instructor can select a window to see transitions of particular users.Also the transitions are in the form of lines and when the instructor hovers over a line he can view the author id for which the transition is displayed.

Network graph: This is a force directed graph in which the nodes represent each author and the edges represent three types of activities:Comment threads,Comments and Replies. The authors who have started a comment thread are connected to a null node and the authors who have commented in a comment thread are connected to the originator of the thread.The authors who have replied to a comment are connected to the author of the comment.When the instructor hovers over a node,the size of the node increases indicating its selection and information like author id,number of comment threads,comments and replies is displayed.Each node can be pulled in order to get a better view of the network graph.Also the nodes which are closer to the center are strongly connected in the network which indicates higher participation while the nodes in the outer radius indicate that they have lesser participation.

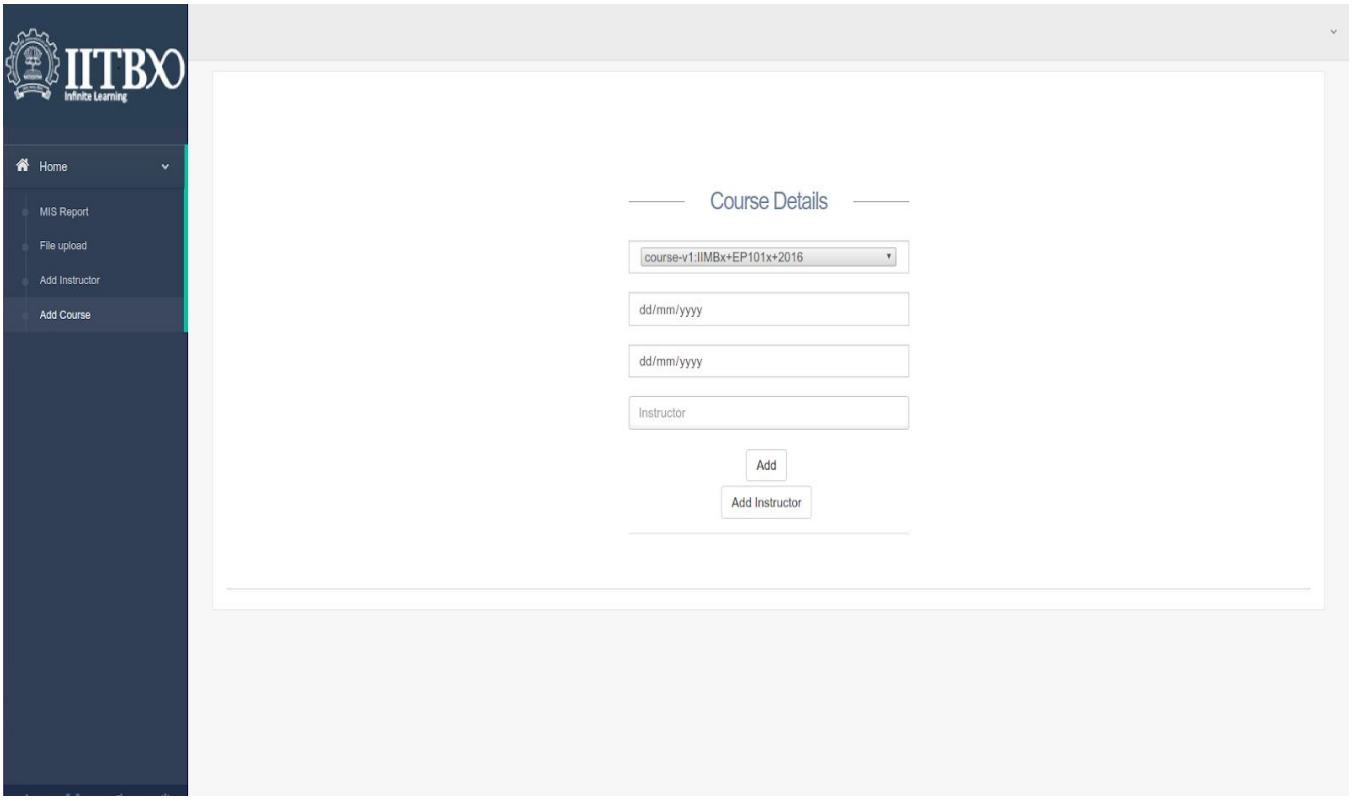
Scatter plot: The comments are analyzed by calculating their polarity and subjectivity which are then visualised using scatter plot. polarity is whether an expressed opinion is positive or negative. subjectivity expressions can be in the form of one's opinion,view etc. polarity ranges between -1.0 to 1.0 whereas subjectivity ranges between 0 to 1. In scatter plot, the x-axis represents polarity and y-axis represents subjectivity. The nodes represents author_id's when you hover on it will give sentiment(polarity,subjectivity) of that node.

6.2 Snapshots

Login Page:

