2024-10-28

Risk Mitigation in Software Engineering: Practices, Challenges and Opportunities

Siddiqua, Mahpara Sayema

IUB

Risk Mitigation in Software Engineering: Practices, Challenges and Opportunities

By

Mahpara Sayema Siddiqua

Student ID: **2211838**

Autumn,2024

Supervisor

Mahady Hasan,PhD

Associate Professor

Department of Computer Science & Engineering

Independent University, Bangladesh

October,2024

Dissertation submitted in partial fulfillment for the degree of Master of Science in Software Engineering

Department of Computer Science & Engineering

Independent University, Bangladesh

Attestation

I, Mahpara Sayema Siddiqua, hereby certify that the report titled 'Risk Mitigation in Software Engineering:Practices, Challenges and Opportunities' has been prepared and completed by me. The submission is done for the purpose of partial fulfillment requirement for the Master of Science in Software Engineering degree.The support and experienced guidance by Mahady Hasan,PhD as supervisor throughout has been conducive.
I affirm that the contents of this report is original and envisaged through the research outcome.The resources used for the study have been properly acknowledged to the best of my knowledge.

Signature

Date

Name

Acknowledgment

I would like to start with my appreciation to the department of Computer Science and Engineering for providing the opportunity to explore my research interests through the program Master of Science in Software Engineering.

My sincere gratitude to my supervisor Mahady Hasan,PhD, Associate Professor, Department of Computer Science and Engineering at Independent University Bangladesh. While his expertise in academia and guidance has heavily contributed to enriching my interest and knowledge in research, it is his kindness and humble attitude that has made an even lasting effect.

I am thankful to the faculties associated with the Master of Science in Software Engineering program for making the classroom a place for generating ideas, having impactful discussions and overall learning by combining their academic knowledge and practical experience. Even though this has been a mixed experience, I am grateful for the learning opportunities.

Last but not least, my family, for being there by me through thick and thin.

Letter of Transmittal

October,2024

Mahady Hasan,PhD
Associate Professor
Department of Computer Science and Engineering
Independent University Bangladesh

Subject: Graduate Project Report Submission for the completion of Master of Science in Software Engineering Degree

Dear Sir,
I am grateful to be able to submit my Graduate Project Report to fulfill one of the requirements to complete the Master of Science in Software Engineering degree.
The report titled as 'Risk Mitigation in Software Engineering: Practices, Challenges and Opportunities' combines three separate studies that were done focusing on security requirements gathering, risk management practices in software development projects and use of system modeling in software development. The studies were all done on Bangladesh based software firms to gain their perspectives and identify and investigate the challenges and present a guideline addressing those. The studies helped to gather insights on the scenario of the firms based in Bangladesh and contributed to the research experience. To the best of my possibilities, I have tried to ensure that the report reflects the gathered information and perspectives correctly.
I sincerely hope that this report fulfills the requirement for the degree and serves as a resourceful research work for future endeavors.

Sincerely,
Mahpara Sayema Siddiqua
Student ID: 2211838

Evaluation Committee

Supervision Panel

.....	.....
Supervisor	Co-supervisor

External Examiners

.....	.....
External Examiner 1	External Examiner 2

Office Use

.....	.....
Program Coordinator	Head of the Department

.....

Director Graduate Studies, Research and Industry Relations

.....

Library

Abstract

The development of software based systems often deals with unexpected challenges. Risk mitigation plans and practices can play a vital role in dealing with such challenges. The study combines three separate studies that were done focusing on the key areas of software development life cycle (SDLC).Security requirements, risk management plan and use of Model Driven Engineering (MDE) for secure development. Questionnaires prepared based on the literature findings and online surveys were conducted as an attempt to gather information about the standards and practices along with the challenges faced by the practitioners from the perspective of Bangladesh based software firms. The responses were thoroughly analyzed and based on the response analysis a five step plan for risk mitigation is software engineering was proposed.

