Development Time Analysis

[1]Jibhaoo N. Bhadane, [2]Yogesh R. Bhabad, [3]Sumit Sonawane

[1,2,3]UG STUDENT, Department Of Computer Engineering, Late G. N. Sapkal College of Engineering

Nashik, Maharashtra, India.

[1]bhadanejibhaoo@gmail.com, [2]yogeshbhabad1991@gmail.com, [3]sumitsonawane34@gmail.com

Abstract **To illustrate how emotions play a role in project success. This typically occurs as resources are reduced relative to the planned requirements. For example, a project may have six months remaining on its schedule with a projected rate of completion in one year. Facing such a situation, managers may push developers to work harder. Some developers may even enjoy the hero's role as they strive to complete a project under difficult circumstances. Unfortunately, such projects typically fail. Your dan's death march applies mainly to commercial development, because developers are under contract to complete a project. In open-source development, when developers see the beginnings of a death march they are be more likely to simply abandon the project. The death march illustrates an interesting issue about software development projects: due to the dynamic structure of the teams, and the unique ways of collaborating, projects can evolve substantially. Thus, analyzing a static snapshot does not allow us to understand an evolving trajectory.**

Keywords: Behavior, Frustrated, spam, Social network.

I. INTRODUCTION

This is a system in which we have to analyze the behavior of the user who is the active user on the social network. If any user continuously keep the sad status and wants to talk with other friends on the network. Then our system analyze the behavior of the user is frustrated. User will upload the post, then system will categorize the post whether spam or not. The will track the users login and logout time for active user on the network. According to time if user login at midnight then user in tension or having more workload. The projects were also mined for developer sentiment. Finally, a regression model shows how sentiment varies with behavioral differences a change in behavior is correlated with a change in sentiment we have analyze each projects event sequences over time, we aggregate each projects behavior and sentiment data to create a multiple project regression, with behavior predicting sentiment.

This is a system in which we have to analyze the behavior of the user who is the active user on the social network. A multi-project level regression supports the conclusion that routine change is positively associated with sentiment change.

To illustrate how emotions play a role in project success. This typically occurs as resources are reduced relative to the planned requirements. For example, a project may have six months remaining on its schedule with a projected rate of completion in one year. Facing such a situation, managers may push developers to work harder. Some developers may even enjoy the hero's role as they strive to complete a project under difficult circumstances. Unfortunately, such projects typically fail. Yourdan's death march applies mainly to commercial development, because developers are under contract to complete a project. In open-source development, when developers see the beginnings of a death march they are be more likely to simply abandon the project. The death march illustrates an interesting issue about software development projects: due to the dynamic structure of the teams, and the unique ways of collaborating, projects can evolve substantially. Thus, analyzing a static snapshot does not allow us to understand an evolving trajectory. Evolutionary qualities are important, and may be a good guide to forecasting future qualities, such as project sentiment, followers, downloads, and eventual success This research project is aimed at understanding the relationship between developer behavior and developer sentiment. This is part of the larger project that aims to create a developer behavioral-model to predict project characteristics, such as

success. The ultimate goal is to provide automated support for project managers. As development occurs, the predictive model is continually updated and its results presented on a project dashboard. Raising alerts on a project dashboard when a team appears to be losing its effectiveness is a goal of the work described herein. We rely on the team's repository to observe their activities. Our data mining techniques provide for automated analysis, which can be incorporated into the dashboards common to Agile development.

An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, response time, availability, portability, maintainability, footprint, security and speed of recovery from adverse events are evaluated.

II. LITRATURE SURVEY

A. Diversity in Software Development Routines are Attractive

A Preliminary Analysis of GitHub Repositories. Introduction libre, open-source software projects (FLOSS) are known for their chaotic development style and unique collaboration model. How does such chaotic development produce high quality software and attract users and developers? To provide insight into this conundrum, this study explores the roles of diversity and change in design routines. It investigates the relationship between routine diversity and change on project attraction to users and developers.

B. Free/Libre Open Source Projects: A Longitudinal Study

The ability to attract resources is an indicator of the perceived project legitimacy, which in turn is a strong predictor of the projects future sustainability. Thus a projects ability to attract developer and user resources is shown to play a mediating role between the demographic and ecological (niche) characteristics of the project and its future sustainability. Our results support the applicability of tenets of organizational ecology related to the liability of smallness, the liability of newness, and population characteristics (niche size) to the FLOSS development environment. The implications of the results for future research and practice are discussed.

C. Socialization in Open Source Software Projects: A Growth Mixture Modeling Approach

The success of open source software (OSS) projects depends heavily on the voluntary participation of a large number of developers. To remain sustainable, it is vital for an OSS project community to maintain a critical mass of core developers. Yet, only a small number of participants (identified here as joiners) can successfully socialize themselves into the core developer group. Despite the importance of joiners socialization behavior, quantitative longitudinal research in this area is lacking. This exploratory study examines joiners temporal socialization trajectories and their impacts on joiners status progression. They also found that these distinct latent classes of joiners varied in their status progression within their communities. The implications for research and practice are correspondingly discussed. Keywords latent class analysis, latent class growth models, latent growth models, longitudinal data analysis.

