Faculty of Engineering
Unit for Energy and Technology Systems

Research Proposal for MEng/MSc or PhD study

Student Information

NWU Number Initials & Surname	**25899880, C. Scheepers**
Qualification Direction Option	**M.Eng. Computer Engineering**
Research Group	**CRCED Pretoria**

Study leader/Promotor Details

Study leader / Promotor	**Dr Jan Vosloo**
Co-Study leader / Co-Promotor	

USER-BASED ACTIVITY LOGGING AND ANALYSIS TO IMPROVE SYSTEM MAINTENANCE

Keywords: Software maintenance, event logging, logging mechanisms, user activities, data-driven decision making, system utilisation analysis, Web-based

1 Introduction

1.1 Background

Maintenance of software systems is continuous and is a reduced form of software development. [1] Most companies will strife to increase their digital products and services over the life cycle of the software project. [2], [3]

Software systems are complex and inefficient as the need of new development increases. Human and software resources are limited or ineffective. [4] It is expensive to allocate new resources to software maintenance without a cost model for software maintenance. [3]

A lot of time needs to be invested in software maintenance to build an effective model to implement and it makes up to 15% of the development cost. [5] Maintaining software systems will receive lower development priority as the need for new functionality for the software project increases. [1] Unused or software not meeting the user's requirements will increase over the life cycle of the software project. [6]

It is difficult to track unused software systems that can be deprecated. Some parts of the software can be removed or revealed to address the needs of the user. [7], [8] Tracking the behavioural patterns of the user. It may be that the user is uninformed of how to use the system or the developers did not design the system to meet the user's requirement specifications. [9], [10]

Tracking any user activities can be strenuous task depending on the method that is used get the data. The use of questionnaires is will not be viable solution as the software systems may be too large and diverse to get any meaningful feedback from the user. [9], [11] Using logs to track the user's activity gives more valid data of how the user is interacting with the software systems. [12]

The event logging is used for maintenance to collect data of the behaviour of software systems. Event logs are useful for: [13]

- Debugging of software systems
- Failure analysis and system recovery
- To get key behavioural information of individual software components
- System utilisation
- Users contribute to initiating events in software systems

This can be used to track user activities to make data-driven decisions of how to maintain the software systems as such as Web-based systems. [14] [15], [12]

1.2 State of the Art

Several studies have been is obtained for user-based activity tracking where five key components are identified that is relevant to the topic:

- **Log analysis** is a data mining process focused on the analysis of computer-generated records. [16], [4], [17]
- **Logging implementation** by using different types of logging mechanisms for event logging. [18], [14], [19], [20]
- **Logging points** generated like database transactions. [18], [21]
- **Pattern discovery** of user behavioural activities in the event logs. [22], [9], [23], [24]
- **System utilisation analysis** based on the user behavioural activities of these event logs. [13], [9], [23], [10]

In Table 1 the above components is used to identify which studies are relevant to user-based activity tracking and analysis. Red indicates if the study is less relevant and green indicates if the study is relevant to the topic.

Table 1: State of the Art

Source	Logging Analysis	Logging Implementation	Logging Points	Pattern Discovery	System Utilisation Analysis
[1]					
[6]					
[13]					
[14]					
[16]					
[18]					
[19]					
[21]					
[23]					
[25]					
[26]					

In Table 1 the previously discussed topics can be split into main topic which is the logging user activities and system utilisation by analysing the obtained logs. Some the sources proposed data-driven decision making models to improve software maintenance.

1.3 Need for the study

No study was found where both the logging mechanism and analysis were combined. There is a need to develop a method to implement an user-activity logging mechanism to do further analysis of the logs to improve software maintenance.

2 Development of solution

To create a solution to the problem the solution is split into two parts. The first part that consists of creating or updating the logging mechanisms will focus on ensuring that the needed logs are obtained from the software system.

The second part will use the generated logs to create a system utilisation framework of how to further improve software maintenance. The flowchart in Figure 1 shows development of both parts of the solution.