Keywords: Software Engineering, Risk Mitigation,Software Development,Risk Management, Software Project Development,Software life-cycle, Risk Management Plan,Software Modeling, System Modeling, Model Driven Engineering (MDE), Bangladesh based software firms

Contents

List of Figures

List of Tables

1 Introduction

We live in a world where technology is being used in almost all aspects of our daily life. With the advancement of technology and the world enjoying its blessings there are certain areas of challenges that demand to be addressed. One such challenge is mitigating risks to successfully developing a software project. It requires the crucial task of analyzing the threats and ensuring security of a system [1].

Security, being a non-functional requirement, is often dealt with at the end or later stages of the software development life cycle (SDLC) that costs time and money that sometimes lead to compromising on the quality [2]. There are researches and studies that show that addressing security requirements and challenges at the beginning or requirement stage of SDLC helps to develop a system that is more secure and efficient [3, 4].There are systems that require personal and confidential information (like online banking systems) which need to be protected to ensure the privacy of the user [5].With this requirement security is no longer an issue that can be taken lightly [6] .

It has become essential to successfully develop software based projects within budget and schedule. Software project management is one of the crucial factors for successful software development [7]. Proper management of the software development plays a vital role in the success of a project [8]. Software project management can be sectioned into different parts and one of them is risk management. While successful software development is challenging, implementing proper risk management has proven to be beneficial [7, 9]. Risk management generally refers to identifying the risk factors and analyzing the chances of the risk involved in order to be prepared in advance to take timely measures [10].

To ensure secure development of software using software modeling could be beneficial. In order to have a better understanding of the system prior to its implementation abstract models of software systems are created and systematically transformed to concrete implementations [11].As the model acts as a clone of the developing software and used to test the system it can help identify the flaws in the earlier stage and act accordingly to ensure smooth development. But, over the years practitioners have faced difficulties and certain challenges such as implementation complexity, lack of data, while implementing software modeling [12, 13].

Some of the objectives of this study are as follows:

-To assess the current state of risk mitigation practices in Bangladesh-based software firms.

-To identify, investigate and document the key areas and best practices for risk mitigation in software project development.

-To analyze the challenges and determine the opportunities related to risk mitigation in the context of Bangladesh-based software firms.

-To propose a guideline in order to improve risk mitigation practices within software development life cycle.

1.1 Overview

The study aims to understand the present scenario and the behavior towards the risk management practices of the Bangladesh based software firms. Acknowledging the importance of risk mitigation in a software development project the study intends to determine the practices, challenges, opinions and potentialities. An extensive literature review was conducted in order to determine areas that are most sensitive to risks and plays a significant role in the success of a software development project. Software security requirements in requirement gathering, risk management plan in the software project management plan, the use of model driven engineering (MDE) in software development are the key areas that were chosen as areas of investigation. A research methodology was developed to examine these key areas and three questionnaires were developed with the help of literature for conducting surveys. The questionnaires included open ended questions which opened the path to responses and opinions that are non-biased and gave an honest insight on the thoughts of the responses. The survey included experts such as Software developers, Software Project Managers, Software Test Engineers, Software Architect, Cyber Security Analyst, Solutions Architect etc. The surveys were conducted mostly online through a set of questionnaires using google forms as a platform. A few in person interviews were also conducted. The survey responses were used to gain an insight on the present practices and understand the perspectives of the respondents on those practices, the challenges they come across and the prospect for improvement. A thorough examination and analysis of the responses lead the way to identifying the practices and problems along with the scopes based on which a guideline to mitigate risks in software development projects was proposed. In conclusion, the study encourages the implementation of risk mitigation plans in software development and highlights the importance and benefits of managing potential risks with the inclusion of risk management practices and strategies. Even though the study responses lead to a positive outcome and promising areas of opportunities, it is not beyond limitations. Further research is required to gain a wider perspective and opinions on the implementation of the proposed guideline and to scale its efficiency.