III. SYSTEM ARCHITECTURE

This is a system in which we have to analyze the behaviour of the user who is the active user on the social network. If any user continuously keep the sad status and wants to talk with other friends on the network. Then our system analyze the behaviour of the user is frustrated. User will upload the post, then system will categorize the post whether spam or not. The will track the users login and logout time for active user on the network. According to time if user login at midnight then user in tension or having more workload.
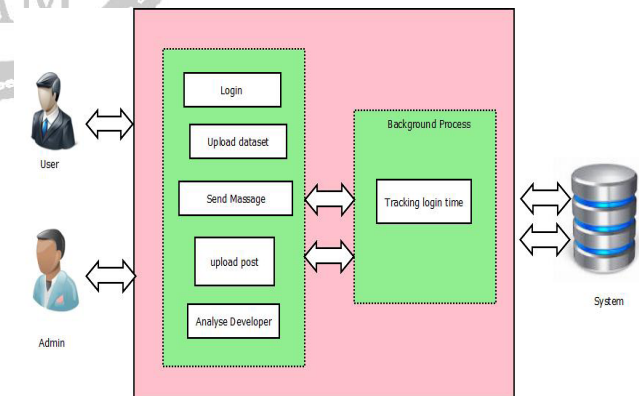

Fig. 1 Architecture diagram

Vulnerability Test Module is a module in which crawling of the website, preparing test cases and analyzing the response that is sent by the web server is done. To check whether SQL injection attacks are possible, the vulnerability scanners send modified requests and analyze the responses returned by the server. A server may respond with a rejection page or with an execution page. A rejection page

corresponds to the detection of syntactically incorrect or invalid inputs. An execution page is returned by the server as a consequence of a successful execution of the request. This page legitimate use of the web site, but may also result from a successful exploitation of an injection attack.
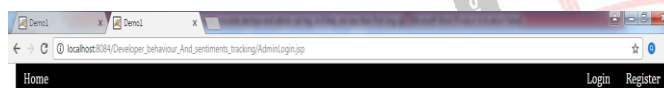
IV. RESULT ANALYSIS

A. Home Page

Fig. 2 Home Page

This is first module in this module developer and admin can log in if they are new then first sing up They are create their own password.

B. Admin Module

Fig.3 Admin Module

The above module are shows the Admin login. The admin are securely login in this module we insert the username and Password.

C. Registration
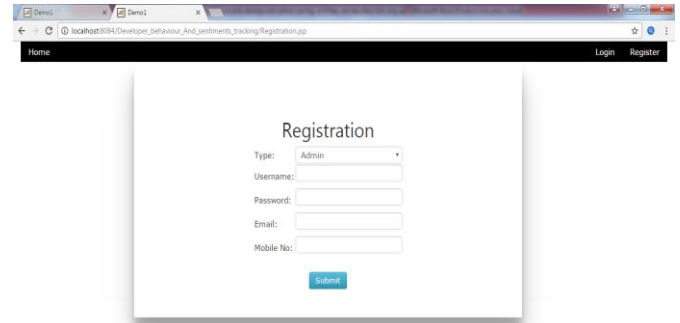
Fig. 4 Registration

The above module are shows the Registration form. In this module are register the new admin & new user.
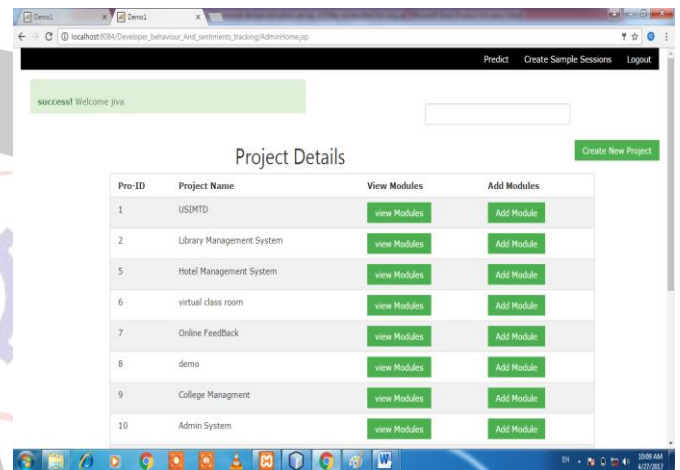
D. Project Details

Fig 5 Project Details

These is admin module admin can log in and assign different project to different developers Also admin view project and add new project and create new project.
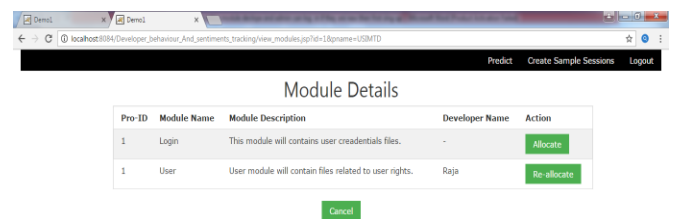
E. Module Details

