

Augmenting The Compliance of ISO 27001 Operations Security by Automating the Manual Change Request Process in a Fin- Tech Environment

MSc Research Project
MSc in Cybersecurity

Junaid Ijaiya
Student ID: X20101350

School of Computing
National College of Ireland

Supervisor: **Vikas Sahni**

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: JUNAID ADEDAMOLA IJAIYA.....

Student ID: X20101350.....

Programme:MSc Cybersecurity..... **Year:** 2021.....

Module: INTERNSHIP.....

Supervisor: VIKAS SAHNI.....

Submission

Due Date: 06/09/2021.....

Project Title: **Augmenting The Compliance of ISO 27001 Operations Security by Automating the Manual Change Request Process in a Fin-Tech Environment**.....

Word

Count:5103..... **Page Count**.....18.....

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: JUNAID IJAIYA.....

Date: 05/09/21.....

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Augmenting The Compliance of ISO 27001 Operations Security by Automating the Manual Change Request Process in a Fin-Tech Environment

Junaid Ijaiya
X20101350

Abstract

The upgrade of faulty IT infrastructure can result in key system and application disruptions, resulting in security breach which leads to significant financial losses for organisations. This breach can be mitigated or even prevented against with the introduction of a change request process. All phases of the change request procedure should be covered by the change request specification.

A method for specifying change request and aligning it with ISO 27001 operations security is described in this paper. This technique automates the whole phase of the conventional change request procedure starting with the Implementer making the request for change to the assigned owner for approval and down to communication to offer a precise specification of the change request that is closely tied to the software architecture. This is achieved with the design of a process workflow, creation of a web application and integrating both together. This gives a path to proper documentation of scheduled and implemented activities and a trailed change request chain for referral and accurate logging.

Keywords – Change Request Process; Automation; IT Infrastructure

1 Introduction

Throughout the life cycle of a software system, it evolves. Support for development-related adjustments and a wide range of tools and strategies or processes are available to support changes that arise during the development process. Enhancing a better performance and a better security posture is frequently the driving force behind changes that arise during the maintenance phase. These changes, known as change requests, are often submitted as Microsoft word documents via email for review and approval. These change requests move through the phases of a standard change process that is integrated into the software development process, which is planning, analysis, design, implementation and testing after original definition and submission

In today's era where cyber-crimes and cyber-attacks are rampant, companies and organisations should lay enough emphasis and place focus on the operations security and its documentation. Organizations can use the ISO 27001 information security management standard to analyze and document their information security practices. Information security management guidelines, on the other hand, have been chastised for focusing on the process's presence rather than its content [1]. Information systems are

routinely subjected to a wide range of risks that come arise from various reasons such as use of equipments manufactured by different manufactureres/vendors, different operating systems can be used and most importantly the widely connection of devices to the internet can tremendously jeopardize the three dimensions of information security: confidentiality, integrity, and availability. There's no guarantee that information security management standards will influence employee adherence to security policies which is the main drive of this research, design, and automation of the process [2].

Information security should not only be excluded to technology, but also includes the users of the technology and the process of executing or using the technology. Knowing fully well that Information and cyber security is the responsibility of all employees in an organisation, from junior staff to high management. The ISO 27001 helps guide organisations to have an overall security posture, with its over 100 security controls divided into 14 different domains [3]. The is responsible for ensuring the proper and secure operation of information processing facilities. On several occasions have the operations of various organisations been jeopardised by cyber attacks and they are unable to fully recover from the impacts because of lack of proper documentation. This security risk can be countered with the implementation of an automated operations procedure change management process that has every activity well documented and up to date. This will aid also in the consistent and effective running of systems for new employees or shifting resources, and they are frequently required for disaster recovery, business continuity, and when personnel availability is a concern.

Unauthorized modifications can result in costly and error-prone service interruptions if there is no defined IT change management strategy that provides visibility and control. Process control is vital in a financial technology environment, adequately managed change management is required to guarantee that changes are suitable, effective, and ensure it goes through the right chain of authorisation before any implementation takes place. This should be done in a way that minimizes the risk of purposeful or unintentional compromise of the process, data, or metadata [4]. The automation of the already existing change management process will not only help with the change management process but will also make audit logs available for the purpose of justification or evidence of compliance.

This paper is to describe a strategy for aligning various Information Security Risk Management methodologies with change management process because of various business requirements. These approaches aid in the operational decision-making process while also assisting with central risk reporting. This paper describes a practical method to such an approach. The successful implementation of this change request process automation will be a response to the research question; **Is it possible to develop a change management process that will comply with ISO 27000?**

2 Related Work

The concept of change management has long been a hot topic, not just in the technology industry but also in academic research. As a result, a few articles on change management body of knowledge have been published; nevertheless, the number of papers focusing on change management strategies is restricted. Khalid et al [5] gives shortage of information knowledge, as well as insufficient training and very low awareness programs, are among the reasons as why ISO 27001 has low industry usage. Organizations overlook the dangers that come with lack of information security, such as cyber-attacks and viruses. Information security can be improved by synchronizing processes, policies, and systems to control IT risk [6].

Aaron et al [7] presented an approach for gradually bringing best-practices based automation into the delivery of IT service and change management to add greater structure to the work of automating systems management. Using best-practice methods, they organized the automation, identified high-value automation opportunities, identified interaction points between automation and its broader environment, and revealed the impact of the automation on the service delivery organization. To employ this methodology to automate change management process while conforming with best practice by automating, requires a large amount of work which counters flexibility.

