SSRG International Journal of Recent Engineering Science
ISSN: 2349–7157 / https://doi.org/10.14445/23497157/IJRES-V11I2P101

Volume 11 Issue 2, 1-8, Mar-Apr 2024

Review Article

A Proposed Mathematical Model for Networked and Distributed System Reliability Evaluation in Tanzanian Universities

Paul K.M. Masenya[1], Seleman Ismail[2], Khamisi Kalegele[3]

[1]*Faculty of Information and Communication Technology Ruaha Catholic University, Tanzania*.
[2,3]*Faculty of Science, Technology and Environmental Studies, Open University of Tanzania, DSM, Tanzania.*

[1]*Corresponding Author : masenyapkm@gmail.com*

Received: 11 January 2024 Revised: 19 February 2024 Accepted: 05 March 2024 Published: 20 March 2024

Abstract *- Today, most higher learning Institutions in Tanzania use student record Management Systems (SRMS) in their daily activities. SRMS, being among the Networked and Distributed Systems, contains multiple components, i.e. hardware, software middleware, data, users, and documents. The effective reliability and availability of the system depend on the specifications of individual components, network configurations, and redundancy models. Failure or incorrect functioning of one of the components can result in a catastrophic impact. This paper proposes a mathematical model to measure and predict the reliability of Networked and Distributed systems in higher learning Institutions in Tanzania*.

Keywords *- SMRS Reliability model, System reliability model, Mathematical modeling of distributed system reliability*.

1. Introduction

A distributed system, sometimes referred to as distributed computing, is a system made up of several parts that are dispersed over various machines but communicate and plan tasks together so that the user perceives the system as a single cohesive unit [1]. A networked and distributed system is an essential component of modern life. Its applications range from simple devices like cell phones and remote vehicle keys to complex machinery found in power plants, nuclear reactors, airplanes, and healthcare systems. Hardware breaks and software crashes as a result. Networks fill up with traffic. Worms and viruses cause systems to fail. Data becomes tainted. Information technology is used by users in ways that designers never intended, for better or worse. Coordination relationships are reorganized, and communication flows change as a result of processes. Information systems adapt in response to new issues and environmental shifts even when they are operating efficiently [2]. Given the vital necessity of this system, any element or component failure can have a substantial impact on the component's performance, which could be disastrous or catastrophic.

It is hard to find a mathematical model which models the reliabilities of all networked and distributed system components such as hardware, software, hardware-software interaction, users, data, and documentation together for the case of student record management systems in Tanzanian higher learning institutions that why this article proposes the mathematical modeling of Networked and Distributed systems for all components in the case of students' record management systems (SRMS) in Tanzanian higher learning institutions.

Section two outlines the literature reviews discussing the possible sources and incidents of the need for modeling; section three is the model formulation, and the last part is the conclusion which points out what to do next.

2. Literature Review

Approximately 172 distinct cloud computing outage occurrences occurred between 2008 and 2012, according to a study on cloud computing vulnerabilities [3]. These accidents are mostly caused by (i) unsecured application programming interfaces (APIs) and interfaces, (ii) data leaks and loss, and (iii) hardware malfunctions. These vulnerabilities mostly affected Google, Amazon, Microsoft, and Apple, and they caused significant financial damage [3]. On April 21, 2011, it was claimed that the Amazon Web Service (AWS) experienced a 12-hour outage, resulting in the shutdown of hundreds of well-known websites. The company lost 66,240 US dollars per minute due to this outage.

A programming error resulted in 100,000 extra votes being cast in one Texas county during the 2000 US presidential election, and another incident involved machines in North Carolina losing over 4,000 votes due to memory

overload [4]. Additionally, some candidates received votes cast for other candidates as a result of a programming error. Customers of AT&T lost phone service for voice and data for hours due to a software issue in a four-million-line program [6]. BlackBerry users did not receive email for nine hours after the business installed a defective software update [5]. Others include the May 1998 Galaxy IV satellite computer, which caused Pager service to be discontinued for an estimated 85% of US users, including law enforcement and hospitals, major chain petrol stations were unable to validate credit card payments, and airlines that relied on satellite data for weather updates had to postpone flights [6][7].