Fig. 6 Module Details

The above module are shows the module details. These are shows the module are allocate the developer and these Status.
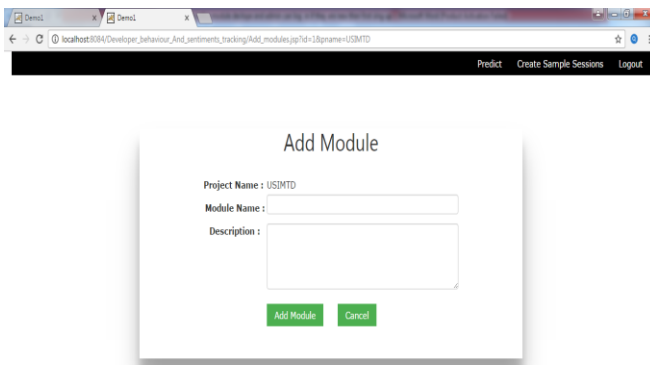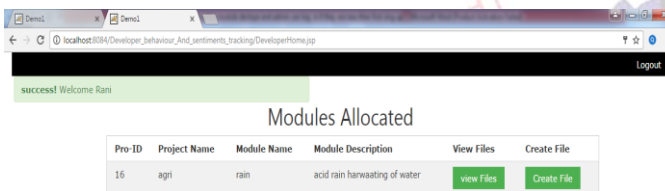
F. Add New Module

Fig. 7 Add module

The above Module are shows the add module. These module are shows the add the new module and these description.

G. Module Allocation

Fig. 8 Modules Allocation

In these module developer can see project id, project name, module name, module description

V. CONCLUSION

This is a system in which we have to analyze the behaviour of the user who is the active user on the social network. If any user continuously keep the sad status and wants to talk with other friends on the network. Then our system analyze the behaviour of the user is frustrated . User will upload the post, then system will categorize the post whether spam or not. The relationship between sentiment and success is not

directly explored, herein. This research project is a preliminary step in a larger research project aimed at understanding and monitoring FLOSS projects using a process modeling approach.in this project aimed to Analyse Developer Behaviour.

REFERENCES

[1] Chengalur-Smith, A. Sidorova, and S. Daniel, "Sustainability of free/libre open source projects: A longitudinal study," Journal of the Association for Information Systems, vol. 11, pp. 657-683, 2010.

[2] Samoladas, L. Angelis, and I. Stamelos, "Survival analysis on the duration of open source projects," Information and Software Technology, vol. 52, pp. 902-922, 2010.

[3] E. Yourdon, Death march: Pearson Education, 2003.

[4] W. N. Robinson, A. Akhlaghi, and T. Deng, "Transition Discovery of Sequential Behaviors in Email Application Usage Using Hidden Markov Models," in Hawaii International Conference on Software Systems, HI, USA, 2013, p. (best paper nominee).

[5] W. N. Robinson, A. R. Syed, A. Akhlaghi, and T. Deng, "Pattern Discovery of User Interface Sequencing by Rehabilitation Clients with Cognitive Impairments," in Hawaii International Conference on Software Systems, HI, USA, 2012.

[6] W. N. Robinson, A. Akhlaghi, A. Syed, and T. Deng, "Discovery and Diagnosis of Behavioral Transitions in Rehabilitation Patient Event-Streams," ACM Transactions on Management Information Systems (TMIS), 2012.

[7] S. W. Hansen, W. N. Robinson, and K. J. Lyytinen, "Computing Requirements: Cognitive Approaches to Distributed Requirements Engineering " in Hawaii International Conference on Software Systems, HI, USA, 2012.

[8] W. N. Robinson, "A Roadmap for Comprehensive Requirements Monitoring," IEEE Computer, vol. 43, pp. 64-72, May 2010.

[9] W. N. Robinson, "A requirements monitoring framework for enterprise systems," Requirements Engineering Journal, vol. 11, pp. 17-41, 2006.

[10] A. H. Van de Ven, Engaged Scholarship: A Guide for Organizational and Social Research: A Guide for Organizational and Social Research: Oxford University Press.

AUTHORS PROFILE

1 st Author Name :- Jibhaoo Bhadane **Qualification** :- Diploma in Computer Engg from Mumbai University, B.E Appear of computer Engineering department from Late G.N College Of Engineering Savitribai Phule Pune University.

2 nd Author Name :- Yogesh Bhabad **Qualification** :- Diploma in Computer Engg from Mumbai University, B.E Appear of computer Engineering department from Late G.N College Of Engineering Savitribai Phule Pune University.

3 rd st Author Name :- Sumit Sonawane **Qualification** :- Diploma in Computer Engg from Mumbai University, B.E Appear of computer Engineering department from Late G.N College Of Engineering Savitribai Phule Pune University.