Tracer for Estimation of the Data Changes Delivered and Permalinks

N. H. Prasad, S. Kavitha, Laxmi Narayana, and G. R. Sanjay

1 Introduction

Source Code Management (SCM) plays an important role in organizing, managing, and controlling the changes delivered by the developers to the source code, documents, and other entities. In LGSI, there is no system that can track the changes delivered by the various stakeholders. The Change Tracker system aims to allow the developers to create a list of interesting files and send notifications to the intended stakeholders if any changes are committed and also the committed changes need to be reviewed by the reviewer. The proposed system is intended for use within LGSI to keep track of changes delivered by the LGSI developers. In addition, it is helpful in keeping stakeholders in communication by sending notifications when any new change is committed. This scope of the work can also be extended to other domains, like the TV team managed by the service provider. Users need not have to worry about the services or management. The best example is the web-based email service. The change Tracker system has been developed with Agile methodology. Agile methodology is popular for its nature of incorporating change in the industry with Agile frameworks such as scrum, Kanban, and extreme programming. The main components of the Scrum framework are organizing small teams, daily scrum meetings to know the status, making sprint planning and review, collaboration with other teams, and meetings with stakeholders. At first, the initial requirement of the Change Tracker system is to track the changes made to code by each developer in later stages based on discussions happened with the product manager and the internal stakeholders; the other requirements are

N. H. Prasad · S. Kavitha (✉) · L. Narayana · G. R. Sanjay
Department of Masters of Computer Applications, Nitte Meenakshi Institute of Technology, Bangalore, India
e-mail: kavitha.s@nmit.ac.in; narayan@nmit.ac.in

A. Haldorai et al. (eds.), *5th EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, EAI/Springer Innovations in Communication and Computing, https://doi.org/10.1007/978-3-031-28324-6_5

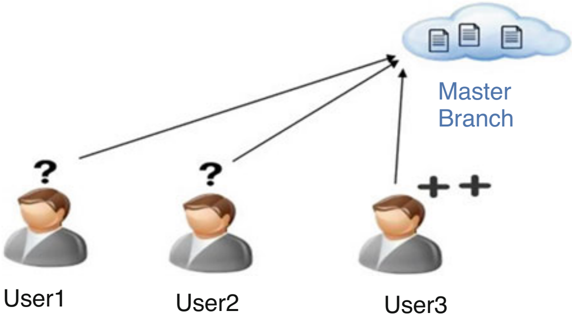
captured such as displaying analytics, exporting permalinks details to excel, and other functionalities. The feedback from various stakeholders play important role in developing a Change Tracker system. The proposed methodology Change Tracker is used in an IT security and compliance management platform. The Change Tracker benefits enterprises track and authenticate changes to the performance of devices, files, and configurations. The important features in Change Tracker are agentless integrity monitoring and intrusion detection.

2 Literature Survey

In SDLC, the same file is modified by many developers from different teams and geographical locations [1]. The latest version of the file may undergo several changes by the developers (Fig. 1). It is very difficult to identify who made the changes to a specific file and also difficult to find or recover accidentally lost files. This problem will directly affect project deliverables and may cause delays in identifying the problem and resolving issues [2]. In addition, the current system is not notifying the developers [3]. The advantages of the protractions are diminished when whiskers make contact with objects, according to the author [4], which causes whiskers to have a tendency to only barely touch the environment. There has been research on how this mechanism affects sensory input, but less is known about how sensory input alters the motor pattern [4].

Currently, there is no system to notify other developers who previously modified the same file, and there is no system to track developer's action when exaggerated past designations file modified by others. Introducing changes accidentally may cause project risk.