According to a study conducted by the International Telecommunication Union (ITU) [9], 50% of Internet users admit to having been the victim of security breaches. An organization's cost of a data breach is estimated to be USD 3.92 million, with an average of 25,575 records compromised annually. A data breach erodes trust and makes investors and customers reluctant to do business with the affected organization (Gordon, Loeb, & Zhou, 2011), quoted in [8].

Tanzanian higher institutions are not in isolation; thus incidents of this nature are happening, so it is clear that the reliability of an information system is fundamental such that stakeholders of higher learning institutions in Tanzania have to understand the elements affecting information system reliability. For this reason, a model to predict and measure the reliability of networked and distributed systems is essential.

2.1. SRMS in Higher Learning Institutions in Tanzania

Student Record Management Systems (SRMS) in higher learning Institutions in Tanzania are faced with several challenges, such as poor recoverability, which does not allow users to correct mistakes they made while using it, and poor internet and network connectivity. Poor ICT expertise, including an insufficient understanding of how to use basic and sophisticated functions to do various tasks and a delay in receiving feedback. Outdated contents (related documents are not frequently updated), frequent power cuts, poor user help, language barrier, and high cost of internet bundles, Lack of information system manager(system administrators) to support other users[10-13]

The study [3] also revealed that the failure to implement smooth SRMS is due to inadequate preparations for change, a lack of support from the Information Technology department, a lack of training for users of the system, and a lack of management support, hence resulting unreliable SRMS. The study by [4] reveals that some courses of unreliable systems are due to the problems of system developers, such as Lack of coordination of software development efforts. Poor and inadequate understanding of user requirements and Rigidity in design efforts and also poor organizational control over the source code and systematic support of users.

2.2. Reliabilities of the System

A system is considered reliable if it can operate as intended for a specified period in a specified environment, work precisely as intended, i.e., following requirements, resist various failures, and recover in case of any failure that occurs during the system execution without producing an incorrect result, have a probability that a functional unit will perform its required function for a specified interval under stated conditions, and can continue to function correctly even after scaling is completed regarding some aspects. [14]–[16].

Thus, the reliability function $R(t)$ is defined as the probability of failure-free operation until time t. Thus, if the random variable X denotes the lifetime of an item, then

$$R(t) = P(X > t). \tag{2.1}$$

and

$$F(t) = 1 - R(t). \tag{2.2}$$

$F(t)$ Is the unreliability and is the complement of the reliability, and Its derivative is called the failure density function[5].

[6] Reported that the contents of a Distributed System are hardware, software, and software-hardware interaction components that act as a bridge between hardware and software, users, data, and documents. Data are the raw material that is manipulated by software and hardware to information and vice versa; the user is essential in recovery, particularly in examining how effective the system is, while documentation of its availability, clarity, and usage directions helps the users on ease on using the system hence facilitating the reliability of the system. It is known that a reliable system can work better on decision-making for companies, sales trends, and increase the performance of a particular work. Hence the need of proposing to formulate a mathematical model to measure and predict the reliability of Networked and distributed systems in the case of Higher Learning Institutions in Tanzania.

2.2.1. Hardware Components

Hardware includes the physical parts of a computer, such as a Central Processing Unit (CPU), monitor, mouse, keyboard, computer data storage, graphics card, sound card, speakers, and motherboard[7], and network hardware, such as networks share devices, servers, clients, transmission media, shared printers and other hardware and software resources, Network Interface Card(NIC)[8].

Hardware failures are further divided into two categories: total and partial. Hardware failures are brought about by hidden defects in the hardware that cause hardware components to cease performing as intended. In particular, catastrophic failures that result in the whole cessation of the intended function are referred to as total hardware failures or

hard failures by certain studies [18]. Soft failures, also known as partial hardware failures, are defined as a situation in which a hardware component continues to work as intended. However, the system is in a state of degeneration. Overall, minor hardware failures may cause the system to continue operating in a degraded condition, but complete hardware failures will not [19].