Several research methods have been conducted on the protection and compliance of data centre equipment's and cloud computing instances with information security frameworks. Dedy et al [8] created an ISMS framework for the data centres based on ISO27001 Annex A with 21 information security controls covered by the proposed ISMS framework. They compared the proposed ISMS framework for data centres to the ISMS frameworks for government offices and telecommunications companies. And it was evident that the policies can be bypassed without implementers of the change management being compelled to stick to standard. Lacroix et al [9] stated that the ability to model the CR process has several advantages. For starters, the system can enforce and manage the process. When there is no CR that has been identified as having a high Impact, the system will, for example, block someone from adding additional activities. According to the Capability Maturity Model (CMM), [10] proposed a change process model based on Software Configuration Management tool that clearly aligned standardised process in line with automation.

Leila et al [11] focused on change request scheduling in IT change management. They defined change scheduling as the process of allocating change requests to available change windows while considering other constraints such as the order in which changes are made. Various hinderance were highlighted such as change request dependencies and the change request window. They only concentrated on and offered a mathematical model for IT change management scheduling using a mixed integer programming approach in their research, and they did not pay attention to other aspects that make up the change management process.

Knowing fully well that faulty IT infrastructure upgrades can result in key system and application disruptions, resulting in significant financial losses. Kadar et al [12] proposed a

change planning assistance tool that intends to help change requesters leverage aggregated information about the change, such as previous failure reasons or best implementation techniques. They proposed using various information retrieval and machine learning approaches to classify incoming modification requests automatically.

Zeljko et al [13] agreed that software and hardware systems evolve throughout their lives and changes that originate in development are supported. A vast range of equipment and procedures are available to aid the process. During software maintenance, there may be some changes. Customers are usually the ones who drive the phases. These software or hardware modifications referred to as change requests, are often submitted as Microsoft Word documents or Microsoft Excel documents and sent by email. But they proposed development of a website where the initial specifications of the changes to be made are input to pass through phases of standard software process that has a mechanism built in for procedural development. A case study is presented in their paper which a method for specifying modification requests in the context of a running application on the customer's end. This technique proves IT service management operations are more efficient when they are automated by modifying the first phase of the conventional change request procedure in order to offer a precise specification of the change request that is closely tied to the software architecture [14].

Sanjay et al [15] described the main objective of their research, as stated at the outset, as to make the creation of a fully working service desk, replete with efficient and trained agents, as well as the associated processes, procedures, and paperwork, as simple as possible. They used AI for smart automation, strategic insights and for predictive analysis that can generate reports that may flag violation of change management SLA, detecting possible IT issues and modifying failure patterns. Hechler et al [16] also proposed the implementation of AI to have a significant impact on IT change management and also counter the challenges highlighted in [17], introduced new paradigms characterized by data insight-driven decision making and optimization applicable to all process steps, they also highlighted and made it known that AI does not have infinite powers; there are evident limitations that have sparked both skepticism and investigation.

There are two types of change requests: those made prior to the first release and those submitted after the first release (maintenance phase). The link between software development, software maintenance, and change management is depicted in [18].

In the research conducted by [19], he proposed a strategy to create a standard set of categories to help you and your customers understand the required balance for each service; defined a change model with fewer details than a standard change and reduced the number of approvers required for each change; and proposed a strategy to create a standard set of categories to help you and your customers understand the required balance for each service [20].

The article by Noema et al [21] looked at the process of change management and addressed a variety of software packages and other tools that may be used to assist organizations adopt change management. Change is often badly managed in businesses, and the knowledge in this article should be helpful in managing the inevitable changes that occur in all organizations.

In order to improve communication between the firm and its clients, a method of applying change management processes in IT support and development organizations was offered by

[22]. On the other hand, [23] was inspired by enterprise customers' drive to reduce the time of change managements and ensuring they align with the laid down procedures which led to the development of an automated change management process based on electronic contracts, Furthermore, they suggested at the end of their research that in order to relate SLAs more closely to resource instrumentations, as well as support for Service Level Agreement negotiation, then more research needed to be done [24].

3 Research Methodology

3.1 Spiral Model

The research methodology that was adopted for this project is the Spiral model, the spiral model is a combination of the idea behind the iterative development and the waterfall model with a focus on risk analysis. This is because of the nature of risk involved in this project, where development is incremental, and several other features will be integrated in several iterative phases. Also, it will help in capturing the project requirements accurately. The spiral methodology allows for incremental deployment of a software product. There are four phases to the spiral model. Spirals are iterations in which a software project goes through these phases again.