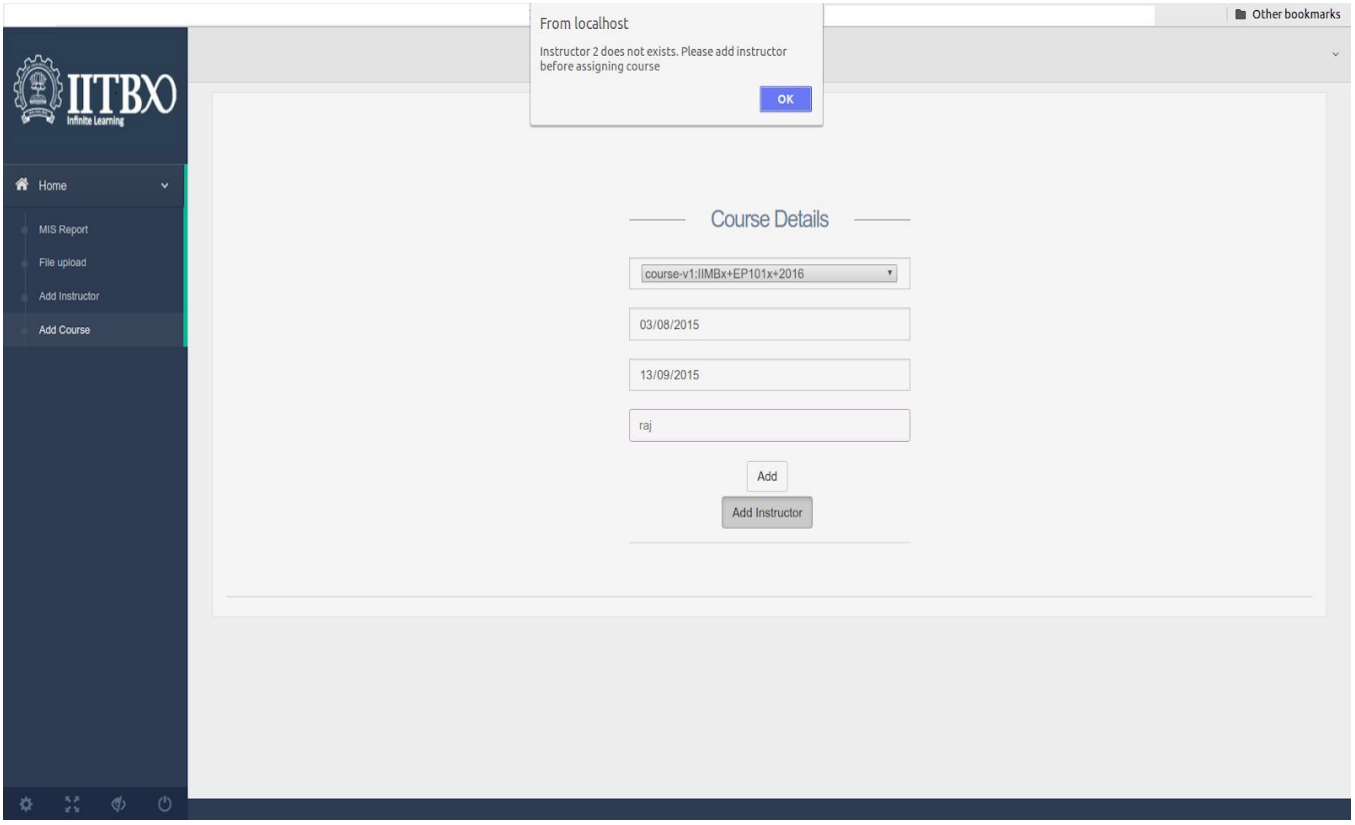


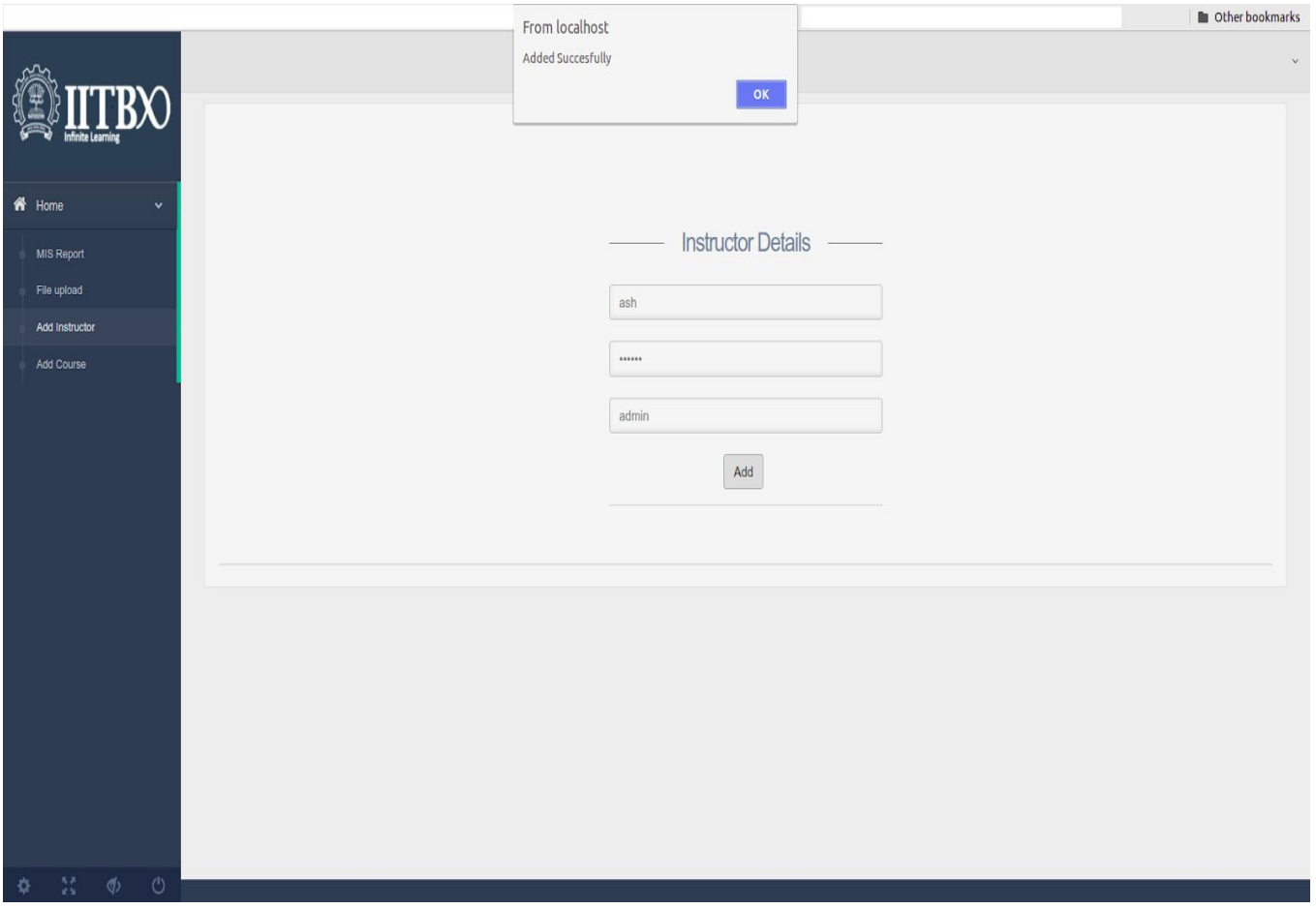
Admin Dashboard:



Add Course Page

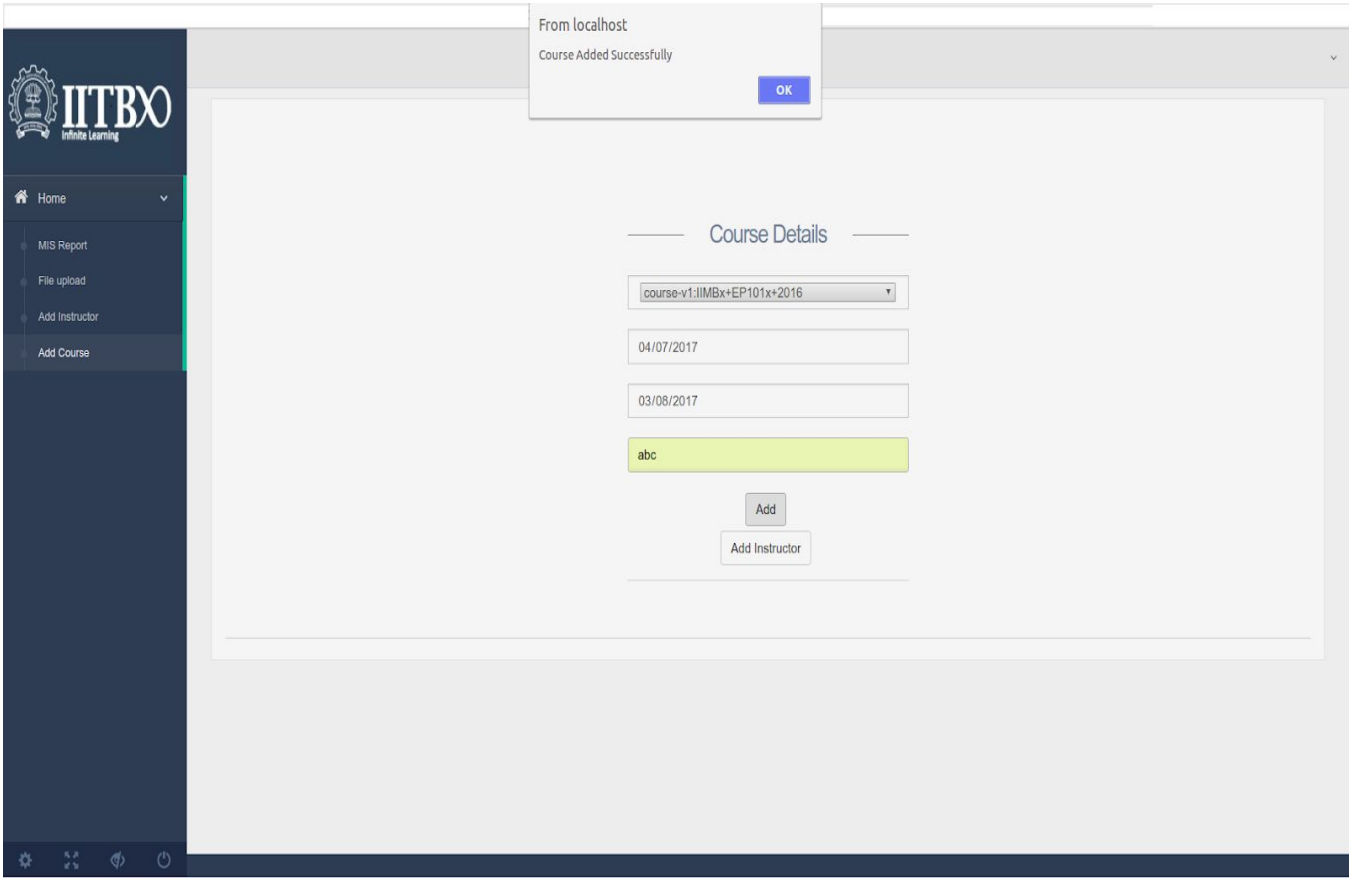
Assigning a course without adding instructor