Faults can be due to a malfunction, physical damage (e.g., bumping, jostling, or dropping), theft, or fire (hard disks). Power Failures, Power Surges, Overheating, Overloading, People(i.e. Human error (e.g. spilling coffee on a device and damaging its internal components or downloading an attachment infected with malware or may be due to a lack of training rather than negligence[9]). Failure rates for any particular component can be obtained from the manufacturer's manual or documentation [10]

The pie chart below shows the percentage magnitude of each hardware component failure in percentage, as quoted from[15, 18, 19].

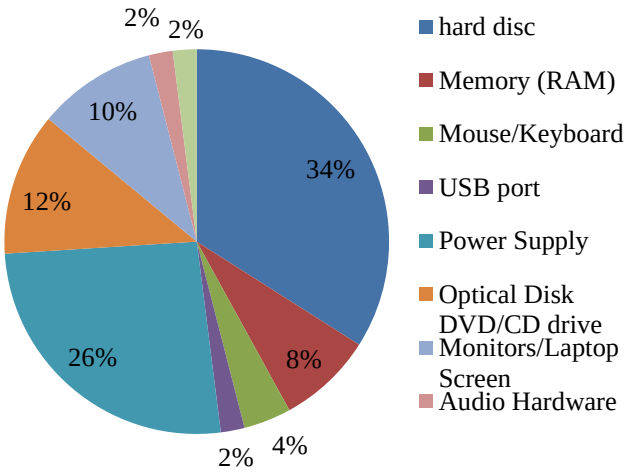

Fig. 1 Percentage magnitude hardware failure (source: Chukwuchekwa et al., 2017)

2.2.2. Software Components

Software components are the set of instructions that are stored and run on the hardware, and these include an operating system (Linux, Windows, etc.), programming languages such as Java, C, C++, etc., and device/utility programs which software required to run a respective device, network operating system(NOS), and application software such SRMS software's [20, 21]

For this case, the software is categorized as Non-Developmental (ND) Software and Developed Software such as Students' Records Management System (SRMS) software. The ND software first includes an operating system, programming languages, and other supporting software such as Windows OS and Linux OS[7]. Secondly is ND reused software; this is software that supports the development of SMRS and has been used somewhere. The Developed Software (SMRS) includes the academic systems, catering systems, library systems, and accounting systems with others included in the definitions of SRMS[15]. Since SRMS are locally developed, it hardly to find out that they have an automated failure recording system where you can obtain the failure rate, such as an in-house failure analysis and corrective action system (FRACAS)[16], Office Customer Experience Improvement Program (CEIP) such as Microsoft Office Systems 2003 which deal with measurement platform for products running in MS Office context and the Microsoft Reliability Analysis Service (MRAS) which focus on reliability (and availability) tracking of Windows servers, and products running on servers like MS SQL database, MS IIS web server, MS Mail Exchange, and the Windows Active Directory[17] [18]. Also, it hardly to find reliability measurement done; that is one of the objectives of this study, i.e. to survey and measure the failure rate of SRMS either by available methods (physical means) or using reliability metric techniques such as product metrics[19], function point metric which is based on the count of inputs, outputs, master files, inquires, and interfaces. [20].

Software failures[21] refer to the occurrence of an incorrect output that is triggered by a specific input because of the latent faults left in software programs, e.g., design errors, that are unrelated to hardware components. The failure, in this case, is all unplanned events, such as software crash, hang, incorrect or no output, which are caused by software bugs and possibly untimely response (too fast or slow).

2.2.3. Hardware-Software Interaction Component

These components fall into two categories: software-induced hardware failures and hardware-induced software failures. They are intermediate components between hardware and software (middleware) [33]. Hardware malfunctions brought on by the operation of an embedded software system are known as software-induced hardware failures. Hardware components can sustain physical damage due to the electrical stress caused by software execution. Failures in software caused by changes in hardware configuration that result in an operational environment that differs from the testing environment are known as hardware-induced software failures [34].

Failures of these types are either planned Events (Updates requiring a restart, Configuration changes requiring a restart) or Configuration failures (Application System incompatibility errors and Installation/setup failures)[11]. Network protocol problems or network anomalies or failures due to network device configuration are part of the component also[22].