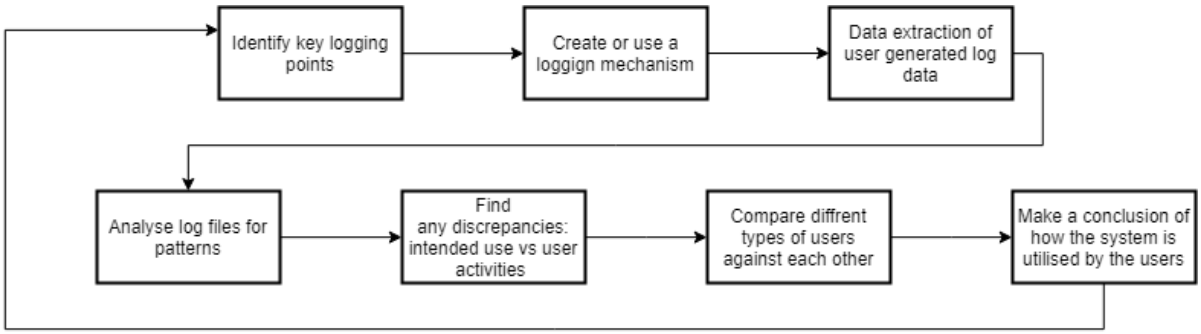

Figure 1: User activity tracking development

2.1 Logging mechanisms for the software environment

The logging mechanism will need to meet the requirements of the second part the solution. Before starting to create the logs, key logging points will be identified. This will from the basis of how the logs will be generated and obtained from the activities of the user.

For a Web-based system the activities will be in the form of which pages the users access and how they use it like clicking on a button or filling in a input text box. These events can be tracked and logged within the system in a structured database.The events that will be tracked will only be in the form of any user input that sends request to the server.

2.2 System utilisation framework to improve software maintenance

The logs will be obtained and filtered for certain key events as there can be large amount of logs generated by every event that is being tracked. There will exists certain patterns of how the users interact with the system. This can be useful to benchmark different types of users against each other like the developers against the average user. This will give valuable data if the system is used as intended.

The logs will be further analysed to get key information about how the system is utilised by the users. Some menu's may be unused or have significantly high traffic. [24] A data-driving decision model can be made to prioritise maintenance on these menu's or which of them can be deprecated if the menu is underused and not necessary for the system. Prioritising maintenance will increase the user experience and software maintenance. [9]

3 Expected results

The results can also be divided into parts. The logs that is generated from the logging mechanism and the results of the system utilisation analysis.

3.1 Logging mechanism results

The result from the logging mechanism will show whether the key logging points and data is obtained. This will be stored in MySQL table and the following key data will be logged:

- **Timestamp** when the activity took placed.
- **Event origin** which is the Web page that the event took placed.
- **User ID** that identifies the users.
- **Activity type** that describes if a button or menu is accessed.
- **Session information** that is applicable to the event.
- **Meta data** any additional parameters that user is using to trigger a request from the server.

3.2 System utilisation analysis

In Figure 2 the number user activity log events of six different menu's are logged in any given time frame. This includes all events that will initiated a request to the server. These logs shows how much the menu's are used to based on the amount of logs.

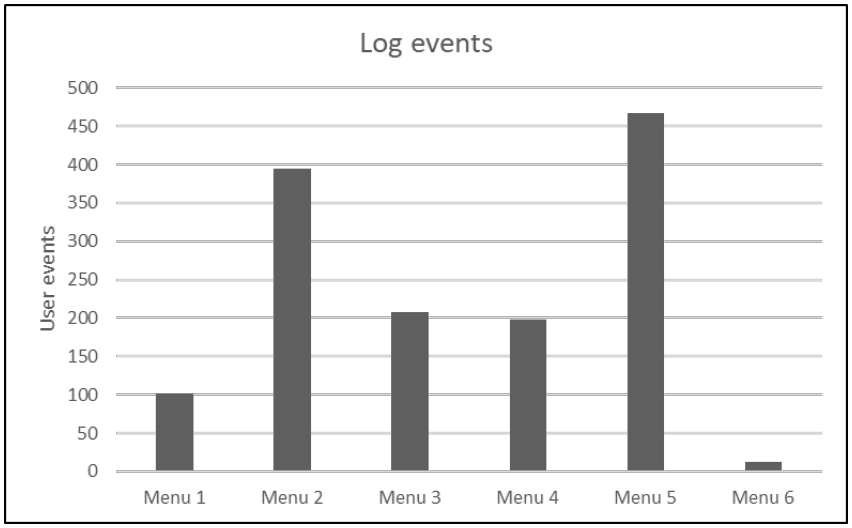

Figure 2: User activities

In the Figure 2 the amount of logs for both menu 1 and 6 are low. This shows how much user activities there are on these menu's. For menu 1 it can be further investigated by a developer if the menu is used as intended or whether it can be deprecated.