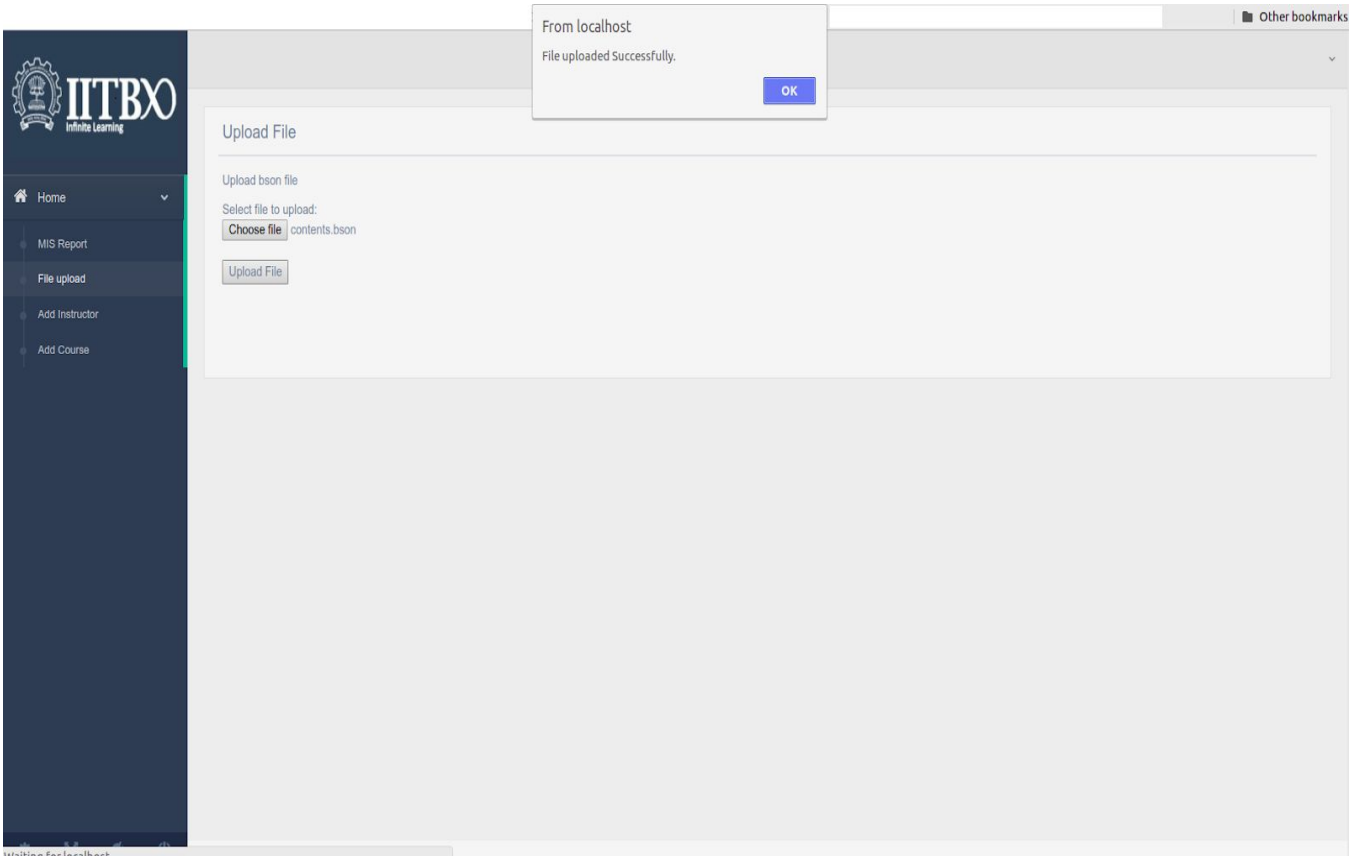




Adding an Admin

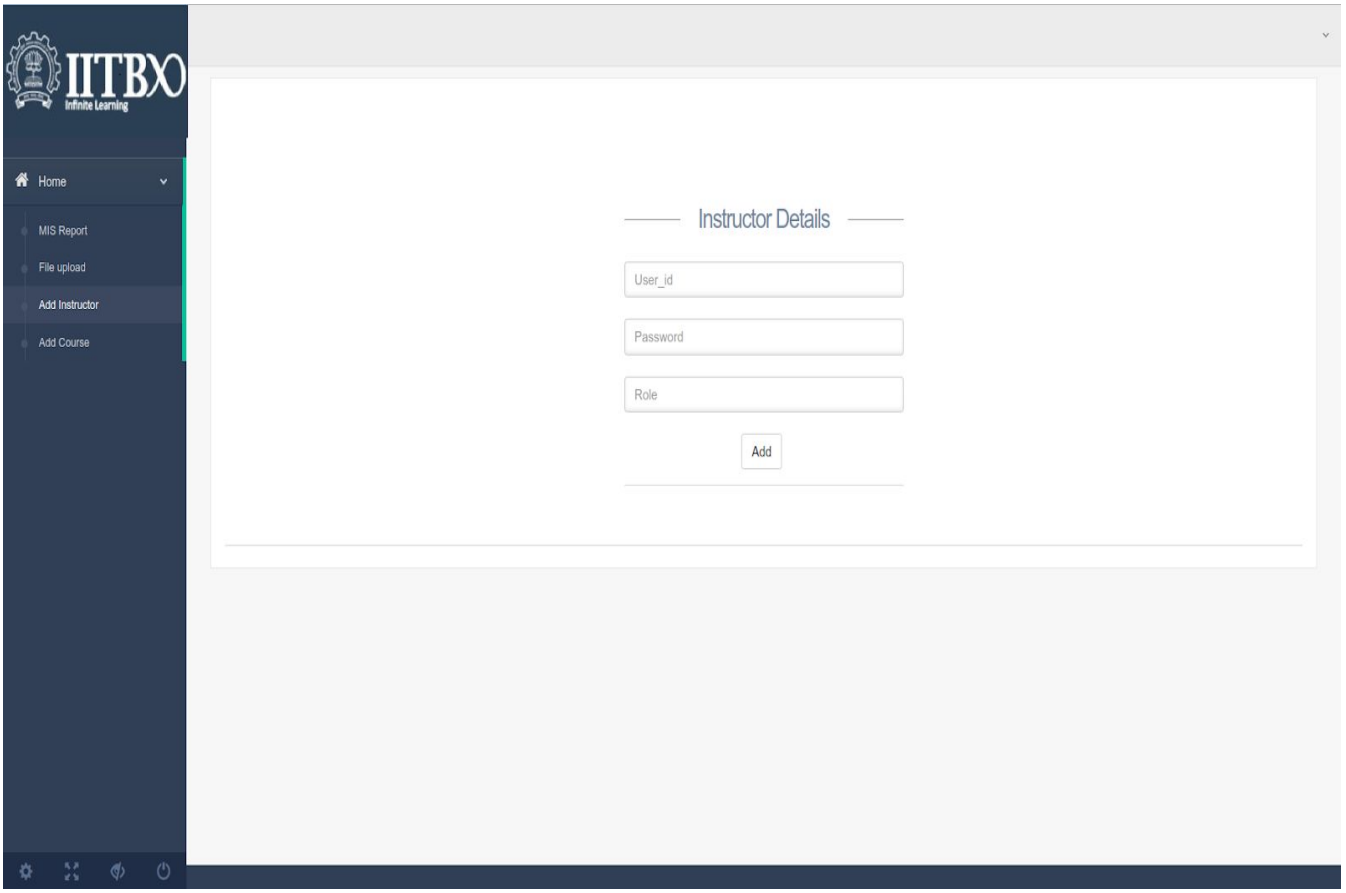
Adding a Course

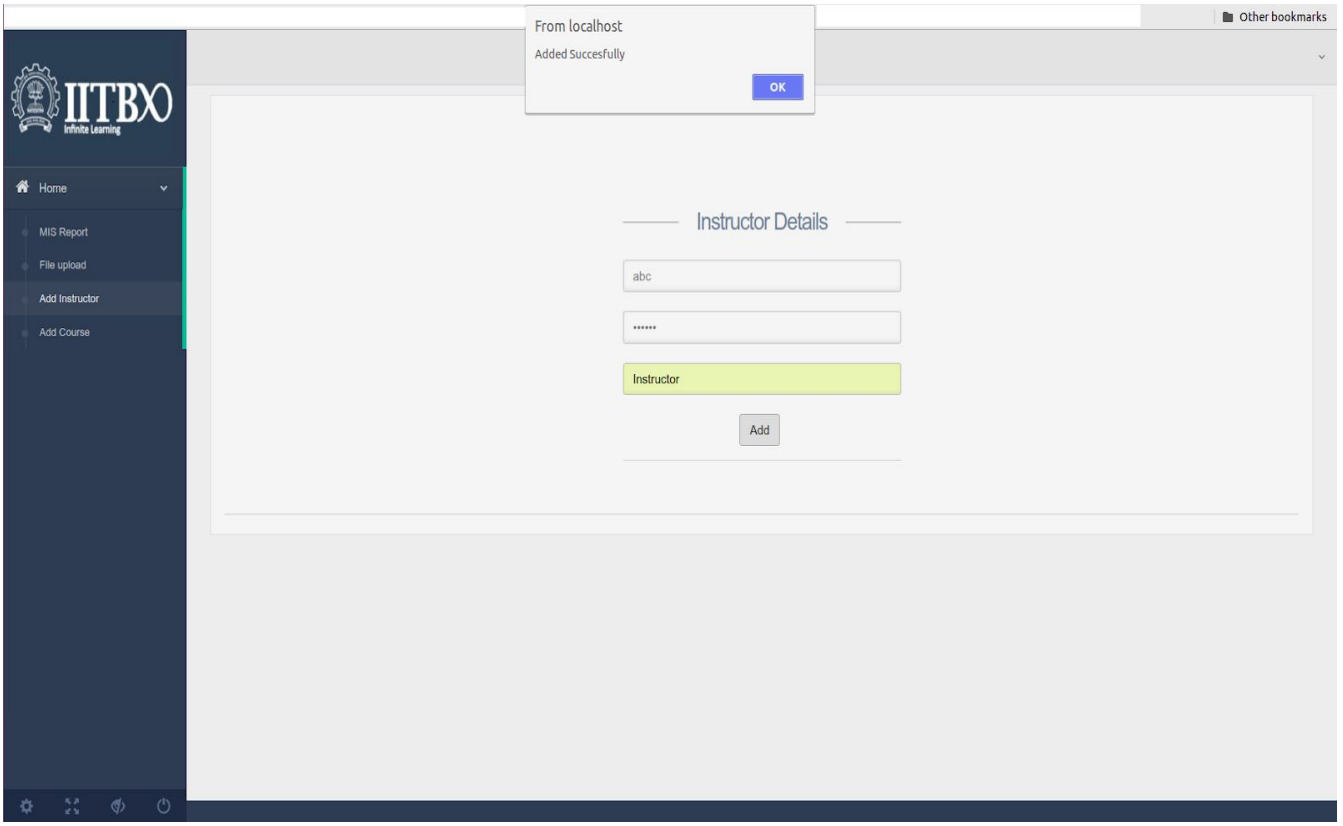




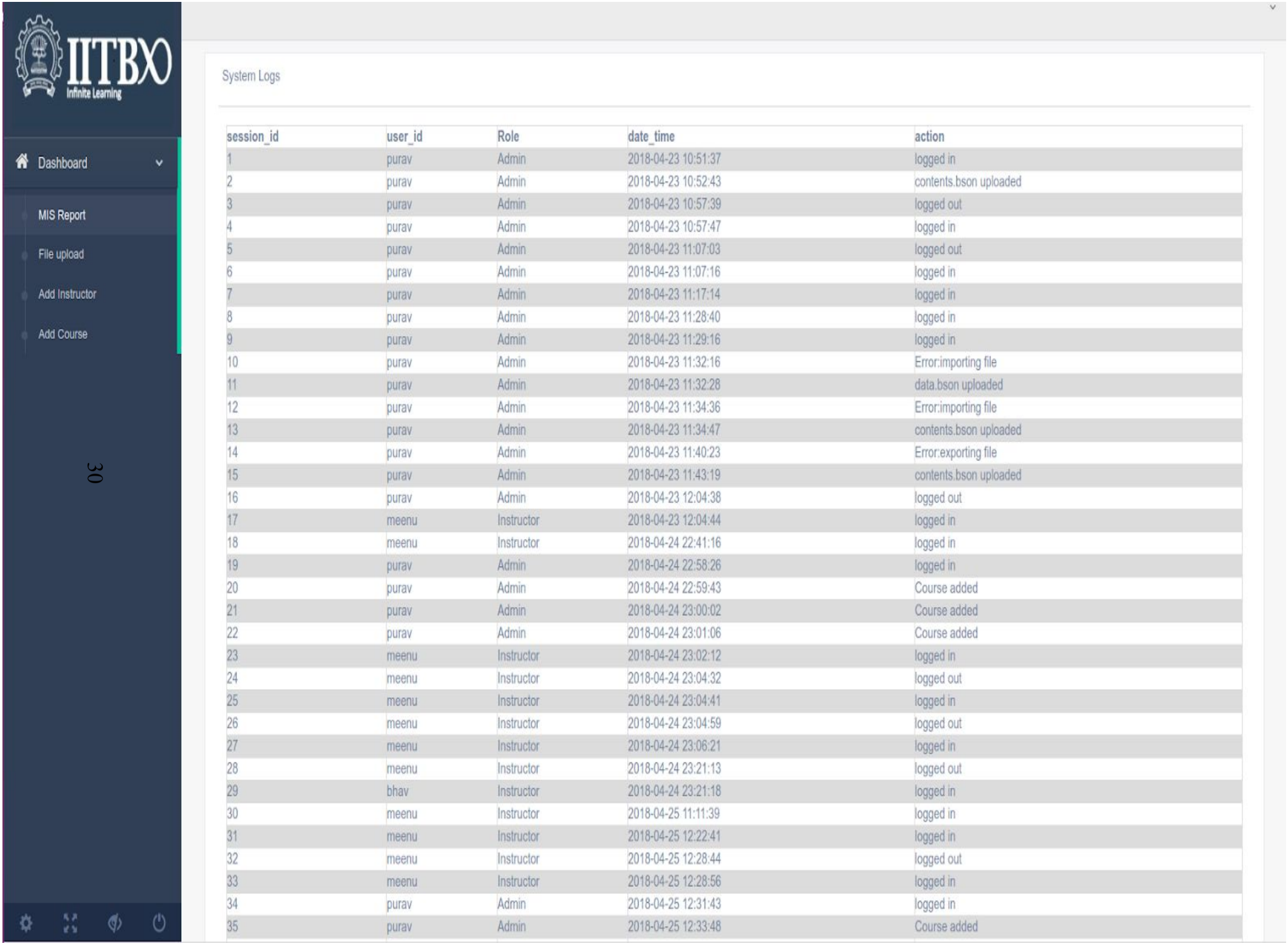
Uploading the bson data-dump

Add Instructor Page





Adding an Instructor



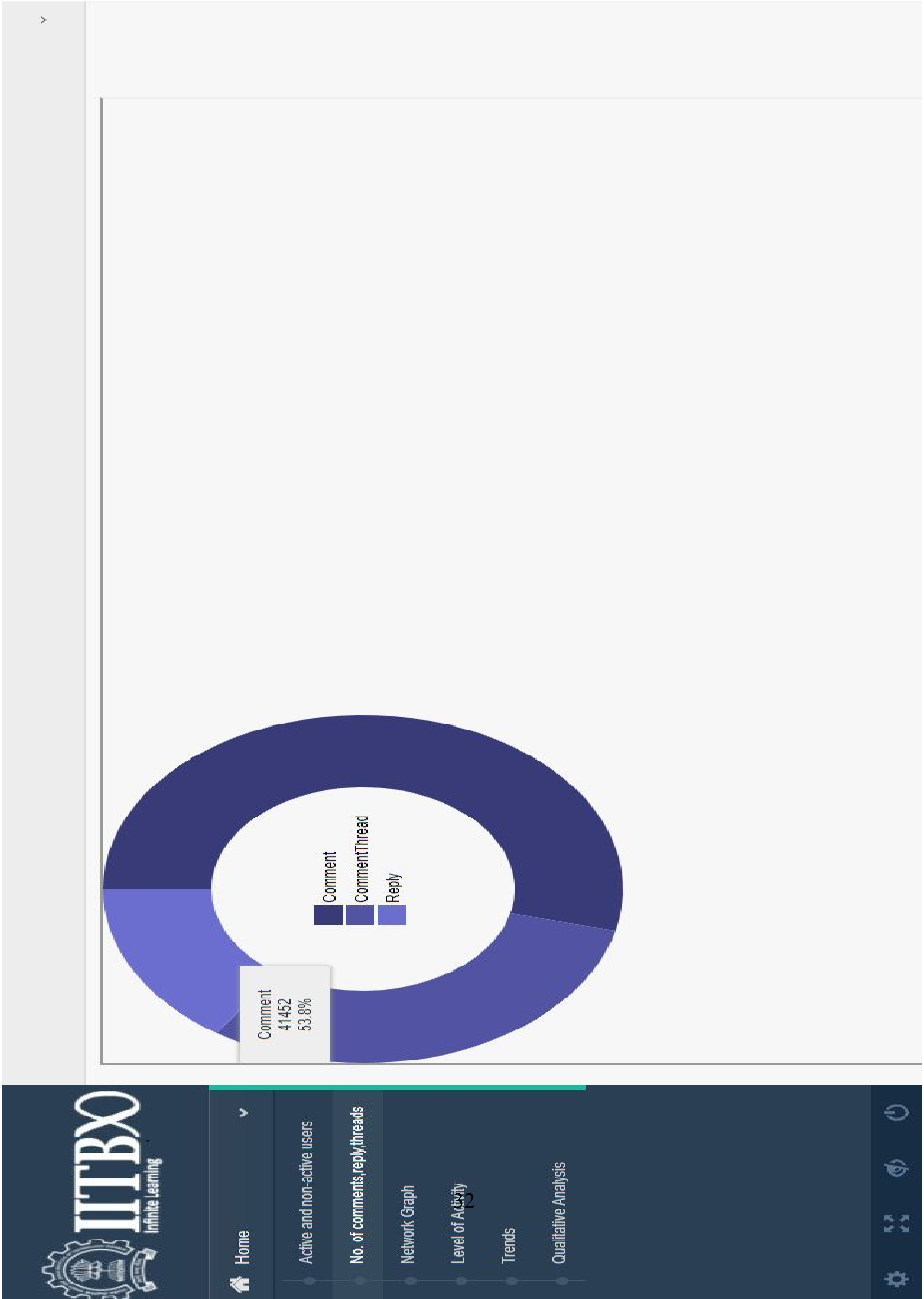
System Logs

session_id	user_id	Role	date_time	action
1	purav	Admin	2018-04-23 10:51:37	logged in
2	purav	Admin	2018-04-23 10:52:43	contents.bson uploaded
3	purav	Admin	2018-04-23 10:57:39	logged out
4	purav	Admin	2018-04-23 10:57:47	logged in
5	purav	Admin	2018-04-23 11:07:03	logged out
6	purav	Admin	2018-04-23 11:07:16	logged in
7	purav	Admin	2018-04-23 11:17:14	logged in
8	purav	Admin	2018-04-23 11:28:40	logged in
9	purav	Admin	2018-04-23 11:29:16	logged in
10	purav	Admin	2018-04-23 11:32:16	Error:importing file
11	purav	Admin	2018-04-23 11:32:28	data.bson uploaded
12	purav	Admin	2018-04-23 11:34:36	Error:importing file
13	purav	Admin	2018-04-23 11:34:47	contents.bson uploaded
14	purav	Admin	2018-04-23 11:40:23	Error:exporting file