2.3. Reliability Models

[23] pointed out that mathematical modelling is widely recognized to be a powerful tool for understanding the

behaviour of many systems ranging from natural to human-made systems since it provides insight into the complex interaction of the system, can estimate expected value (mean/average outcome), most likely (mode outcome) and dispersion/shape (statistical distribution of all outcomes). It is a tool for controlling statistical experiments and allows statistical evaluation (comparison) of alternate strategies. That is why this study aims to develop a mathematical model to study the reliability of networked and distributed systems in higher learning institutions in Tanzanian environments. For reliability models, most of the computing architectures are modeled using a technique such as Reliability Block Diagram (RBD) analysis[24], Fault Tree (FT) analysis[25], Markov Chains (MC)[26], and Bayesian analysis[27].

2.3.1. Hardware Reliability Models

Hardware reliability is typically based on the age of the hardware and the stress of the operational environment[28]. Hardware reliability decreases with increasing age[29]. Hardware reliability consists of components of computer systems such as CPU, Storage Devices, Memory, etc. and servers (such as File Server, DB servers, Web servers, and email servers, etc.), communication infrastructure (Network equipment (switches, routers, etc.)) and connecting devices.

The basic hardware reliability model consists of all hardware elements of the system in series or parallel.

Several distribution functions can be used to model hardware components, such as Exponential Distribution, Normal Distribution, Weibull Distribution, etc. [30][31].

2.3.2. Software Reliability Models

Software reliability is due to increased requirements, design, coding, or interoperability with other related software. It manifests when the software is operated in an environment in which it was not designed or tested[32]. Many factors may cause software failure, and one in particular is that software failures are due to faults in the design. IEEE-Std-729-1991 defines "Software reliability as the probability of failure-free operation for a specified period in a specified environment", and ISO9126 says, "Reliability is the capability of the software product to maintain a specified level of performance when used under specified conditions" [33][32]. It differs from hardware reliability in that it reflects design perfection rather than manufacturing perfection. The high complexity of software is the major contributing factor to Software Reliability problems[34].

Many models exist to measure or predict software reliability, such as Jelinski-Moranda[35][36], Goel-Okumoto, Musa-Okumoto, and Musa's basic execution time models[37]. NHPP model based on the Lindley distribution proposed by [38]. There is no single model that is universal to all situations. In this paper modified Jelinski –Moranda[36] is proposed to model software reliability for network and distributed systems for higher learning institutions in Tanzania.

2.3.3. Hardware-Software (Firmware) Interaction Reliability Models

[32] Created a model that encompassed software, hardware, and hardware-software interaction. The Markov process was utilized to detect malfunctions in the hardware-software interaction. [33] Make note of the fact that hardware-software components can be further divided into two categories: hardware-software failures and software-induced hardware failures. Hardware malfunctions brought on by the operation of embedded software systems are known as software-induced hardware failures. For instance, hardware components may sustain physical harm as a result of the electrical stress caused by software execution. Software failures that are caused by alterations in hardware configuration that result in the program operating outside of its intended operating environment are referred to as hardware-induced software failures [32]. The model is also proposed to be employed to model SRMS in higher learning institutions in Tanzania.

3. Model Formulation

The reliability of the system depends on its component reliability; as we noted above, the Distributed and Networked systems consist of Hardware, Software, hardware-software Interaction, Users(people), Data and Documents. Thus, it is expected to have each component its own model, but due to its independence and dependence, the proposed model has only three main sub-models to measure and predict the reliability of the systems; these are system units discussed below, Data discussed in [39], and user discussed in [40]components.

3.1. System Unit Reliability Model

Due to its compactness and dependability then, the proposed system unit model will consist of hardware, software, and hardware-software interaction components as outlined above; its failure constituents are the failure from hardware failures (*hf*), software failures (*sf*), and hardware-software interaction failures (*hsf*) since any failure from respective component will result into system unit failure. Reliability Block Diagram (RBD) technique is used to model the unit.

