

Link 1: <file:///E:/Assignments/S.E%20Assignment/Papers/1st%20paper.pdf>

1. Employing Contribution and Quality Metrics for Quantifying the Software Development Process

Summary:-

Title:

Employing Contribution and Quality Metrics for Quantifying the Software Development Process.

Authors:

Themistoklis Diamantopoulos (Electrical and Computer Engineering Dept, Aristotle University of Thessaloniki, Greece), Michail Papamichail, Thomas Karanikiotis, Kyriakos Chatzidimitriou (Aristotle University of Thessaloniki, Greece), Andreas Symeonidis (Aristotle University of Thessaloniki, Greece).

Conference Name:

The 42nd International Conference on Software Engineering ICSE 2020
(6-11 July 2020).

Published:

ICSE 2020.

Introduction:

As we know that conventional method of software development has been changed with the latest method software development technique called is open-source platform such as GitHub, Bitbucket or GitLab. As compared with old technique of software development with latest one as more transparent and very scalable. Addition features also give us these open-source platforms that entire code of software are disperse in whole team member which are presented in different part of the world. These software is not only provide us the code management but also give us many other features like collaboration, versioning, backup and restore features also. In this research paper we built a platform that crawls the infrastructure of GitHub, analysis entire contributions data and evalutes that data and give us answer multiple research questions relevent to the software development process.

Motivation Of The Paper:

GitHub has a very large repository of data and has multiple aspects are hidden in different metrics so we initiate to further dig up that data with different our creating tools(Data downloader, MongoDB, Contributions Analyzer and Quality Analyzer) and find out the researchers question's answers.

Research Methodology:

One could study the correlation between issues and GitHub stars/forks [5], explore the effects of issue labeling on the time required to resolve them [11] , or even determine with resolving issues as early as possible is beneficial for the project [6]

.

Results:

Basically the result of the paper gives us that produce more and more useful metrics of the software development aspects and also give the answers of the researcher's questions. It also gives the message that we create more repositories and covered more scenarios and languages to find out the more scenarios of the hidden metrics of the GitHub platform data.

Link 2: <file:///E:/Assignments/S.E%20Assignment/Papers/2nd%20paper.pdf>

2. Capture the Feature Flag: Detecting Feature Flags in Open-Source

Summary:-

Title:

Capture the Feature Flag: Detecting Feature Flags in Open-Source.

Authors:

Jens Meinicke (Carnegie Mellon University), Juan Hoyos (Universidad Nacional de Colombia, Colombia), Bogdan Vasilescu (Carnegie Mellon University, United States), Christian Kastner (Carnegie Mellon University, United States).

Conference Name:

The 42nd International Conference on Software Engineering ICSE 2020
(6-11 July 2020).

Published:

ICSE 2020.

Introduction:

Feature flags continuously having important in software development but having controversial in continuous development and delivery. Technically, feature flags apply conditionally on code path and decision depends on external configuration mechanism. The practical study is very less on this topic. The open-source software development wealth is very large but feature flags applies on open-source systems are very less. According to this work the feature flags divert from custom development to commercial development. In this paper feature flags based on textual analysis. Technique applied for GitHub public repositories. We study with small quantity of data and increases with the time in future. In our practice the small amount of flags can be effective as compared with the large amount of flags. Finally the paper goals is to find out the ways to apply the feature flagging on a open-source projects.

Motivation Of The Paper:

Feature flags are not worldwide applied on the different nature of the projects and we do not know about the outcome results after applying the feature flagging so our study motivational aspects are to apply different open-source projects and mining software repository of future flags.

Research Methodology:

Feature flags are a topic frequently discussed by practitioners in blog posts and at practitioner conferences [2, 3], provide an extensive overview of grey literature on feature flags [7] . To the best of our knowledge, the only study on feature flags

using mining software repository techniques is the analysis of feature flags in Chromium, the open-source implementation behind Google Chrome [9, 10].

Results:

It is very difficult to identify that we are implement feature flags on entire repositories, so we explain semi-automated process on limited dataset of 100 open-source feature flagging projects and also their individual flags and their lifecycle. The research explain us faster flag clean up and also control pending confirmation experimental methods. Also our dataset gives practitioners to qualitative studies in future and also gives technical evidence-based guidelines for effective practices.

Link 3: <file:///E:/Assignments/S.E%20Assignment/Papers/3rd%20paper.pdf>

3. Threat modeling: from infancy to maturity

Summary:-

Title:

Threat modeling: from infancy to maturity.

Authors:

Koen Yskout (imec – DistriNet, KU Leuven), Thomas Heyman (Toreon), Dimitri Van Landuyt (Katholieke Universiteit Leuven), Laurens Sion (imec – DistriNet, KU Leuven), Kim Wuyts (imec – DistriNet, KU Leuven), Wouter Joosen (Katholieke Universiteit Leuven).

Conference Name:

The 42nd International Conference on Software Engineering ICSE 2020

(6-11 July 2020).

Published:

ICSE 2020.

Introduction:

The purpose of the threat modeling is to identify the risk flows, mitigate and identify the security flows and also implementation of the entire process. Microsoft is basically implement them past years in many industries and academics. The paper showing that there is no practically implementation but just only practice apply both on academic and industries. The thesis depends on the handshaking between practitioners and researchers and also improvements in high level research.

Motivation Of The Paper:

We studied many research paper related to the topic and find out that threat modeling is only depends upon practically only but there is no research work on it and we found that there is no relation between practitioners and researchers. And also we studied the COBIT and CMMI models and inspired from those. Depending on those two models we research further and proposed multi-levels for threat modeling.

Research Methodology:

Threat modeling is very typical research work so we starting from sketching the entire process modeling as practiced by earlier Toreon. Research focus on

academic literature because it is very less practical implemented in this field. According to our goal set we having many challenges to accomplish it.

Results:

After throughly research work we proposed five levels of threat modeling. Currently available model is initial level 1 model and has no standardized process model. To develop a reference model and implementation academic for benchmaking. Every process has evolve another process and built upon another process. The created process is also well-defined manner. Also results of the proposed model being correct and practically measurable. Entire process are optimized and if we put their feedback process then it qualify the quality assurance process. Because sketched research work is very challenging process so it will take time to practically implementation in maturity level and also practically engineering descipline.