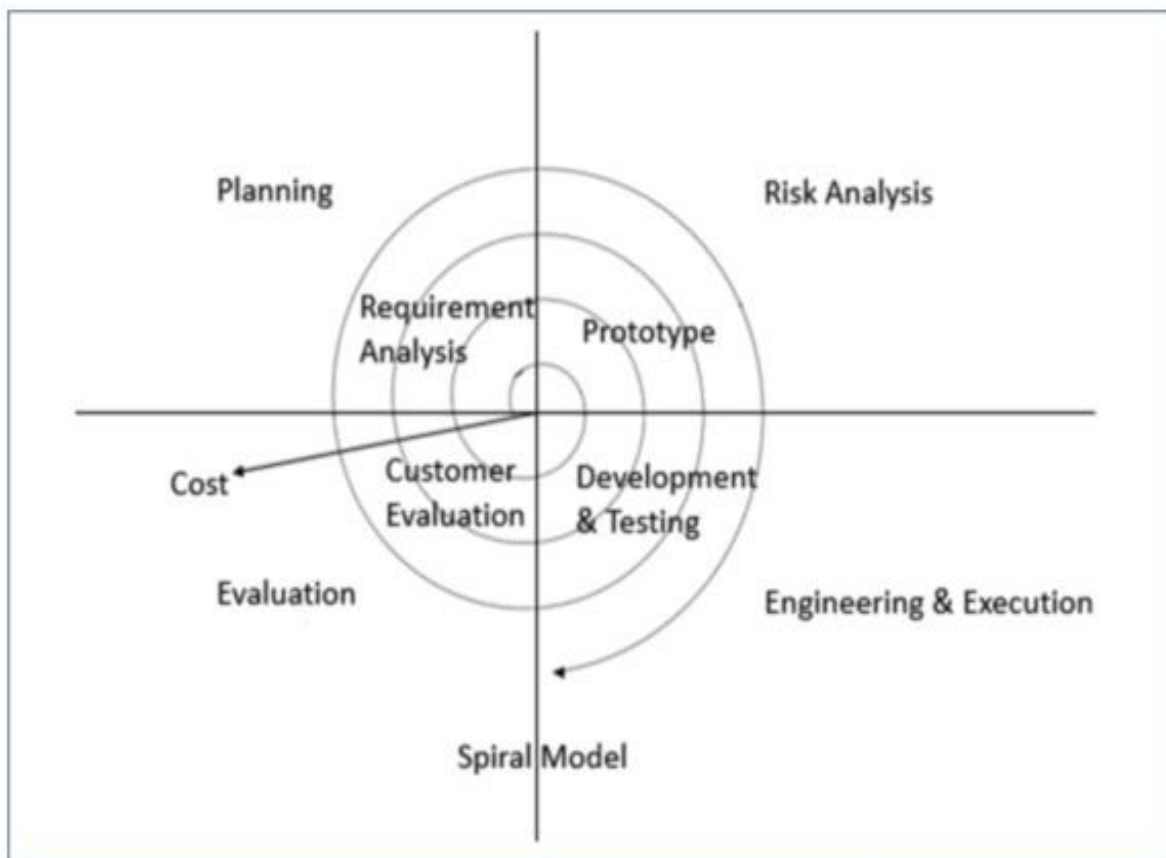


Figure 1: Spiral Methodology

The phases of the spiral methodology are four which are:

3.1.1 Planning

This is the spiral model's starting point, and it's utilized to gather business requirements at this stage. As the product matures, this phase is followed by the identification of system requirements, subsystem requirements, and unit requirements. This phase also includes ongoing communication between the customer and the system analyst to ensure that both parties are on the same page regarding the system's requirements. This is the stage where communication with staffs of the company occurred with the aim of getting the understanding of how the existing change management process occurs and the approval method before any operational change activity is performed on the network. In consecutive spirals, the design process begins with the conceptual design of the baseline spiral and goes through architectural design, logical module design, and final design.

3.1.2 Risk Analysis

Risk analysis is used to identify, quantify, and monitor technical feasibility and management risks such as timetable slippage and cost overrun. After testing the software, the customer evaluates it and provides feedback at the end of the first iteration. Based on the customer's feedback, the software development process moves on to the next iteration, and then follows a linear strategy to implement the customer's suggestions. Throughout the life of the software, the spiralling process of iterations continues.

3.1.3 Engineering

It entails software testing, coding, and deployment. The code is generated and tested several times until it achieves the desired outcome, resulting in a low-risk environment for following development phases. Software is developed as well as end-stage testing during this phase. As a result, testing takes place during this time.

3.1.4 Evaluation

The system uses this phase to check the project output signal to the deadline before going on to the next spiral. At the end of each stage, the client evaluates the results and provides comments to the team.

4 Design Specification

The process workflow and design specification of our suggested model are provided and examined in detail in this part. The process flow is divided into three stages which are:

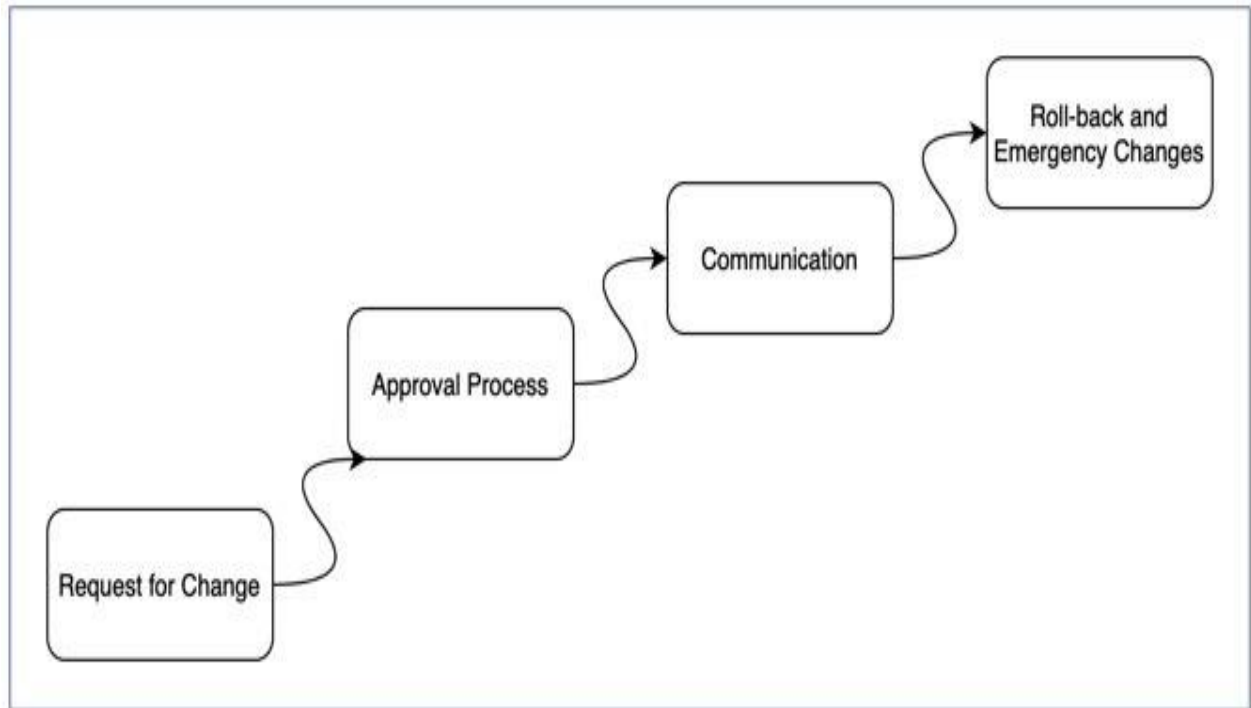


Figure 2: High-Level Process Workflow Design

4.1 Request For Change

A Request for Change can also be called RFC. It can be used to initiate any change. This request will also serve as a record and verification that the requested change was made.