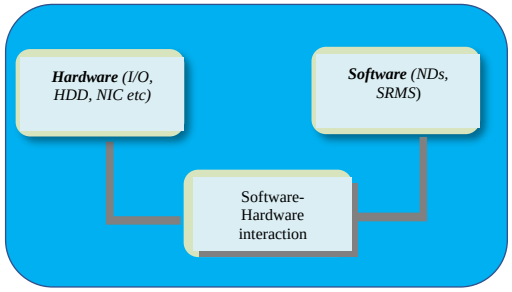

Fig. 2 System Unit Reliability Block Diagram (RBD) (Source: Authors)

Since the components are assumed to be connected in either serial or parallel then, the proposed System Unit reliability model

$$R_{su}(t) = f(R_s(t), R_h(t), R_{hs}(t)), \quad (3.1)$$

Where $R_s(t)$ $R_h(t)$ and $R_{hs}(t)$ are computed from software, hardware, and software-hardware interaction reliabilities models, respectively as shown below.

3.1.1. Software Model

Using the Reliability Block Diagram (RBD), the diagram below shows.

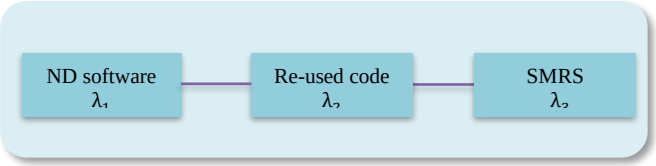

Fig. 3 Software Reliability Block Diagram (RBD) (Source: Authors)

The software Reliability $R_s(t)$ will depend on Non developmental software, reused code, and SRMS thus,

$$R_s(t) = f(R_1(t), R_2(t), R_3(t)) \quad (3.1.1)$$

Where $R_1(t)$is reliable for non-developed software, $R_2(t)$ for reused software and $R_3(t)$for SMRS.

3.1.2. Hardware Model

Using the Reliability Block Diagram (RBD)[41] concept, the model is represented in the following logical connections of components within a piece of equipment. Thus, assuming that all components are connected serially the diagram below shows.

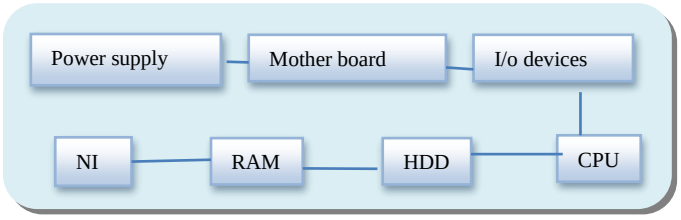

Fig. 4 Hardware Reliability Block Diagram (RBD) (Source: Authors)

Reliability $R_h(t)$ Functions of hardware component's reliabilities (h_i).

$$R_h(t) = f(h_1, h_2, \ldots h_{n-1}, h_n) \quad (3.1.2)$$

3.1.3. Software-Hardware Interaction Model

Using the Markov Process[42] and the concept deployed by [21] and also used by[43], It is assumed that three states for hardware-software interactions modeling are Full working state, Degradation states, and Failure states

Reliability $R_{hs}(t)$ will be a function of Full working state(w), Degradation states(d), and Failure states(f).

$$R_{hs}(t) = f(hs_w, hs_d, hs_f) \quad (3.1.3)$$

3.2. User Model

According to the discussion in [53], User models are accurate depictions of the characteristics (demands, preferences, cognitive, and behavioral features) of each unique user. Typically, the properties are represented as variables. The variables (properties/features) involved in modeling user reliability are mindfulness, background (including one's immediate social context, one's prior knowledge, experience, education level, and work environment), and knowledge in the form of a document that aids the user in completing his task. Therefore,

User reliability model (U_r)

$$U_r(t) = f(m, bk, doc) \quad (3.2)$$

Where m stands for user's mindfulness, bk stands for background factors of a user, and doc stands for document reliabilities

3.3. Data Model

As outlined by [40]to model data, the following assumption regarding data used in networked and distributed systems are used to model Students Record Management Systems (SRMS) in Tanzanian Higher learning institutions.

- Policies and procedures that apply to all or a significant portion of an entity's information systems and aid in ensuring the correct operation of information systems, accessibility of any data authentication, and authorization are required. Assuming that there are two types of controls: general and application.
- Users can exercise control over information systems through several means, such as user privileges, which determine which information a user is allowed to view or use.
- Assuming also that there is a means of ensuring the validity, completeness, accuracy, and confidentiality of data transactions during the processing of applications.