Fig. 1 Existing system

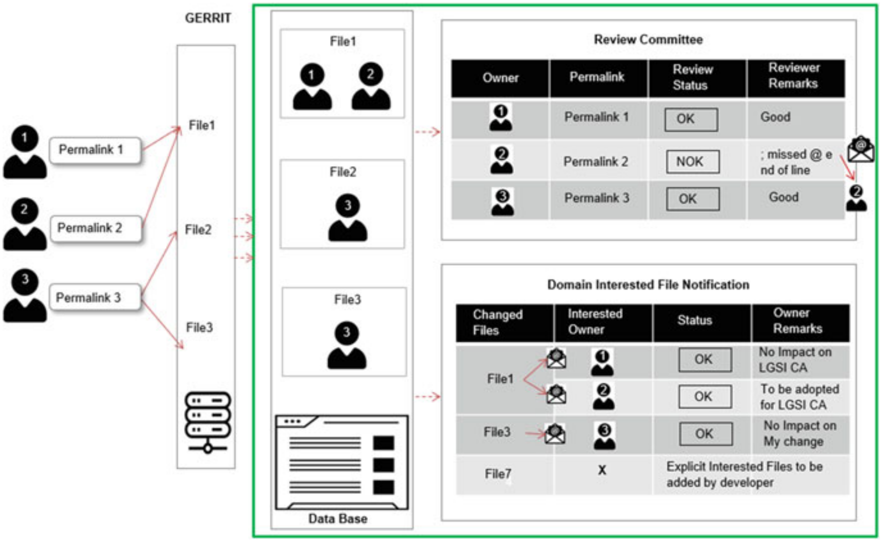


3 Proposed System

The proposed system is capable of tracking changes delivered by the LGSI developers and enables communication between respective stakeholders. The proposed system is capable of tracking changes delivered by the LGSI developers and enables the communication between respective stakeholders. The Change Tracker tracks the files delivered by all users. It keeps track of the users interested in a file. Once a particular file is modified, Change Tracker notifies all the developers interested in that file. PL can track LGSI applicability on developer interested file. He proposed system facilitate developers to create list of interested Files. The Change Tracker tracks the files delivered by all users. It keeps a track of the users interested in a file. Once a particular file is modified, Change Tracker notifies all the developers interested in that file. The study is conducted before developing the Change Tracker application and analyzed the facts that will affect the project completion. In this study, we identified the technology stack essential for developing this Change Tracker system, such as HTML, CSS, Bootstrap, and JavaScript for the front end and Java, JSP, MySQL, and Java Servlets for the back end and also identified the organization has a strong technical team with essential skills and has the necessary hardware requirements. The development of this application is possible within the threshold budget for the entire project; the costs will include hardware and software costs, development costs, and operational costs. The Change Tracker application can be developed without any extra cost to the organization as this project can be carried out with the existing infrastructure of the organization [5]. This application is beneficial for the organization as it can track changes made to the project and avoids any accidental changes and ensures the software quality which will be beneficial for the organization in terms of finance. The proposed system facilitates developers to create a list of interested files [6].

4 Methodology

In the recent years, study has been conducted to map the behavior of components of complex systems and to clarify material flow and mixing properties. Although research groups have established systems to track rodents through video, with the exception of [7], prior approaches all need a secured computer for computation. In software engineering, software engineering researchers have provided many software development methodologies that fit or are adapted by various industries' Software solutions. During the initial stages of the project Change Tracker, there were various methodologies planned, and the Agile methodology was best suited for this project and worked accordingly. In recent years, the organization faced more difficulties as there is no system or software application for tracking permalinks and all their information. Therefore, the organization planned such an application that adapted one of the software development methods, agile development (Fig. 2). The



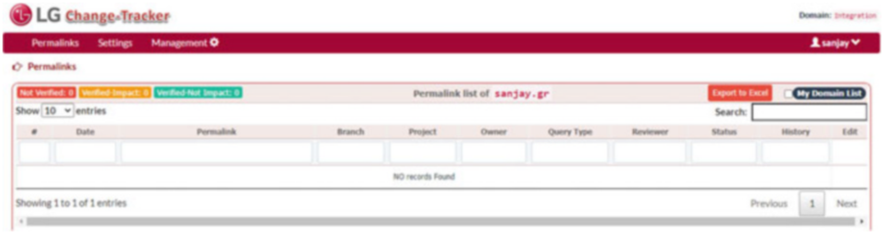
Review Committee

Owner	Permalink	Review Status	Reviewer Remarks
1	Permalink 1	OK	Good
2	Permalink 2	NOK	; missed @ end of line
3	Permalink 3	OK	Good

Domain Interested File Notification

Changed Files	Interested Owner	Status	Owner Remarks
File1	1	OK	No Impact on LGSI CA
	2	OK	To be adopted for LGSI CA
File3	3	OK	No Impact on My change
File7	X	Explicit Interested Files to be added by developer	