Either an internal change made by an employee or the one by an external client, the modification will be recorded in a certain form. Changes to the organization's assets (hardware, software, networks, and so on), as well as procedures, services, and agreements, can all be affected. As a result, it's critical that the RFC contains precise information regarding the type of change. Additional information, such as the name and contact details of the implementer, the supposed date of implementation, the department in which the change will affect, and so on, should be recorded.

4.2 Approval Process

The Request for Change is received by a person who oversees examining it and, in this case, the examiner is a Level 2 Engineer which is the first filter. This person's sole responsibility is to examine the request's specifics and determine the possible impact the change might have on the business, both in a financial way and the information security implications (for example, if the modification is to upgrade the operating system of a live server). Following that, another

person (often the person in charge of modifications, such as the Team Leader or Change Manager) will decide whether the change is accepted or refused based on the information gathered before. It is critical to assess all the potential consequences of the change, including internal ones (departments, compliance with information security regulations).

Finally, because not all changes are Standard, Normal, Emergency and Major). The impact on the business and the ISMS can be used to construct this classification.

4.3 Communication

It is critical that the organization maintains contact with the change request initiator, as well as interested parties involved in the change (stakeholders, users, customers, the or the teams partaking or affected by the change, because they must be informed of every decision or action taken in relation to the change that is being managed (for example, through the person responsible for changes). These communications can take the form of phone calls or emails.

4.4 Roll-back and Emergency Changes

A key factor to address is what happens if an error occurs during the change's implementation. In this instance, having a fallback process to revert to the previous state is critical. The person in charge of performing the fall-back method is also in charge of the change implementation. Finally, at the implementation stage, this fall-back procedure can be defined, defining what must be done to return to the previous level.