Therefore, the proposed reliability data model will depend on authentications, authorization methods used to access data, Validation techniques to make sure that data accessed are valid, and Certification/verification techniques used for inputting and outputting data are certified to be correct, relevant, accurate, and complete.

Hence Reliability (Da)

$$R_d(t) = f(R_{au}(t), R_{at}(t), R_c(t), R_v(t)) \quad (3.3)$$

Where $R_{au}(t)$, authorization method reliabilities, $R_{at}(t)$, Authentification method reliabilities, $R_c(t)$, Certification/verification technique reliabilities and $R_v(t)$ is validation techniques reliabilities.

3.4. System Reliability Model

As pointed out above reliability of the system will depend on the reliabilities of its components, such as hardware (*h)*, Software *(s)*, hardware-software interaction (*hs*), Users (*u*), data *(d)* and Document *(m)* reliabilities. And due to components dependability's, then

Reliability *R(t)* of a system.

$$\mathrm{R(t)} = f(h, s, hs, u, d, m) = \mathrm{f}(R_{su}, \mathrm{D_a}, \mathrm{U_r}) \quad (3.4)$$

Where

- R_{su} is the System Unit Sub model from Equation (3.1)
- D_a is the Data sub-model from *Equation* (3.2), and
- U_r *is the* User Sub model from *Equation (3.3)*

4. Conclusion

The primary goals of the system reliability model are to confirm that the components are meeting the reliability requirements, to identify component flaws so that corrective action can be taken, to establish failure histories for comparison and use in the prediction of Networked and Distributed Systems (SRMS) reliability, and to provide information about logistics, maintenance, and operations of the system. Thus, it offers helpful assistance in making trustworthy decisions inside the organizations.

This work is the first step towards the development of formulating a mathematical model for SRMS in higher learning Institutions in Tanzania.

The reliability model is composed of hardware, software, software-hardware interaction, data[40], and users[39] sub-components in totality; it is just an abstract model; next is to develop a specification model, i.e., a more detailed and this will include the developing algorithms (sub-models algorithms to be integrated into the main model) necessary for computing and computation model, i.e. an executable model, for simulation purposes.

Acknowledgement

The authors wish to acknowledge the Open University of Tanzania (OUT) and Ruaha Catholic University (RUCU) for their support and for providing a conducive environment, friends, and other individuals for their support during the preparation and production of this paper.

References

[1] Mniko Simon, "An Assessment of The Impact of Online Systems on Enhancing Service Delivery: A Case of the Open University of Tanzania," *A Dissertaion Masters Information Communication Technology, Busisiness Education, Dar es Salaam University*, 2018. [Google Scholar] [Publisher Link]

[2] Justice Agyei Ampofo, "Challenges of Student Management Information System (MIS) in Ghana: A Case Study of University for Development Studies, Wa Campus," *International Journal of Management & Entrepreneurship Research*, vol. 2, no. 5, pp. 332-343, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Karisha D. Kavuta, and Samwel Nyamanga, "The Factors Affecting the Implementation of Students' Records Management System to Higher Learning Institutions in Tanzania a Case of the Institute of Accountancy Arusha," *International Journal of Scientific & Technology Research*, vol. 7, no. 2, pp. 150-156, 2018. [Google Scholar] [Publisher Link]

[4] H.C. Kimaro, and J.L. Nhampossa, "Analyzing the Problem of Unsustainable Health Information Systems in Less-Developed Economies: Case Studies from Tanzania and Mozambique," *Information Technology for Development*, vol. 11, no. 3, pp. 273-298, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[5] Marvin Rausand, and Arnljot Høyland, *System Reliability Theory Models, Statistical Methods, and Applications*, Wiley, pp. 1-636, 2004. [Google Scholar] [Publisher Link]

[6] Joaquin dela Cruz, Computer Systems & The Internet, pp. 1-4, 2014.