1.2 Contribution of the Research

The study explores the current scenario: practices, challenges and opportunities for the software firms in Bangladesh to understand the risk management plans and strategies to mitigate risks. This study significantly contributes to figure out the trends and patterns of the risk management challenges and gaps in the practices. The survey that was conducted

for this study validates the literature findings and internationally practiced methods and steps to mitigate risks in software development. The survey responses include opinions and suggestions by the experts working in the industry who have practical knowledge and wide experience in the respective fields providing a deep insight of the actual issues and ways to resolve them. The findings from the result analysis could be used to compare with the global trends and practices and figure out the gaps in the risk mitigation strategies in Bangladesh based software firms. It could also be utilized to figure out the presence of any unique scenario that might be interesting to observe. As the study is done focusing on the Bangladeshi software firms, it believed that it would contribute to the literature and will open the doors to further in depth research on mitigating risks in software development projects from the perspective of Bangladesh. Mostly, it is hoped that the study findings would encourage the inclusion of risk management practices in order to benefit from them by mitigating potential risks in software development projects.

1.3 Organization of the Thesis

The introduction that is section 1 consists of the overview of the project that describes the entire concept of the study in brief, discusses the contribution of the project and mentions how the thesis is organized and being managed as a project. Followed by section 2 which includes the literature review of the relevant literature to provide a background of the study that was based on these literature findings. The next chapter explains the research methodology that was followed for this project. Section 4 discusses the survey responses analysis and findings. Section 5 presents the proposed guideline for risk mitigation based on the survey responses. The thesis concludes with Section 6 with a synopsis of the whole project and its outcome and the prospect for future research.

1.4 Management of the Thesis

A work breakdown structure (WBS) and a Gantt chart was prepared for the purpose of managing the thesis. The WBS defines the significant stages of the project and includes the steps required to complete the project. It makes the project goals manageable by breaking the whole project into smaller parts.The Gantt chart represents the plan and time period allocated to complete each task for the defined stages required for the completion of the project. Both the tools are used for the visualization of the project tasks that makes accomplishing the project goals and reaching the milestones easier.