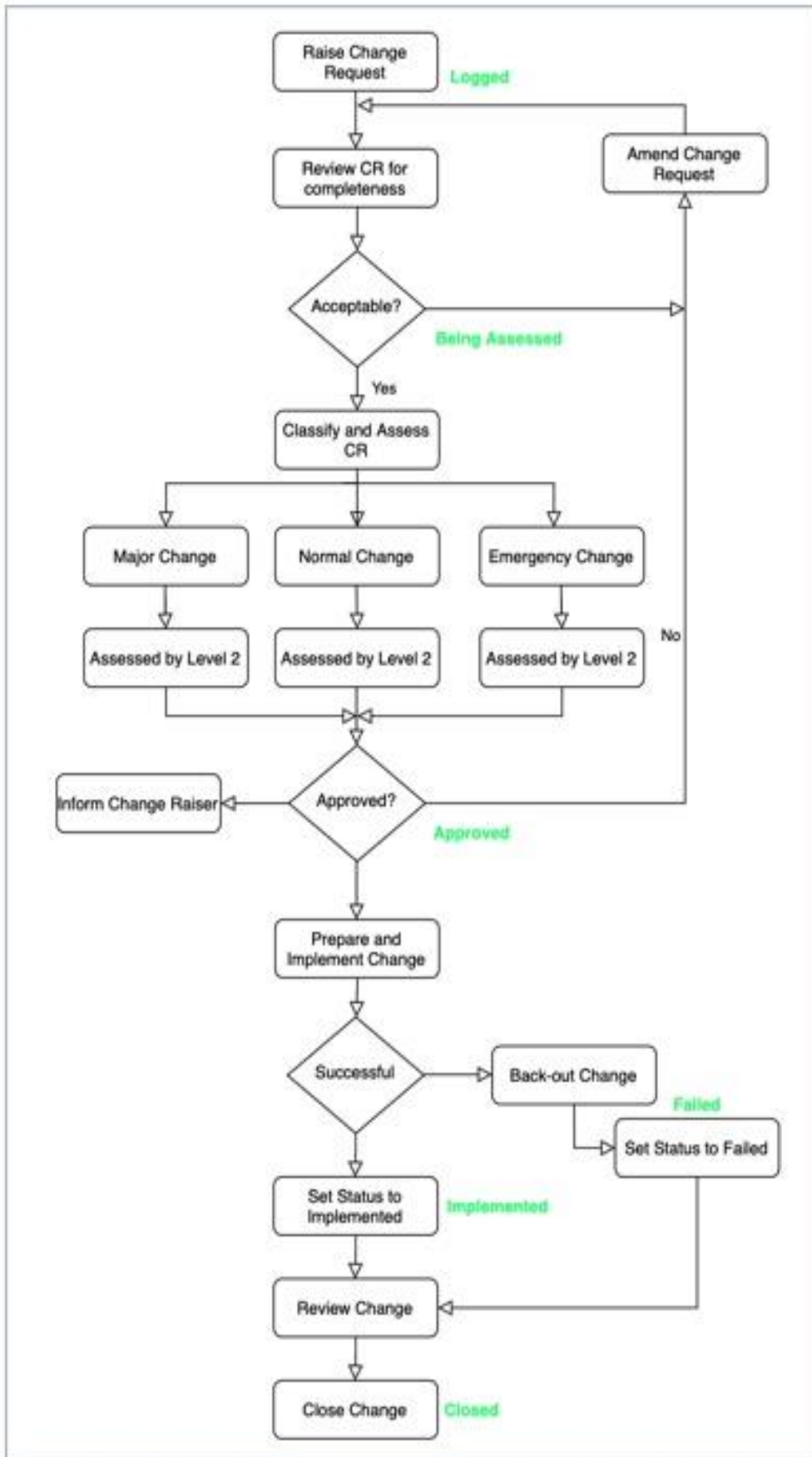


Figure 3: Low-Level Architectural design of the automation

5 Implementation

All the tools and technologies used in the model's implementation technique are discussed and explained in this section. This automated web application's software architecture is divided into two parts: front-end and back-end. They are hosted and deployed separately on distinct Node.js instances but being kept in the same Git repository.

5.1 Front-End

5.1.1 Figma

This is a web-based design software that was used for the design of the user graphical interface, user interface, and prototype of the user experience.

5.1.2 HTML

Hypertext Markup Language is a mark-up language for organizing and presenting elements of a web page on the internet. Tags are used to indicate different parts of the web app such as the title, header, paragraph, and body. HTML elements and attributes are not shown by web browsers; instead, they are used to comprehend the content of the page. HTML5, the most recent version of HTML was used for the implementation of the project.

5.1.3 React

React is an open-source JavaScript library for creating user interfaces also known as front-end designs. There is only one html file in this web application, it is a Single Page Application (SPA), and new elements, sections, or pages are mounted or unmounted from the page as the need arises based on the state of the application.

5.1.4 CSS

Cascading Style Sheets (CSS) is used to describe how the HTML elements are presented on a computer screen. CSS can be used to control the layout of multiple web pages at once. CSS can be used inline, on the inside, or on the outside. Tailwind CSS, a highly flexible CSS utility framework that aids in the creation of a responsive system, was employed. It comes with many style classes that may be applied to HTML elements in stages.

5.1.5 Redux

Redux is a state-management JavaScript package that allows the application's state (dynamic data) to be accessed from anywhere in the codebase, regardless of how deeply buried the component is. This results in a codebase that is cleaner, more scalable, and easier to maintain.

5.2 Back-End

5.2.1 Python (Django)

Django is a high-level Python web framework that was used to promote the rapid development and simple, practical design. It's built by professional developers to take care of a lot of the headaches of web development by focusing on the development of the web application. It's open source and free.

5.2.2 Google Cloud SQL (Postgres)

Cloud SQL for PostgreSQL is a fully managed object-relational database service for Google Cloud Platform that was leveraged to help set up, maintain, manage, and administer a PostgreSQL relational database. This means that Postgres has capabilities such as table inheritance and function overloading, which are useful in this application.

5.2.3 Django REST Framework

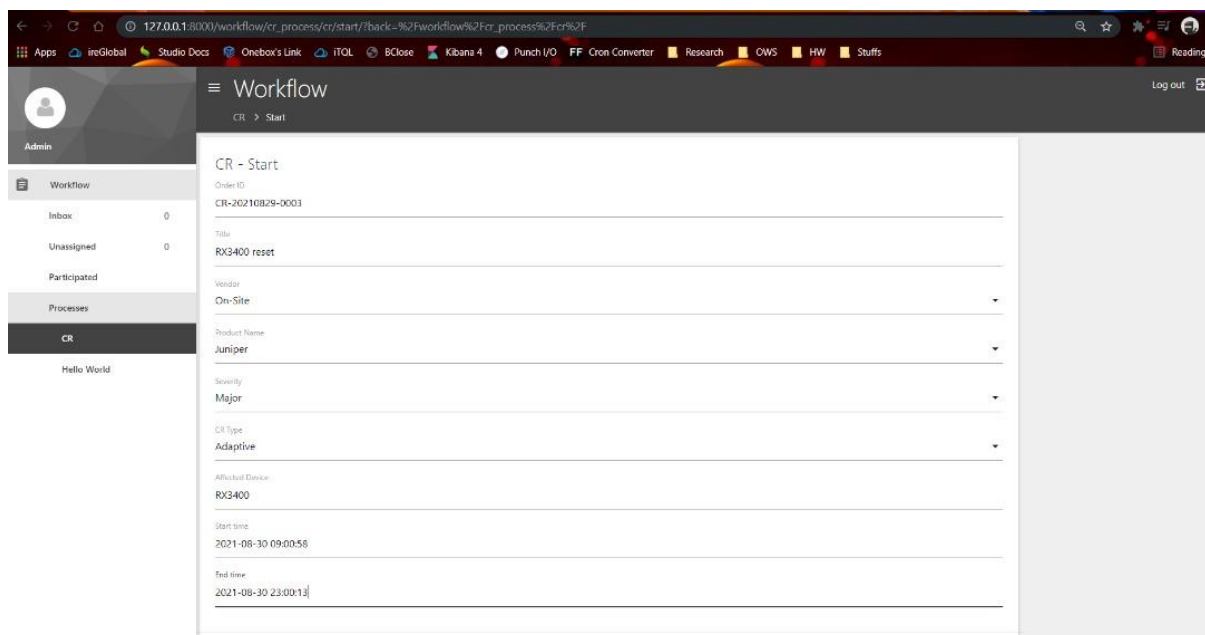
This is the tool that was used in building the web API which requires python and Django to transfer information from an interface to a database.

6 Evaluation

This section assesses the efficacy of the proposed model. Evidence of the strict compliance of the change management process in accordance with ISO 27001 information security management (operations security) standard has been demonstrated with various case studies from the designed system.