[7] Behzad Mahjour Shafiei et al., "Computer Network Routing with a Fuzzy Neural Network," *Advances in Environmental Biology*, vol. 6, no. 3, pp. 1258-1265, 2012. [Google Scholar] [Publisher Link]

[8] R. Gibb, "What is a Distributed System ?," *Computer Engineering*, 2019.

[9] Brandon Hill, Puget Releases Hardware Reliability Report: CPUs, GPUs and SSDs, Tomshardware, 2022. [Online]. Available: https://www.tomshardware.com/news/hardware-reliability-puget-systems-2021

[10] Matt Bach, Most Reliable PC Hardware of 2018, Pugetsystems, 2019. [Online]. Available: https://www.pugetsystems.com/labs/articles/Most-Reliable-PC-Hardware-of-2018-1322/

[11] Christopher Obinna Nnabuife et al., "Conceptual Modeling of PC Hardware Fault Diagnosis System Knowledge Base Using ANN," *IEEE 3rd International Conference on Electro-Technology for National Development*, pp. 491-499, 2017. [Google Scholar]

[12] N. Chukwuchekwa, "Pie Chart Showing Hardware Components Failure Rate Percentage Magnitude Download Scientific Diagram," it's a diagram in 11th ref

[13] Molola B.O. Ajoye, and Williams E. Nwagwu, "Information Systems User Satisfaction: A Survey of the Postgraduate School Portal, University of Ibadan, Nigeria," *Library Philosophy and Practice*, pp. 1-18, 2014. [Google Scholar] [Publisher Link]

[14] Dzifa Peggy Tagbotor, Reindolf Yao Nani Adzido, and Prosper Gameli Agbanu, "Analysis of Records Management and Organizational Performance," *International Journal of Academic Research in Accounting, Finance and Management Sciences*, vol. 5, no. 2, pp. 1-16, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[15] Barbara S. Clements, "*Building an Automated Student Record System: A Step-by-Step Guide for Local and State Education Agencies*," E.S. Publishing, pp. 1-42, 2000. [Google Scholar] [Publisher Link]

[16] DoD, "Military Handbook Electronic Reliability Design Handbook," *DoD*, 1998. [Google Scholar] [Publisher Link]

[17] P. Jalote, and B. Murphy, "Reliability Growth in Software Products," *15th International Symposium on Software Reliability Engineering*, Saint-Malo, France, pp. 47-53, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[18] Stacie Petter, William DeLone, and Ephraim McLean, "Measuring Information Systems Success: Models, Dimensions, Measures, and Interrelationships," *European Journal of Information Systems*, vol. 17, no. 3, pp. 236-263, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[19] M.A. Friedman, and P. Tran, "Reliability Techniques for Combined Hardware/Software Systems," *Annual Reliability and Maintainability Symposium 1992 Proceedings*, Las Vegas, NV, USA, pp. 290-293, 1992. [CrossRef] [Google Scholar] [Publisher Link]

[20] Pankaj Jalote et al., "Measuring Reliability of Software Products," *The International Symposium on Software Reliability Engineering*, pp. 1-8, 2004. [Google Scholar] [Publisher Link]

[21] Xiaolin Teng, Hoang Pham, and Daniel R. Jeske, "Reliability Modeling of Hardware and Software Interactions, and its Applications," *IEEE Transactions on Reliability*, vol. 55, no. 4. pp. 571-577, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[22] Yunzhou Han, Xianglin Zhao, and Jianbin Li, "Computer Network Failure and Solution," *Journal of Computer Hardware Engineering*, vol. 1, no. 1, pp. 16-26, 2018. [Google Scholar] [Publisher Link]

[23] Tuncer Ören, "Future of Modelling and Simulation : Some Development Areas," *Proceedings of the 2002 Summer Computer Simulation Conference*, pp. 1-6, 2002. [Google Scholar] [Publisher Link]