15	purav	Admin	2018-04-23 11:43:19	contents.bson uploaded
16	purav	Admin	2018-04-23 12:04:38	logged out
17	meenu	Instructor	2018-04-23 12:04:44	logged in
18	meenu	Instructor	2018-04-24 22:41:16	logged in
19	purav	Admin	2018-04-24 22:58:26	logged in
20	purav	Admin	2018-04-24 22:59:43	Course added
21	purav	Admin	2018-04-24 23:00:02	Course added
22	purav	Admin	2018-04-24 23:01:06	Course added
23	meenu	Instructor	2018-04-24 23:02:12	logged in
24	meenu	Instructor	2018-04-24 23:04:32	logged out
25	meenu	Instructor	2018-04-24 23:04:41	logged in
26	meenu	Instructor	2018-04-24 23:04:59	logged out
27	meenu	Instructor	2018-04-24 23:06:21	logged in
28	meenu	Instructor	2018-04-24 23:21:13	logged out
29	bhav	Instructor	2018-04-24 23:21:18	logged in
30	meenu	Instructor	2018-04-25 11:11:39	logged in
31	meenu	Instructor	2018-04-25 12:22:41	logged in
32	meenu	Instructor	2018-04-25 12:28:44	logged out
33	meenu	Instructor	2018-04-25 12:28:56	logged in
34	purav	Admin	2018-04-25 12:31:43	logged in
35	purav	Admin	2018-04-25 12:33:48	Course added

Instructor Dashboard:

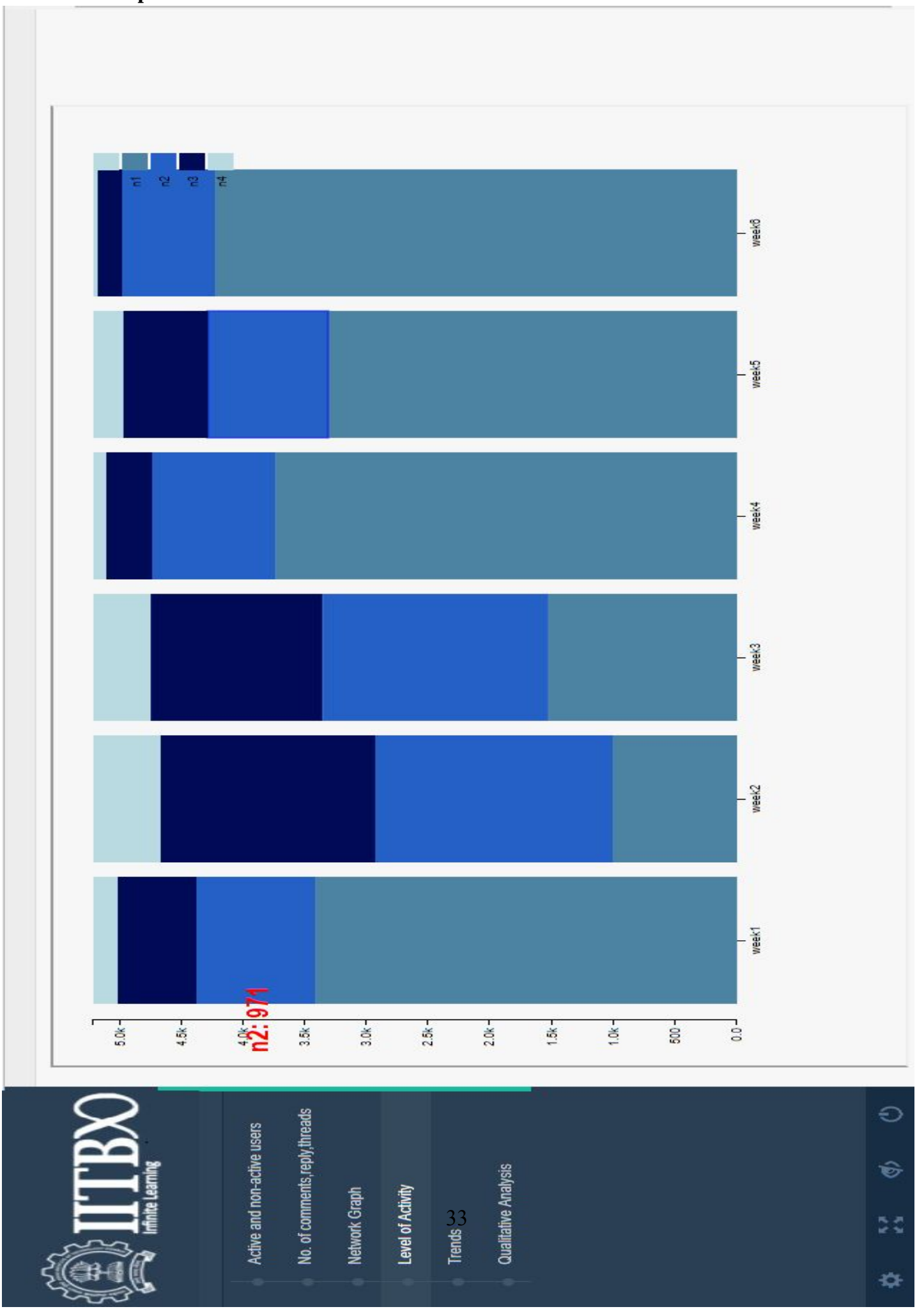
1. Bar Graph

2. Pie Chart

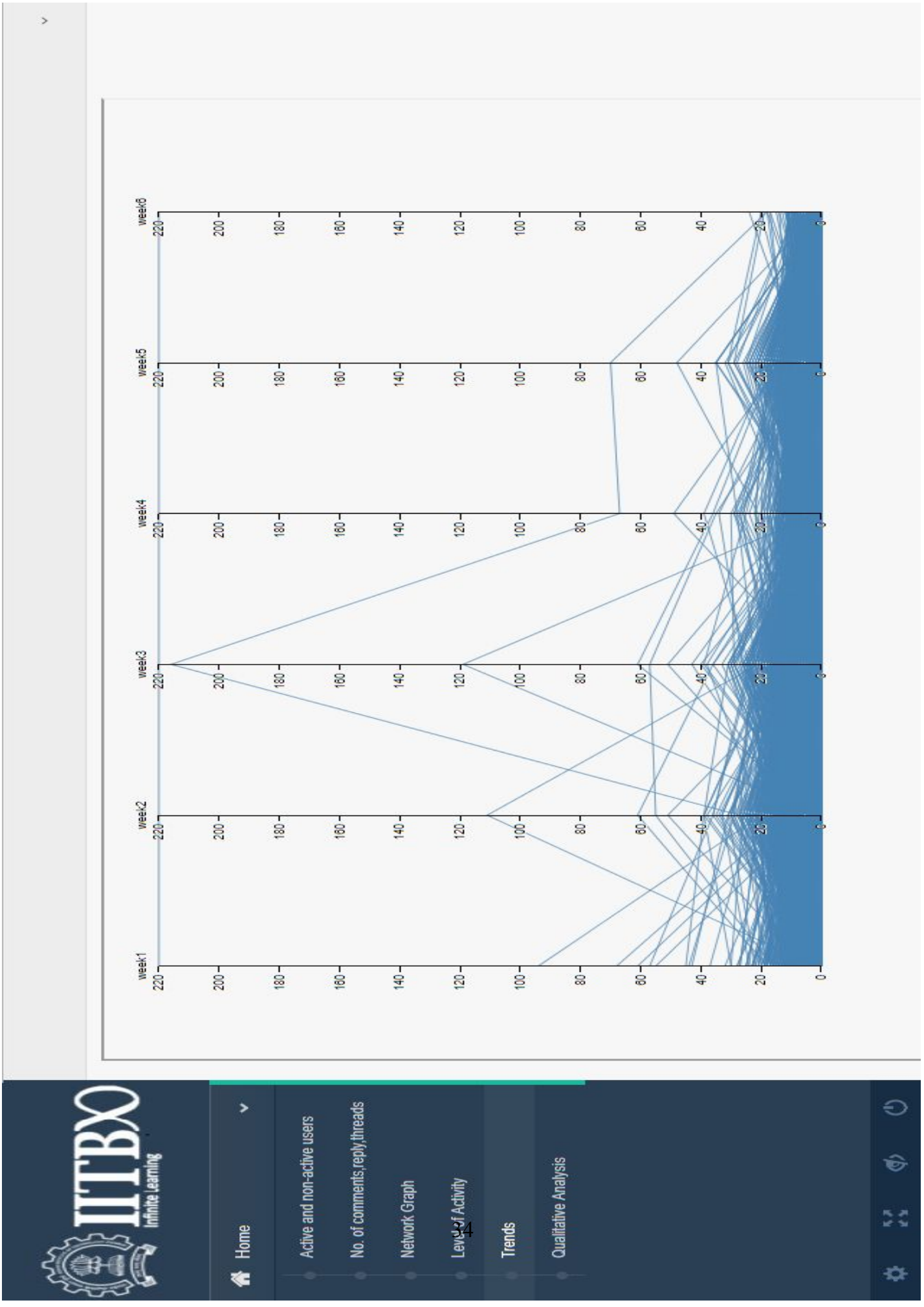


Comment
CommentThread
Reply
Comment
41452
53.8%
IITBX
Infinite Learning
Home
Active and non-active users
No. of comments,reply,threads
Network Graph
Level of Activity
Trends
Qualitative Analysis