6.1 Case Study 1

This first case study shows the dashboard that is popped up with the request form with the details of the activity that is to be filled. The order ID is the auto-generated ticket ID that uniquely identifies individual operations. The Title field is a text field that allows for the title of the operation. Dropdown feature was used for fields like Vendor, Product name, Severity and CR Type so they are compelled to pick only from the available features. The start time and End-time were designed in order not to allow for activity overlapping which can result into multiple simultaneous failures.



The screenshot shows a web browser displaying a Change Request (CR) form. The browser's address bar shows the URL: 127.0.0.1:3000/workflow/cr_process/cr/start/?back=%2Fworkflow%2Fcr_process%2Fcr%2F. The page title is "Workflow" and the breadcrumb is "CR > Start". The form is titled "CR - Start" and contains the following fields:

- Order ID: CR-20210829-0003
- Title: RX3400 reset
- Vendor: On-Site
- Product Name: Juniper
- Severity: Major
- CR Type: Adaptive
- Affected Device: RX3400
- Start time: 2021-08-30 09:00:58
- End time: 2021-08-30 23:00:13

The sidebar menu on the left includes "Workflow", "Inbox: 0", "Unassigned: 0", "Participated", "Processes", and "CR". The user is logged in as "Admin" and the page says "Hello World".

Figure 4: Change Request Form

6.2 Case Study 2

This section of the automated change request management is where the ticket assigned to the Level 1 approver is seen, and the approval field is a dropdown which is either an Approve or Reject, followed with a comment that is sent back to the actual CR implementer to see the reason for the decision.

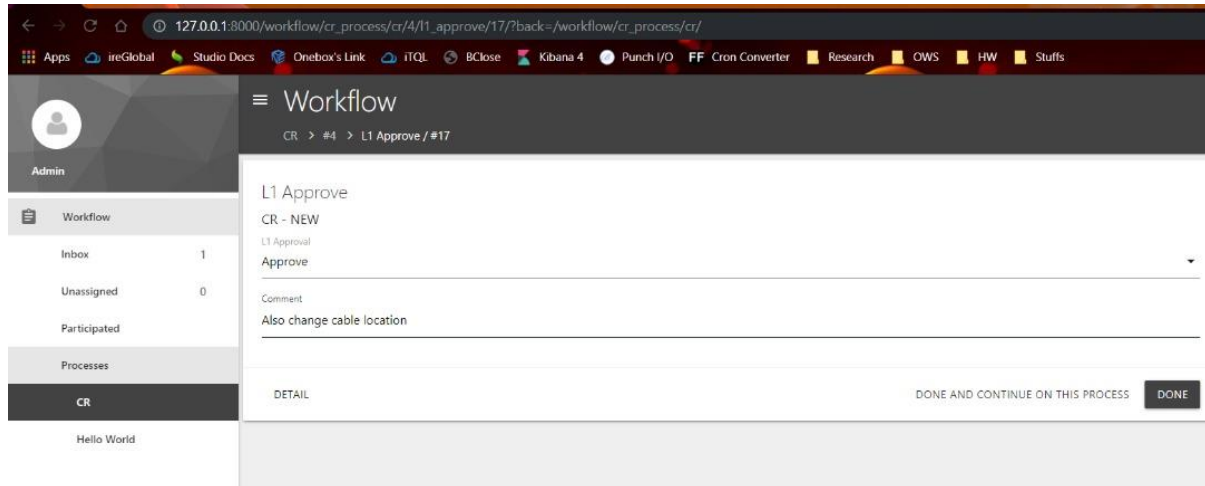


Figure 5: Level 1 Approval Page

6.3 Case Study 3

This section of the automated change request management is where the ticket is designated to the L2 approver which is the customer after it has been approved by the Level 1 approver, and the L2 approval field is a dropdown which is either an Approve or Reject, followed with a comment that is sent back to the L1 approver and actual CR implementer to see the reason for the decision.