[24] Waqar Ahmad et al., "Reliability Modeling and Analysis of Communication Networks," *Journal of Network and Computer Applications*, vol. 78, pp. 191-215, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[25] Tuan Anh Nguyen et al., "Reliability and Availability Evaluation for Cloud Data Center Networks Using Hierarchical Models," *IEEE Access*, vol. 7, pp. 9273-9313, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[26] F. Kachapova, "Representing Markov Chains with Transition Diagrams," *Journal of Mathematics and Statistics*, vol. 9, no. 3, pp. 149-154, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[27] S. Jannicke Moe et al., "Development of a hybrid Bayesian Network Model for Predicting Acute Fish Toxicity Using Multiple Lines of Evidence," *Environmental Modelling and Software*, vol. 126, pp. 1-17, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[28] K. Muralidharan, and A. Syamsundar, *Statistical Methods for Quality, Reliability Statistical Methods for Quality, Reliability and Maintainability*, PHI New Delhi India, pp. 1-360, 2012. [Google Scholar] [Publisher Link]

[29] Anjushi Verma, "Reliability Assessment of Combined Hardware- Software Time Critical Software Systems, Issues and Challenges," *International Journal of Advances in Electronics and Computer Science*, vol. 6, no. 8, pp. 12-16, 2019. [Publisher Link]

[30] Jinyong Wang et al., "An Optimized Method for Software Reliability Model Based on Nonhomogeneous Poisson Process," *Applied Mathematical Modelling*, vol. 40, no. 13-14. pp. 6324-6339, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[31] Eusgeld, B. Fechner, F. Salfner, M. Walter, P. Limbourg, and L. Zhang, "9 Hardware Reliability," *Research In Education*, vol. 2, pp. 773–777, 2014. doi: 10.12681/hjre.8842.

[32] S.M. Nassar, "Software Reliability," *Computers & Industrial Engineering*, vol. 11, no. 1-4, pp. 613-618, 1986. [CrossRef] [Google Scholar] [Publisher Link]

[33] Sampa Chau Pattnaik, and Mitrabinda Ray, "Software Reliability Prediction and Estimation," *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 8, pp. 855-869, 2020. [CrossRef] [Publisher Link]

[34] Jiantao Pan, "Software Reliability," *Dependable Embedded Systems*, pp. 1-14, 1999. [Google Scholar] [Publisher Link]

[35] Jingwei Liu, Yi Liu, and Meizhi Xu, "Parameter Estimation of Jelinski-Moranda Model Based on Weighted Nonlinear Least Squares and Heteroscedasticity," *Arxiv*, pp. 1-17, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[36] G.S. Mahapatra, and P. Roy, "Modified Jelinski-Moranda Software Reliability Model with Imperfect Debugging Phenomenon," *International Journal of Computer Applications*, vol. 48, no. 18, pp. 38-46, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[37] Ganesh J. Pai, "A Survey of Software Reliability Models," *Arxiv*, pp. 1-12, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[38] Norah N. Al-Mutairi et al., "A New Reliability Model Based on Lindley Distribution with Application to Failure Data," *Mathematical Problems in Engineering*, vol. 2020, pp. 1-11, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[39] P. K. M. Masenya, "Modeling and Simmulation of Reliability of Networked and Distributed Systems : A Case User Reliability Component .," *Int. J. Acad. Eng. Res.*, vol. 6, no. 12, pp. 7–13, 2022.

[40] P. K. M. Masenya and S. Ismail, “Modeling and Simulation of Reliability of Networked and Distributed Systems : A Case Data Reliability Model,” *" Int. J. Recent Eng. Sci. vol. 10, no. 4, pp. 14-18, 2023. Crossref, https//doi.org/10.14445/23497157/IJRES-V10I4P103*, vol. 10, no. 4, pp. 14–18, 2023.

[41] Osman Hasan et al., “Reliability Block Diagrams Based Analysis: A Survey,” *AIP Conference Proceedings*, vol. 1648, pp. 1-4, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[42] U. Narayan Bhat, and Gregory K. Miller, “Elements of Applied Stochastic Processes,” *Operational Research Quality*, vol. 25, no. 1, pp. 192-193, 1974. [CrossRef] [Google Scholar] [Publisher Link]

[43] Mengmeng Zhu, and Hoang Pham, “A Novel System Reliability Modeling of Hardware, Software, and Interactions of Hardware and Software,” *Mathematics*, vol. 7, no. 11, pp. 1-14, 2019. [CrossRef] [Google Scholar] [Publisher Link]