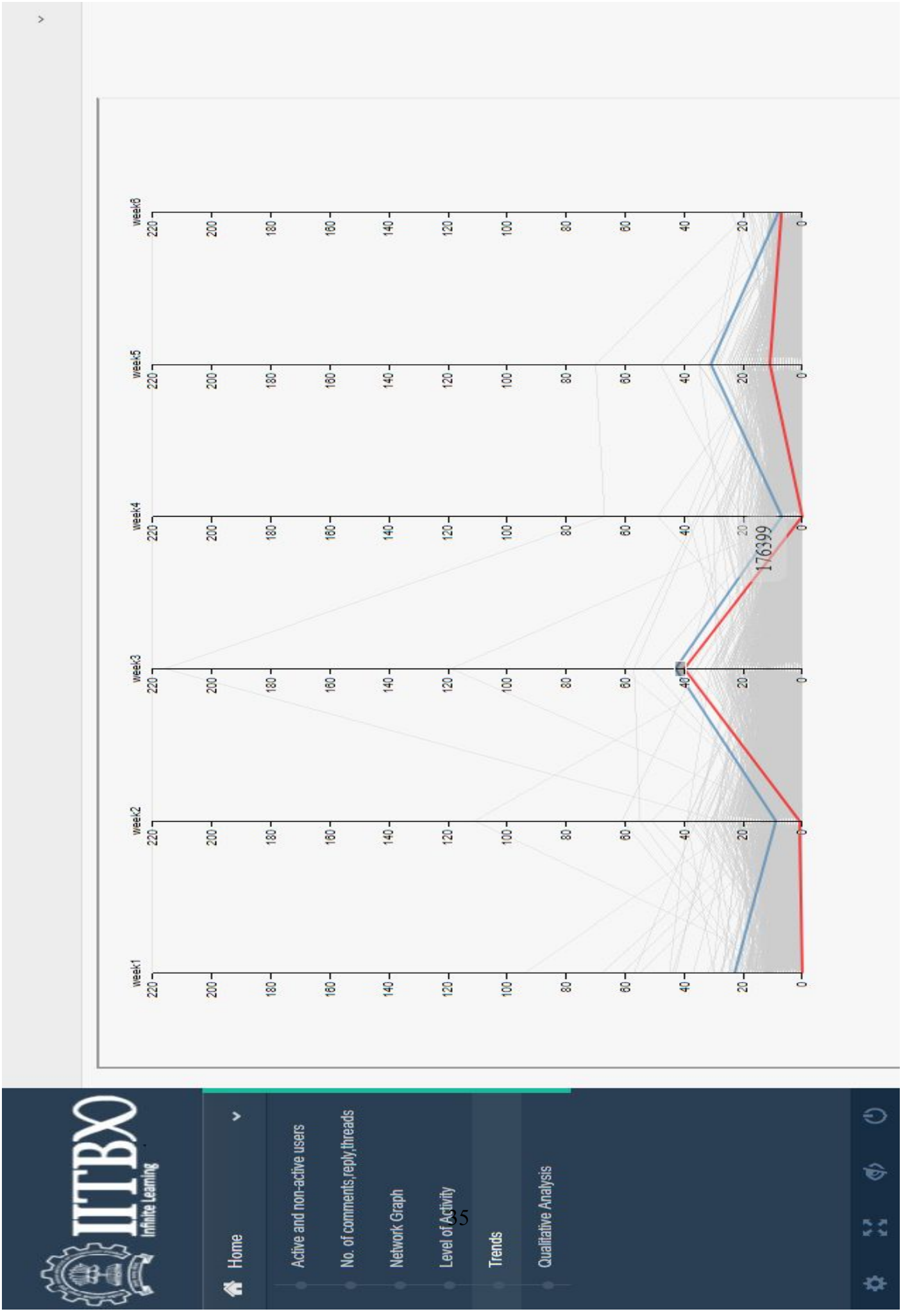
3. Trends Graph



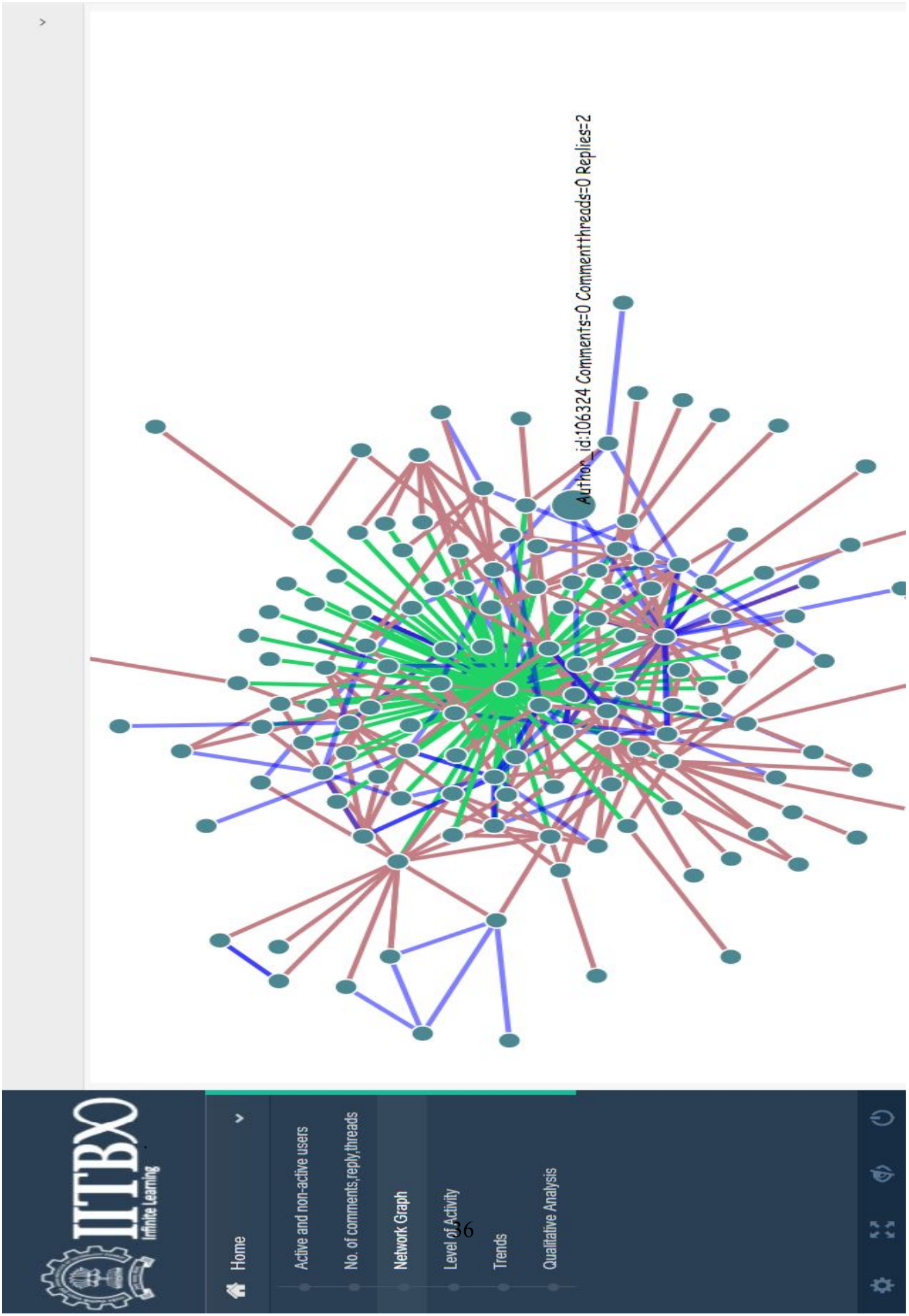
4. Transition Graph (displaying all)



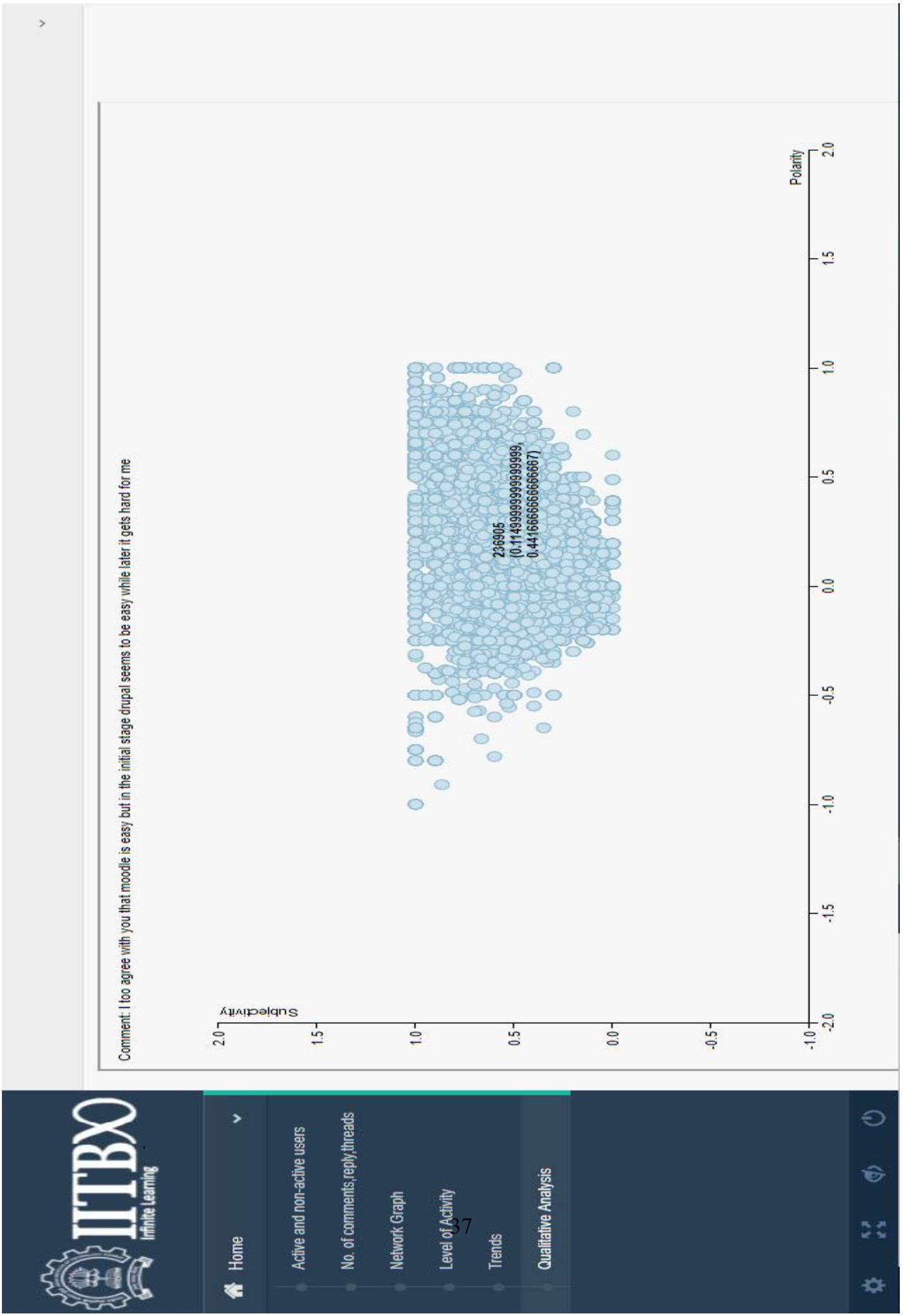
5. Transition Graph (displaying selected)



6. Network Graph



7. Scatter Plot



Chapter 7:

Testing

Test Case ID	Objective	Steps/Descri-tion	Input	Expected Output	Actual Output	Result	Rema rk
T01	check login page	1. enter userid 2. enter password 3. click on ‘log in’	user_id= purav password= abc123	1.user is navigated to dashboard according to its role. 2.system logs should be updated.	As Expected	pass	

T02	To upload a bsondump of courses	1.user logins and select file upload. 2.choose a file by clicking on browse. 3.click on upload file. **precondition:** 1.file should be of bson type. 2.only admin can upload a file	file=contents .bson	1.file uploaded in database and system log is updated. 2.bson to csv is converted and separate csv files are generated for each course and are stored in database. 3. csv files week wise are also generated.	As Expected	pass	
T03	To download a csv file	1. user logins and all csv files under him/her are displayed to download. 2.right click on csv file and select save link as. **precondition:** instructor can download a file.	No input	1.file is stored in system. 2.system log is updated.	As Expected	pass	
T04	To add new course	1.user logins 2.select add course	course_id= start_date=	1.course will be added in	As Expected	pass	

		from side menu. 3.select course from dropdown box. 4.enter startdate and enddate of course. 5.enter userid for that course. 6.click on add course. **precondition:** only admin will have this function.	end_date= user_id=	database. 2.system log is updated.			
T05	To add new instructor	1.user logins 2.select add instructor from menu. 3.enter userid 4.enter password 5.enter course_id 6.click on 'add' if only one instructor is under that course_id else click on 'add instructor' to enter multiple instructors for that course_id. **precondition:**	user_id= password= course_id=	1.instructor is added in database. 2.system log is updated.	As Expected	pass	