Fig. 2 Architecture of Change Tracker


LG Change-Tracker

Domain: Integration

Permalinks Settings Management

sanjay

Permalinks

Not Verified: 0 Verified-Impact: 0 Verified-Not Impact: 0

Permalink list of sanjay.gr

Export to Excel

My Domain List

Show 10 entries

Search:

#	Date	Permalink	Branch	Project	Owner	Query Type	Reviewer	Status	History	Edit

NO records Found

Showing 1 to 1 of 1 entries

Previous 1 Next

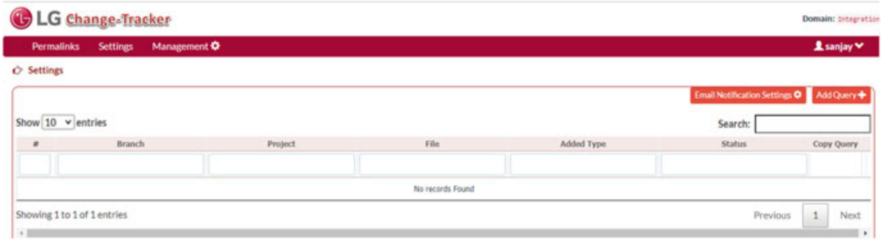
Fig. 3 User Permalink

agile development methodology is the simplest form. As a result of this concern, the advantages of agile software development [8] are that the organization is capable of reducing the overall risk which is associated with the Change Tracker tool.

The user can view records consisting of permalinks (Fig. 3) details created by all the developers. The internal stakeholders of the Change Tracker system can export the record of permalink details, including information such as date, permalink, branch, project, owner, query type, reviewer, status, and history.

In the settings, users can view records consisting of details created by the developers. They include information such as entities branch (Fig. 4), project, file, added type, status, copy query, and add query. Type is maintained to be application and state should be active [9].

The Change Tracker system must enable communication between the admin, reviewer, and user through email notifications (Fig. 5).



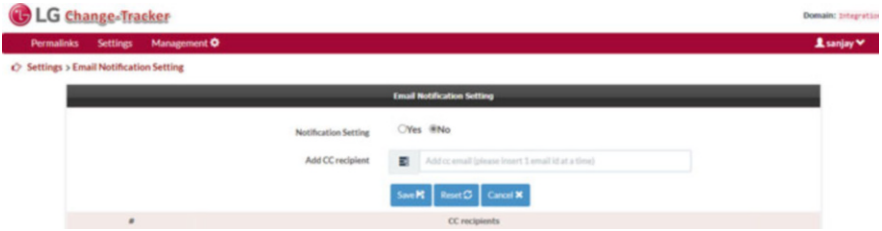
Show 10 entries

Search:

#	Branch	Project	File	Added Type	Status	Copy Query
No records Found						

Showing 1 to 1 of 1 entries

Fig. 4 Record view

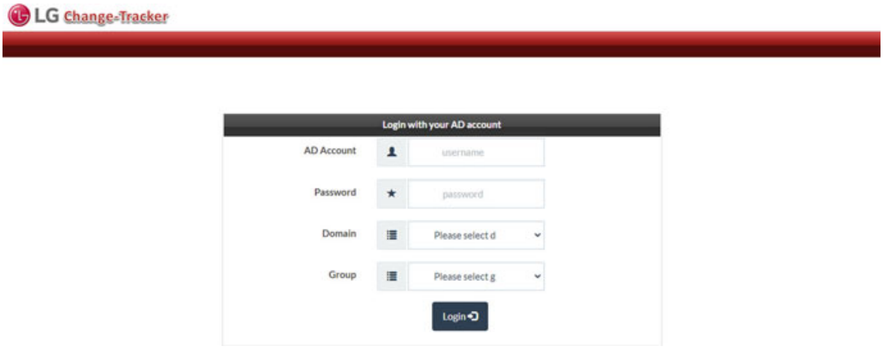


Notification Setting ○ Yes ● No

Add CC recipient: Add cc email (please insert 1 email id at a time)

#	CC recipients

Fig. 5 Email notification setting



AD Account: username

Password: password

Domain: Please select d

Group: Please select g

Fig. 6 Login to the module

5 Results and Discussions

After deploying the site on the elastic instance, we obtain the URL and visit the website that we deployed [10]. We then enter our credentials (Fig. 6) to log in, as the users are authenticated functions.