The remaining four menus have high number of logs generated in the time frame. These menu's can be prioritise to maximise the development maintenance on the software system. This will increase user experience and reduce or eliminate any possible bugs that may exists in the menus.

The menu's that is moved higher up the priority list of the development team can have significant value to the system health and provide valuable feedback what new features needs to be added. The logs will show the behavioural usage of how the users tends do tasks. Necessary improvements can be made evaluated if the users do use the new features.

The limited resources that is available can be correctly allocated to maintain the software system as the proposed in Figure 3. The improvement of software resource allocation may be beneficial to the development team to better plan software development of the system in the future.

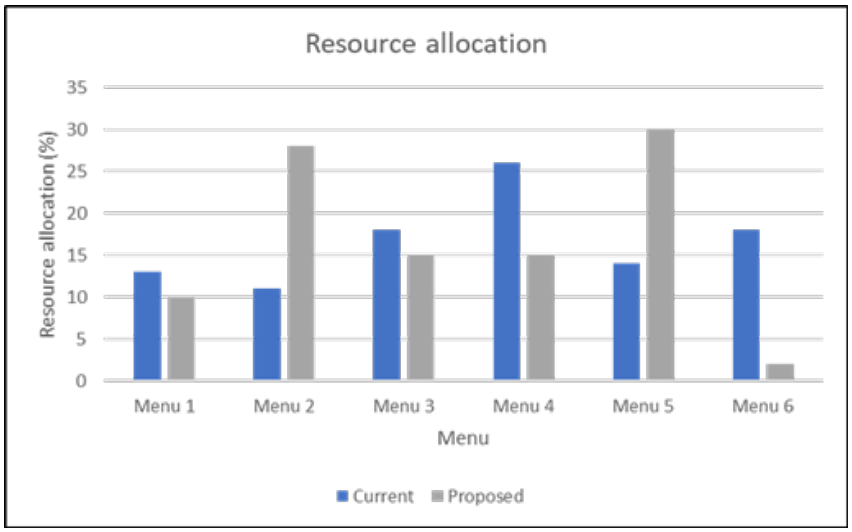

Figure 3: Resource allocation

4 Work Plan

Planned submission for the study as in the proposed timeline in Figure 4 will be for October 2020 and will include the following chapters:

- Introduction
- Methodology
- Results
- Conclusion

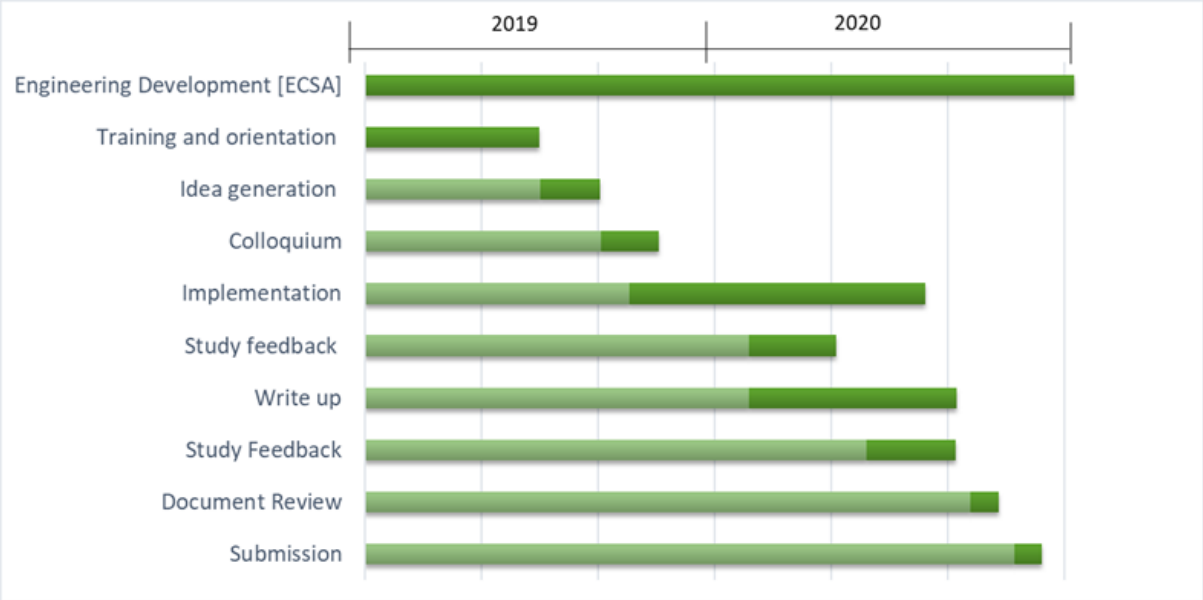

Figure 4: Proposed timeline

5 Conclusion

Development of maintenance software is continuous. It can be complex due the software system. Logging is part of software maintenance that provide information the system.