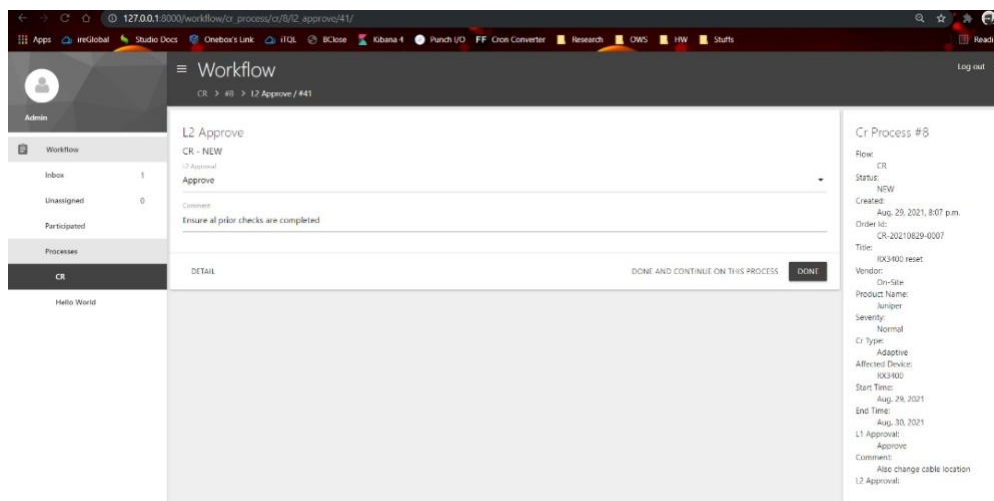
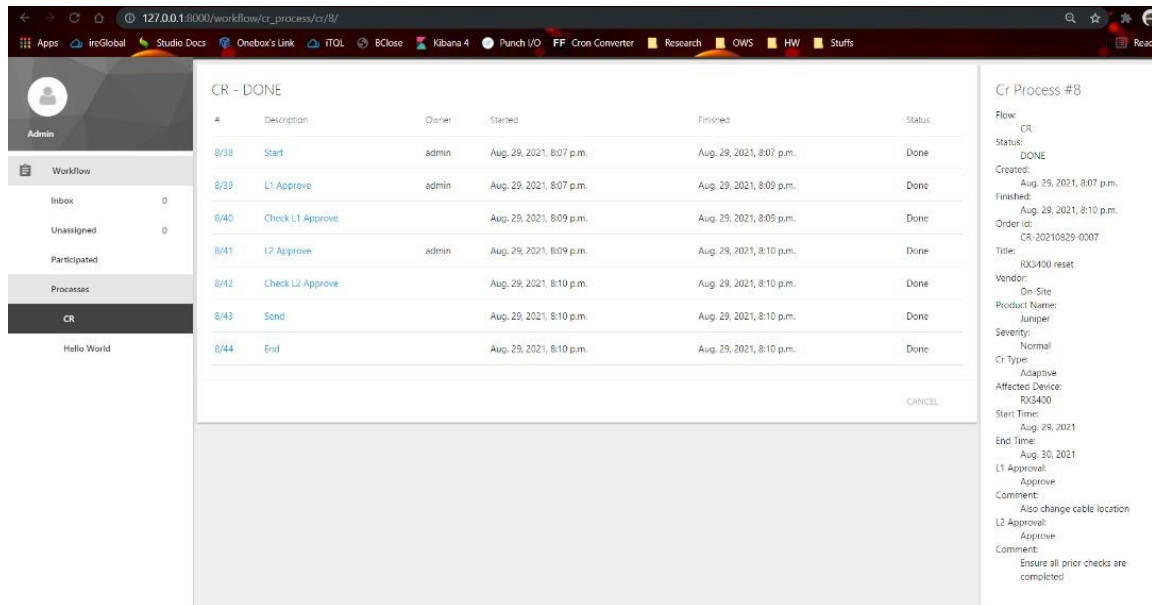


Figure 6: Level 2 Approval Page

6.4 Case Study 4

This section of the change request management system gives an overview of the lifecycle of the ticket showing its current state and the depth of compliance and approval the scheduled operation has attained.



#	Description	Owner	Started	Finished	Status
8/38	Start	admin	Aug. 29, 2021, 8:07 p.m.	Aug. 29, 2021, 8:07 p.m.	Done
8/39	L1 Approve	admin	Aug. 29, 2021, 8:07 p.m.	Aug. 29, 2021, 8:09 p.m.	Done
8/40	Check L1 Approve		Aug. 29, 2021, 8:09 p.m.	Aug. 29, 2021, 8:09 p.m.	Done
8/41	L2 Approve	admin	Aug. 29, 2021, 8:09 p.m.	Aug. 29, 2021, 8:10 p.m.	Done
8/42	Check L2 Approve		Aug. 29, 2021, 8:10 p.m.	Aug. 29, 2021, 8:10 p.m.	Done
8/43	Send		Aug. 29, 2021, 8:10 p.m.	Aug. 29, 2021, 8:10 p.m.	Done
8/44	End		Aug. 29, 2021, 8:10 p.m.	Aug. 29, 2021, 8:10 p.m.	Done

Cr Process #8

Flow: CR
Status: DONE
Created: Aug. 29, 2021, 8:07 p.m.
Finished: Aug. 29, 2021, 8:10 p.m.
Order id: CR-20210829-0007
Title: RX3400 reset
Vendor: On-Site
Product Name: Juniper
Severity: Normal
Cr Type: Adaptive
Affected Device: RX3400
Start Time: Aug. 29, 2021
End Time: Aug. 20, 2021
L1 Approval: Approve
Comment: Also change cable location
L2 Approval: Approve
Comment: Ensure all prior checks are completed

Figure 7: Change Request Ticket Lifecycle Status Verification

6.5 Case Study 5

This case study is a prove that the proposed automated change request process system can achieve another of its aim which is a detailed reporting. Here, previous activities can be found and searched for the purpose of auditing or referring or even when generating insights on general operation maintenance.

#	Summary	Created	Finished	Active Tasks
10	CR - NEW	Aug. 29, 2021, 10:42 p.m.		1
9	CR - DONE	Aug. 29, 2021, 10:36 p.m.	Aug. 29, 2021, 10:39 p.m.	
8	CR - DONE	Aug. 29, 2021, 8:07 p.m.	Aug. 29, 2021, 8:10 p.m.	
7	CR - DONE	Aug. 29, 2021, 8:02 p.m.	Aug. 29, 2021, 8:03 p.m.	
6	CR - DONE	Aug. 29, 2021, 7:59 p.m.	Aug. 29, 2021, 8 p.m.	
5	CR - DONE	Aug. 29, 2021, 7:49 p.m.	Aug. 29, 2021, 7:57 p.m.	
4	CR - DONE	Aug. 29, 2021, 7:23 p.m.	Aug. 29, 2021, 7:36 p.m.	
3	CR - DONE	Aug. 22, 2021, 4:50 a.m.	Aug. 22, 2021, 4:51 a.m.	
2	CR - DONE	Aug. 22, 2021, 4:48 a.m.	Aug. 22, 2021, 4:49 a.m.	

Figure 8: Change Request Ticket Dump

6.6 Discussion

Experiments were conducted in a typical way this automated change management process would be used in the industry. Based on the results conducted as part of this study that was derived from the simulation of actual change management process in order to demonstrate the efficacy of an automated change management approach that complies with ISO 27001 Operations Security. It is evident that this method introduces automated change management, which clearly outlines standardised processes for implementing changes, as well as a standardised manner of supplying relevant information regarding a change management operation and the approval process. It is seen from the above experiment that information security cannot be bypassed using the proposed design.