		1.only admin can add instructor 2.at most 10 instructors can be added for each course_id.					
T06	check logout function	1.user logins 2.click on top right arrow and select logout. 3.another option is at bottom-right of page click on logout icon.	No input	1.user will be directed to login page. 2.system log is updated.	As Expected	pass	
T07	To view MIS report	1.user logins. 2.select mis report from menu.	No input	1.Mis Report is displayed i.e system logs.	As Expected	pass	
T08	To view total number of comments, threads and replies	1.user logins 2.click on one of the course then will be directed to analysis page. 3.user can select the option for this analysis. **precondition**: only instructor can view this.	select course_name from list of courses displayed.	1.pie chart will be displayed. 2.when hover on one part number of that part and percentage will be given.	As Expected	pass	

T09	To view active/non-active users	1.user logins 2.click on one of the course then will be directed to analysis page. 3.user can select the option for this analysis. **precondition**: only instructor can view this.	select course_name from list of courses displayed.	1.double bar graph will be displayed. 2.one bar for active users and one for non-active users. 3.x-axis has week number and y-axis has number of discussion.	As Expected	pass	
T10	To view network graph.	1.user logins 2.click on one of the course then will be directed to analysis page. 3.user can select the option for this analysis. **precondition**: only instructor can view this.	select course_name from list of courses displayed.	1.force-directed graph will be displayed. 2.nodes representing author_id. 3.different colours are used to show comment, thread and reply.	As Expected	pass	
T11	To view trends for active/non-active users.	1.user logins 2.click on one of the course then will be directed to analysis page.	select course_name from list of courses displayed.	1. bar chart with 4 layers will be displayed. 2.different	As Expected	pass	

		3.user can select the option for this analysis. **precondition**: only instructor can view this.		colours used for each layer 3.when hover we can get number of users active and non-active.			
T12	To view transition of users weekwise	1.user logins 2.click on one of the course then will be directed to analysis page. 3.user can select the option for this analysis. **precondition**: only instructor can view this.	select course_name from list of courses displayed.	1.transition graph will be displayed. 2.user should be able to select particular author_id to view its transition.	As Expected	pass	
T13	To view sentiment analysis of discussion.	1.user logins 2.click on one of the course then will be directed to analysis page. 3.user can select the option for this analysis. **precondition**: only instructor can view this.	select course_name from list of courses displayed.	1.scatter plot will be displayed having polarity and subjectivity on axis. 2.when hover on node author_id with	As Expected	pass	

				sentiment(polarity,subjectivity) and for which comment should be displayed.			

table:7.1

Chapter 8:

Result and Analysis

We have performed analysis of the course with course_id ET601x. This course ran for 14 weeks.Following are the results of the analysis:-

8.1 Pie chart: Number of comment threads=5266, number of comments=7737 and number of replies=2532.

8.2 Bar chart: Number of active users for week 1=250 and Number of inactive users for week 1=74.

8.3 Trends: For week 1: Inactive users (n1)=1082, partially active users(n2)=129, most active users(n3)=12, discussion drivers (n4)=2.

8.4 Transition graph: author_id = 114782 transitioned from 0 to 30 between week 1 and week 2 and then his participation gradually decreased till week 14 except for week 6 when his participation increased to 20.

8.5 Network Graph: author_id=112259 replies to a comment of author_id=113930.author_id = 117684 has created one thread.

8.6 Scatter plot: author_id= 107701 has negative polarity (-0.5) and is subjective (1) for comment = felt difficult in summarising.

Few instructors tested our system, they gave us few suggestions or ideas which will helpful to understand the analysis.

Chapter 9:

Conclusion and Future Scope

Conclusion

The proposed system will provide valuable insights from discussion forum. The system mainly focuses on quantitative analysis and basic qualitative analysis at this stage. The system has been designed having separate dashboards with required functionalities for Admin and Instructors. Coming to quantitative analysis, various visualisations(piechart, barchart, network graph etc.) has been implemented to provide insights such as number of active/non-active users, number of comments and threads, weekly summary etc to Instructors. Whereas qualitative part has a scatter plot which gives sentiment analysis in the form of polarity and subjectivity of comments.

Future Scope

The discussion can further be analyzed using sentiment and semantic for qualitative analysis such as for extracting most discussed topics or to get most relevant comments etc. As our system currently supports particular format for bson dump in future it can be generalised.

Appendix

1. Mooc: massive open online courses
2. Bson: binary javascript object notation
3. Csv:comma separated values
4. Gui: graphical user interface
5. Ieee: institute of electrical and electronics engineers
6. Acm: association for computing machinery
7. Mallet: machine learning for language toolkit
8. Sql:structured query language
9. Mis: management information system

References

[1] Siwei Fu, Jian Zhao, Weiwei Cui, and Huamin Qu, “Visual Analysis of MOOC Forums with iForum,” IEEE 1077-2626 (c) 2016.

[2] K.-C. Chu, C.-S. Wang, J.-H. Kuo, “Mining of learning interaction patterns in MOOCs forums,” IEEE 978-1-5090-3207-5/16 2016.

[3] Jian-Syuan Wong, Bart Pursel, Anna Divinsky, and Bernard J. Jansen, “An Analysis of MOOC Discussion Forum Interactions from the Most Active Users,” Springer International Publishing Switzerland, pp. 452–457, 2015.

[4] Tongshuang Wu, Yuan Yao, YuqingDuan, Xinzhi Fan, Huamin Qu, “NetworkSeer: Visual Analysis for Social Network in MOOCs,” IEEE Pacific Visualization Symposium, 2016.

[5] Xavier Thayalan, Alice Shanthi, “Qualitative assessment of social presence in online forums,” IEEE 2011.

[6] Yunping Feng, Di Chen, Haopeng Chen, Chengcheng Wan and Puzhao Xi, “The Assessment of the Points Reward Mechanism in Online Course Forum,” IEEE 2016.

[7] P. Suraj, Dr.Kumari Roshni V. S., ” Social Network Analysis in Student Online Discussion Forums,” IEEE Recent Advances in Intelligent Computational Systems (RAICS), 2015.

[8] M. Accorroni and C.A. Bentivoglio, “Supporting Tutoring by Improved Statistical Analysis of Discussion Forum,” IEEE 2009.

[9] Gary K. W. Wong, Elby, W. Y. Wong, Simon Y. K. li, "Analyzing Academic Discussion Forum Data with Topic Detection and Data Visualization," IEEE 2016.

[10] Jovita M. Vytasek, Alyssa F. Wise, Sonya Woloshen, "Topic Models to Support Instructors in MOOC Forums," in *Proc* Proceedings of the Seventh International Learning Analytics & Knowledge Conference, 2017.

[11] Christopher G. Brinton, Mung Chiang, Shaili Jain, Henry Lam, Zhenming Liu, Felix Ming Fai Wong, "Learning about Social Learning in MOOCs: From Statistical Analysis to Generative Model,"IEEE Transaction on Learning Technologies, vol. 7, 2014.

[12] AysuEzen-Can, Shaun Kellogg, "Unsupervised Modeling for Understanding MOOC Discussion Forums: A Learning Analytics Approach," in *Proc* Proceedings of the Fifth International Conference on Learning Analytics and Knowledge, pp 146-150, 2015.

[13]mbostock, Bring data to life with SVG, Canvas and HTML, (2016), .https://github.com/d3/d3 .

[14]Mike Bostocks, force directed graph, (2018), https://bl.ocks.org/mbostock/4062045.

[15]KatirG's, d3.js clickable stacked bar chart , (2018), http://bl.ocks.org/KatiRG/5f168b5c884b1f9c36a5.

[16]Kai's, Parallel Coordinate Ordinal axis, (2015), http://bl.ocks.org/syntagmatic/4020926.

[17]ArpitNarechania, PieChart,(2017), http://bl.ocks.org/arpitnarechania/577bd1d188d66dd7dffb69340dc2d9c9.