1.4.1 Work Breakdown Structure

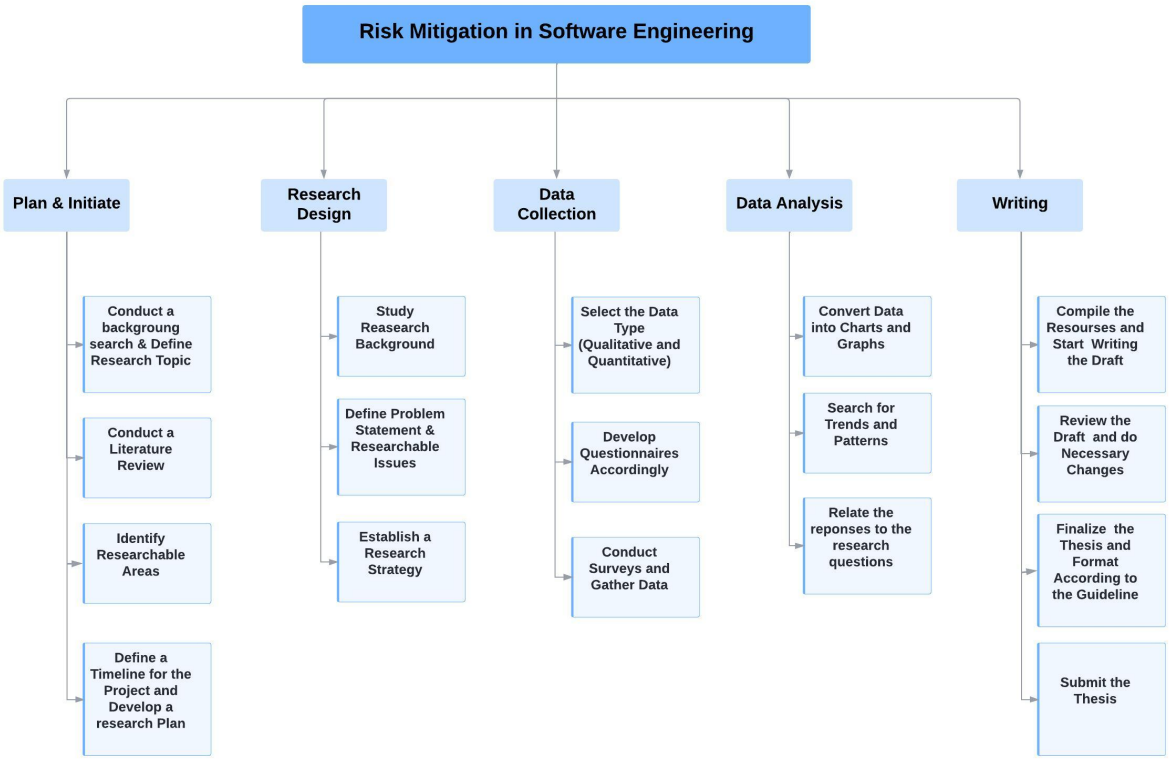

Figure 1: Work Breakdown Structure of the Project

1.4.2 Gantt Chart

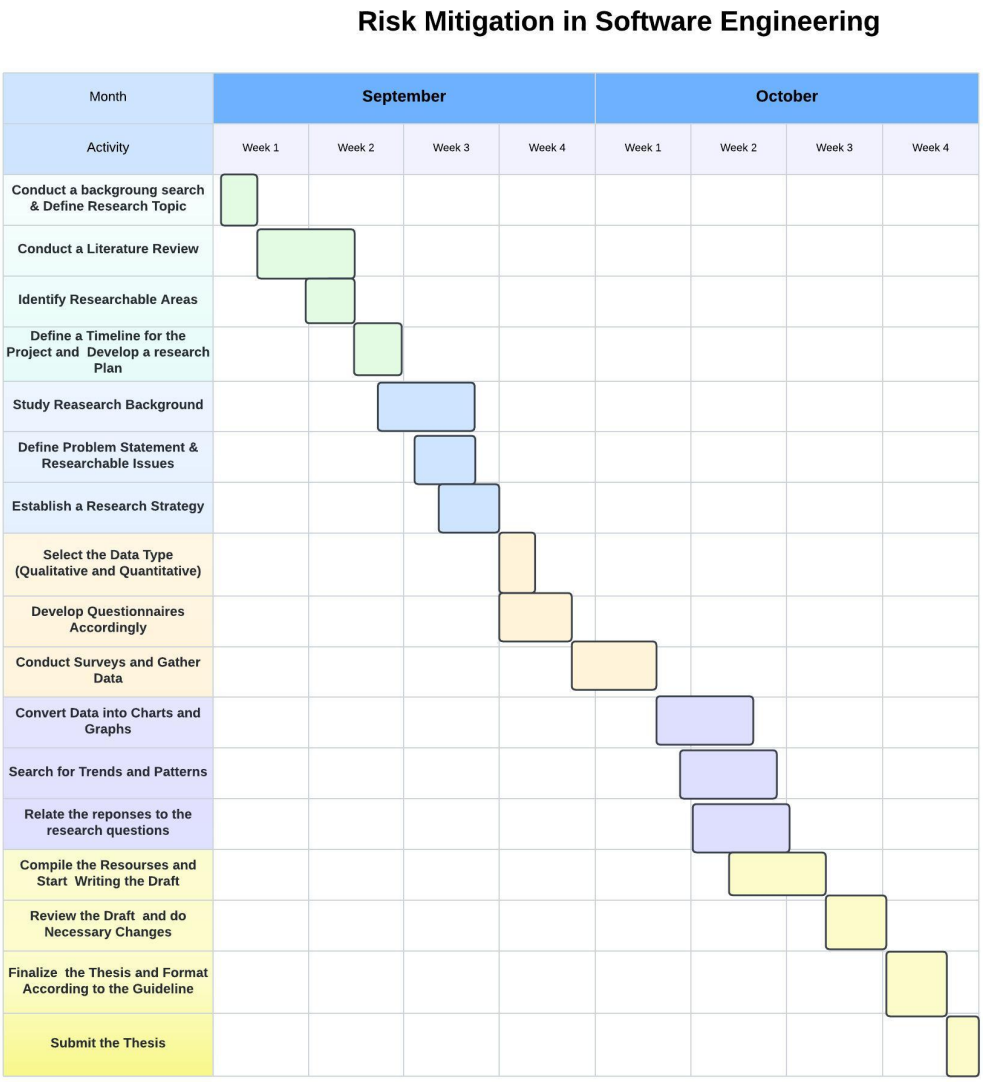

Figure 2: Gantt Chart of the Project

2 Literature Review

Literature review was conducted to gather intensive knowledge on how security requirement engineering, risk management plan and system modeling impacts an organization to achieve their goals and objectives. The in depth literature review, helped to understand the researchable areas and gaps the software companies is currently having in these key area as well an insight on how to address the issues. The literature analysis contributed to gather knowledge on the impacts of risk management practices in the area identified. Implementing models to ensure secure software development, addressing security issues at different stages throughout the SDLC, risk management practices and plans are some of the suggestion that came up. The definition, features, as well as advantages of for implementing security requirement at the initial stage, risk management practices and system modeling in are included in the findings of the literature analysis. Following are some literature reviews that were done from the study.

2.1 Literature Review of Papers

The concept of incorporating security at the early stage of the SDLC is gaining attention day by day. Tuma. K., et al compares 26 threat analysis methodologies that are in use. The authors also discuss the current state and the obstacles of the existing system. Authors stated that the security analysis procedures lacked quality assurance, not well defined and mentioned about the absence of supporting tools and limited validations [1] .Khanneh, S. and Anu, V. et al. In this study they proposed a framework to prioritize criteria for security requirements. To achieve their goal, they used a literature review strategy to establish security requirements engineering and then listed the cutting-edge methods that can be used to apply a recognized prioritized standard for security requirements. Seven techniques for prioritizing security requirements from the literature were found, summarized, and contrasted in their survey [2] .Sravani T. B., et al suggests the paper is to determine the advantages of using Security Requirements Engineering in the beginning stages of software development. Developed two different versions of a similar web application including and without SRE, after which compared the security level for each version using various testing tools. The resulting findings confirm the benefits of early SRE use, with a 38 percent security improvement in the secure version of the software. This real benefit reaches 67 percent for greater vulnerabilities, removing only non-critical and convenient vulnerabilities [3].