Here, the admin (Fig. 7) can post the permalink for other users to access and read them. This is how we can see the permalinks.

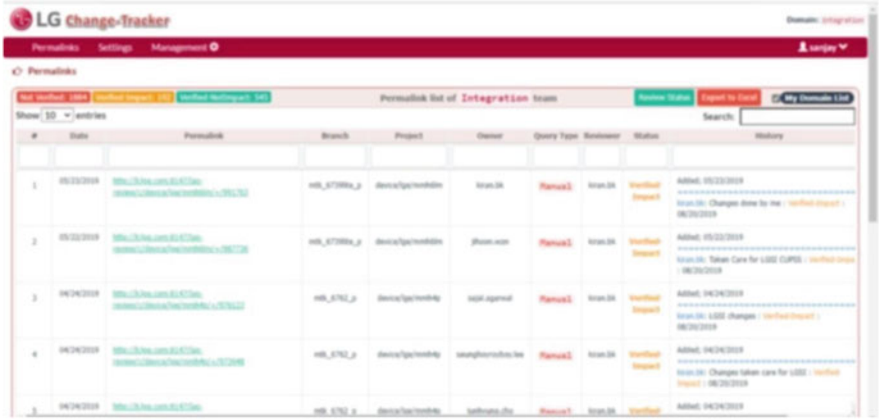


Fig. 7 Dashboard of permalink

6 Conclusion

In this paper, we have proposed the Change Tracker system to address one of the major problems faced by the LGSI employees in identifying who modified the code, removed files during software development. The Change Tracker system allows every individual developer to keep track of changes delivered to his/her interested files and creates permalinks delivered by LGSI developers then the permalinks will be reviewed by the review task team. In addition, Change Tracker notifies the respective stakeholders when the changes are made. Thus, this application helps in taking care of any risk associated when a new change is introduced or modified. In the future, the organization plans to enhance the proposed system by incorporating features like all DevOps activities such as tracking android security patches, domain patches also the Change Tracker system into a single application as its future goal.

References

1. J.M. Vara Mesa, ATL/AMW use case modeling Web applications: Detailed description and user guide. *International Journal of Software Engineering and Its Applications*, **5**(2), 2011 (2009)
2. S. Al-Fedaghi, Scrutinizing UML activity diagrams, in *17th International Conference on Information Systems Development*, Paphos, Cyprus, 25–27 Aug 2008
3. W. Huang, R. Li, C. Maple, H. Yang, D. Foskett, V. Cleaver, Web application development lifecycle for small medium-sized enterprises (SMEs), in *Proc. of the Eighth International Conference on Quality Software*, (IEEE, 2008), pp. 247–252
4. J. Voigts et al., Tactile object localization by anticipatory whisker motion. J. Neurophysiol. **113**(2), 620–632 (2015)

5. J. McCurley, D. Zubrow, C. Dekkers, *Measures and Measurement for Secure Software Development* (Carnegie Mellon University Build Security In, 2008)
6. H. Vikas, Overview of lift web framework, in *Presented at the 4th IndicThreads.com Conference on Java*, (Pune, 2009)
7. M.A. Nashaat, H. Oraby, L.B. Peña, S. Dominiak, M.E. Larkum, R.N.S. Sachdev, Pixying behavior: A versatile real-time and post hoc automated optical tracking method for freely moving and head fixed animals. eNeuro **4** (2017). https://doi.org/10.1523/ENEURO.0245-16.2017
8. F. Baharom, A. Deraman, A. Hamdan, A survey on the current practices of software development process in Malaysia. J. Inf. Commun. Technol. **4**, 57–76 (2006)
9. M.P. Papazoglou, D. Georgakopoulos, Serviced-oriented computing. Commun. ACM **46**(10), 25–28 (2003)
10. B. Benatallah, *Web Services: Life Cycle Intelligence, Power Point Presentation, School of Computer Science and Engineering* (The University of New South Wales, 2006)