Most studies does not provide a complete method to create a logging mechanism and do further analysis on the logs. Tracing user activities can provide a method to do system utilisation analysis. The expected result of using user activity tracking will improve software maintenance.

6 Ethics

Please indicate (with an **X**) the following:

		Yes	No
1.	Does the research involve **human participants (adults andor minors)** at any stage of the research? Adults: > 18yrs, Minors: < 18yrs		**X**
1.1	Does the research involve **adult** participants at any stage of the research?		**X**
1.2	Does the research involve **vulnerable**[1] **human participants** and the context of the research is **health related**?		**X**
1.3	Does the research involve **vulnerable**[1]**human participants** and the context of the research is **non-health related**?		**X**
2.	Is the context of the research **health or health-related**?		**X**
3.	Does the research fall in the humanistic disciplines (social, political, institutional, cultural and historical environments)		**X**
4.	Does the research involve **animals** for the purpose of **human health**, **animal health** or **environmental health**?		**X**
5.	Does the research involve **animals** for the purpose of improving animal **breeding** or improving animal nutrition or genetics?		**X**
6.	Are there any risks associated with the research activities that can potentially harm the (local) **environment**?		**X**
7.	Can the research activities potentially bring **physical harm** to yourself and/or other people (such as civilians, laboratory personnel, assistants and/or students)		**X**

[1] *Vulnerability* refers to the diminished ability to fully safeguard one's own interests in the context of a specific research project; may be caused by limited capacity or limited access to social goods like rights, opportunities and power (Department of Education. Second edition. Ethics in Education, Management, Humanitarian and Social Science Research. Principles, Processes and Structures, 2015). E.g. minors or adults with mental incapacity.

Please indicate (with an **X**) the predicted ethics risk level to the best of your knowledge:

No risk: There is no possible risk that the research may lead to any undesirable effects or unexpected negative consequences as no participants are directly involved.	**X**
Minimal, low or negligible risk: The probability, magnitude or seriousness of unexpected negative consequences, harm or discomfort anticipated in the research is negligible and no greater than that of ordinary encounter in daily life ("Daily life" as a benchmark should be that of daily experiences by the average person living in a stable society). Research in which the only foreseeable risk is one of minimal unexpected negative consequences, discomfort or inconvenience.	
Medium Risk: Research in which there is a potential risk of unexpected negative consequences, harm or discomfort, but where appropriate steps can be taken to mitigate or reduce overall risk. Remedial interventions can be undertaken should harm occur.	
High Risk: Research in which there is a real and foreseeable risk of unexpected negative consequences, harm and discomfort, and which may lead to serious adverse consequences if not managed in a responsible manner.	

References

[1] Harry M. Sneed. "A cost model for software maintenance & evolution". In: *IEEE International Conference on Software Maintenance, ICSM* (2004), pp. 264–273.

[2] Nan Niu et al. "Requirements engineering and continuous deployment". In: *IEEE Software* 35.2 (2018), pp. 86–90.

[3] Matthias Galster, Christoph Treude, and Kelly Blincoe. "Supporting Software Architecture Maintenance by Providing Task-specific Recommendations". In: *2019 IEEE International Conference on Software Maintenance and Evolution (ICSME)* (2019), pp. 370–372.

[4] Antonio Pecchia et al. "Industry Practices and Event Logging: Assessment of a Critical Software Development Process". In: *Proceedings - International Conference on Software Engineering* 2 (2015), pp. 169–178.

[5] Valentina Lenarduzzi, Alberto Sillitti, and Davide Taibi. "Analyzing Forty years of software maintenance models". In: *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering Companion, ICSE-C 2017* (2017), pp. 146–148.

[6] Karun Thankachan. "Data driven decision making for application support". In: *Proceedings of the International Conference on Inventive Computing and Informatics, ICICI 2017* Icici (2018), pp. 716–720.

[7] Fabiano Dalpiaz and Sjaak Brinkkemper. "Agile requirements engineering with user stories". In: *Proceedings - 2018 IEEE 26th International Requirements Engineering Conference, RE 2018* (2018), pp. 506–507.