This design is a good design overall, though improvements can be made in future updates by sending a mail notification when a new change request is created. This will be done by setting a data rule to trigger that will trigger a mail server and send the update to the implementor. Django model rules and google mail servers will be the new addition to be dependent on the already existing technology this built on.

7 Conclusion and Future Work

In this paper, the automation of a workflow management system has been proposed, designed, and implemented, with a case study of change management. The automation includes a section where the request for change will be input, and it will be channelled through the chain of approval without any breach of process. This successful implementation answered the question “**Is it possible to develop a change management**

process that will comply with ISO 27000?” by ensuring changes done on IT infrastructures comply with the ISO 27001 operations security. The findings in this report serve as a starting point and fundamentals. For future works, I would ensure the implementation includes a simple notification service (SNS) in which for every completed request form or approval stage a text message is sent to the implementer and all parties involved in the whole operations.

It is also important to note that the underlying workflow management process can be used in future to implement various automated process such as an Account Management System, Human Resources Management System, Billing and Quoting Automation.

References

- [1] V. Monev, "Organisational Information Security Maturity Assessment Based on ISO 27001 and ISO 27002," in *2020 International Conference on Information Technologies (InfoTech)*, Varna, 2020.
- [2] A. Tanović and I. S. Marjanovic, "Development of a new improved model of ISO 20000 standard based on recommendations from ISO 27001 standard," in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, 2019.
- [3] Angraini, Megawatti and H. Lukman, "Risk Assessment on Information Asset an academic Application Using ISO 27001," in *The 6th International Conference on Cyber and IT Service Management (CITSM 2018)*, Medan, 2018.
- [4] C. Kadar, W. Dorothea, I. Jose, H. Dirk and L. Mario, "Automatic Classification of Change Requests for Improved IT Service Quality," in *2011 Annual SRII Global Conference*, San Jose, 2011.
- [5] K. Alshetri and A. Abdulmohsen, "Exploring the Reasons behind the Low ISO 27001 Adoption in Public Organizations in Saudi Arabia," in *2014 International Conference on Information Science & Applications (ICISA)*, Seoul, 2014.
- [6] EMA, "Adding control to change management:how to assess your requirements.," 2007.
- [7] A. Brown and K. Alexander , "A Best Practice Approach for Automating IT Management Processes," in *2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006*, New York, 2006.
- [8] A. Dedy, S. Yohan and R. Kalamullah, "On Developing Information Security Management System (ISMS) Framework for ISO 27001-based Data Center," in *2018 International Workshop on Big Data and Information Security (IWBIS)*, Jakarta, 2018.
- [9] M. Lacroix and L. P. , "The Change Request Process," in *Proceedings of 2nd International Workshop*, 1989.
- [10] A. Brown and K. Alexander, "A Best Practice Approach for Automating IT Management Processes," in *2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006*, Vancouver, 2006.
- [11] L. Zia, "Optimizing Change Request Scheduling in IT Service Management," in *2008 IEEE International Conference on Services Computing*, New York, 2008.
- [12] C. Kadar, I. Jose, H. Dirk and L. Mario, "Automatic Classification of Change Requests for Improved IT Service Quality," in *2011 Annual SRII Global Conference*, Zurich, 2011.
- [13] Z. Stojanov, D. Dalibor and P. Branko , "An approach in modifying submission phase of change request process," in *2008 6th International Symposium on Intelligent Systems and Informatics*, Subotica, 2008.
- [14] K. A.K, M. F. Bulut and A. Vukovic, "Catalog Recommendation Service for IT Change Request," in *Service-Oriented Computing. ICSOC 2017*, 2017.
- [15] S. Nair, *The Service Desk Handbook: A guide to service desk implementation, management and support*, ITGP, 2020.
- [16] E. Hechler, O. Martin and S. Thomas, *Deploying AI in the Enterprise: IT Approaches for Design, DevOps, Governance, Change Management, Blockchain, and Quantum Computing*, Berkeley: Apress, 2020.

- [17] A. Brown and H. J. , "Reducing the cost of IT Operations-- Is Automation Always the Answer?," in *10th Workshop on Hot Topics in Operating Systems*, Santa Fe, 2005.
- [18] M. Mäkäräinen, "Software change management processes in the development of embedded software: Dissertation," in *VTT Technical Research Centre of Finland*, Espoo, 2000.
- [19] S. Rance, *Tips to help improve change management process*, Toronto: Apart , 2017.
- [20] J. Sauve, R. Rodrigo, M. Antão, B. Claudio, B. Abdel and T. David, "Business-Driven Decision Support for Change Management: Planning and Scheduling of Changes," in *17th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management, DSOM*, Dublin, 2006.
- [21] N. ". Santos and Q. Will, "An Overview of the Change Management Process and Examples of Software to Help Organizations Effectively Manage Change," *GSTF Journal on Business Review (GBR)*, vol. 4, no. 1, 2015.
- [22] K. Dietel, "Mastering IT change management step two: moving from ignorant anarchy to informed anarchy," in *SIGUCCS '04: Proceedings of the 32nd annual ACM SIGUCCS conference on User services*, Baltimore, 2004.
- [23] A. Keller, "Automating the change management process with electronic contracts," in *Seventh IEEE International Conference on E-Commerce Technology Workshops*, New York, 2005.
- [24] R. Reboucas, J. Sauve, A. Moura, C. Bartolini and D. Trastour, "A decision support tool to optimize scheduling of it changes," in *Proceedings of IFIP/IEEE Network Operations and Management Symposium*, Munich, 2006.