[8] Muhammad Shahid and Suhaimi Ibrahim. "Change impact analysis with a software traceability approach to support software maintenance". In: *Proceedings of 2016 13th International Bhurban Conference on Applied Sciences and Technology, IBCAST 2016* (2016), pp. 391–396.

[9] Kateřina Slaninová. "User behavioural patterns and reduced user profiles extracted from log files". In: *International Conference on Intelligent Systems Design and Applications, ISDA* (2014), pp. 289–294.

[10] Boyuan Chen. "Improving the Software Logging Practices in DevOps". In: 2019, pp. 194–197.

[11] Muhammad Waqar and Davood Rafiei. "Tracking User Activities and Marketplace Dynamics in Classified Ads". In: *Proceedings - 2016 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2016* (2017), pp. 522–525.

[12] Xuezhi Lei. "Modeling and intelligent analysis of web user behavior of web user behavior". In: *Proceedings - 2018 International Conference on Engineering Simulation and Intelligent Control, ESAIC 2018* (2018), pp. 192–195.

[13] Byung H. Park et al. "Big data meets HPC log analytics: Scalable approach to understanding systems at extreme scale". In: *Proceedings - IEEE International Conference on Cluster Computing, ICCC* 2017-Septe (2017), pp. 758–765.

[14] Guoping Rong et al. "How is logging practice implemented in open source software projects? A preliminary exploration". In: *Proceedings - 25th Australasian Software Engineering Conference, ASWEC 2018* (2018), pp. 171–180.

[15] Ali Razavi and Kostas Kontogiannis. “Pattern and policy driven log analysis for software monitoring”. In: *Proceedings - International Computer Software and Applications Conference* (2008), pp. 108–111.

[16] Risto Vaarandi and Mauno Pihelgas. “LogCluster - A data clustering and pattern mining algorithm for event logs”. In: *Proceedings of the 11th International Conference on Network and Service Management, CNSM 2015* (2015), pp. 1–7.

[17] Jinliang Song, Tiejian Luo, and Su Chen. “Behavior pattern mining: Apply process mining technology to common event logs of information systems”. In: *Proceedings of 2008 IEEE International Conference on Networking, Sensing and Control, ICNSC* (2008), pp. 1800–1805.

[18] Madhuri A. Potey, Dhanashri A. Patel, and P. K. Sinha. “A survey of query log processing techniques and evaluation of web query intent identification”. In: *Proceedings of the 2013 3rd IEEE International Advance Computing Conference, IACC 2013* (2013), pp. 1330–1335.

[19] Guoping Rong et al. “A Systematic Review of Logging Practice in Software Engineering”. In: *Proceedings - Asia-Pacific Software Engineering Conference, APSEC* 2017-Decem (2018), pp. 534–539.

[20] Xuetao Tian, Honghui Li, and Feng Liu. “Web Service Reliability Test Method Based on Log Analysis”. In: *Proceedings - 2017 IEEE International Conference on Software Quality, Reliability and Security Companion, QRS-C 2017* (2017), pp. 195–199.

[21] Heng Li, Weiyi Shang, and Ahmed E. Hassan. “Which log level should developers choose for a new logging statement? (journal-first abstract)”. In: *25th IEEE International Conference on Software Analysis, Evolution and Reengineering, SANER 2018 - Proceedings* 2018-March.4 (2018), p. 468.

[22] P. Dhanalakshmi, K. Ramani, and B. Eswara Reddy. “The Research of Preprocessing and Pattern Discovery Techniques on Web Log Files”. In: *Proceedings - 6th International Advanced Computing Conference, IACC 2016* (2016), pp. 139–145.

[23] Jie Lu et al. “ Data-Driven Decision-Making (D 3 M): Framework, Methodology, and Directions ”. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 3.4 (2019), pp. 286–296.

[24] Dan Port and Bill Taber. “Actionable Analytics for Strategic Maintenance of Critical Software: An Industry Experience Report”. In: *IEEE Software* 35.1 (2017), pp. 58–63.

[25] Marcello Cinque, Domenico Cotroneo, and Antonio Pecchia. “Event logs for the analysis of software failures: A rule-based approach”. In: *IEEE Transactions on Software Engineering* 39.6 (2013), pp. 806–821.

[26] Apashabi Chandkhan Pathan and Madhuri A. Potey. “Detection of malicious transaction in database using log mining approach”. In: *Proceedings - International Conference on Electronic Systems, Signal Processing, and Computing Technologies, ICESC 2014* (2014), pp. 262–265.