Ansari, M. T. J., et al. highlights security threats that occur earlier in the SDLC process, assisting the requirement engineer in gathering the necessary security requirements methodically to assure secure and high-quality software development. The author proposed the STORE methodology which identifies four points—PoA, PoB, PoC, and PoD—for security requirements based on security threats analysis to achieve the author's objective.

Later the author used this methodology in a case study for validation purposes [4]. R. Subha, et al. assesses security challenges throughout the requirement stage. Here the authors compare methodologies like SQUARE and MOSRE with ISTDRE model. During the requirement phase no specific information was not captured adequately, the SQUARE and MOSRE models and ISTDRE model technique fails to erase all virtual security threats. Compare these techniques and show that ISTDRE is more efficient at recognizing security threats [5]. Fabiana. F. S. F., et al Gathered primary and secondary data using different analysis to discuss the importance of security requirements in the current world. Utilizing those data to focus deeply on experimental investigation to understand dubious practices of omitting them from software requirement elicitation/prioritization and the concurrent conditioning of software use [6].

Val´erio et al. conducted a methodical literature review to find risk management gaps, views, and trends in software development projects from 2010 to 2018. According to the analysis, a number of areas still need to be filled, including the requirement for increased performance, team participation, attention to mistakes, identification of decision-making tools, and alignment with corporate strategy. The Study offers insightful information on new approaches and perspectives for risk management in software development [7].The study by Jarrah, M.A., et al. intends to identify the primary factors that contribute to the successful implementation of a project and investigate the role of project risk management as a mediating factor. To examine the impact of project management processes (project initiation, planning, execution, monitoring and control, and project closure) on effective project implementation, a survey was conducted .The analysis of the survey where 96 project managers took part, revealed that project risk management mediates the effect of various project management processes on successful project implementation. It was suggested that project managers should not only focus on implementing standard project management processes, but also on incorporating effective risk management practices to guarantee the success of their projects [8].

C. Basile, B.D. Sutter et al. presents a rationale for adopting and standardizing the protection of software as a risk management process with the increase of Man-at-the-End (MATE) attacks. It presents a Proof of Concept (PoC) decision support system that instantiates many of the discussed constructs, models, and methods and automates many activities in the risk analysis methodology for the protection of software. Although a prototype, the PoC's validation with industry experts indicated that several aspects of the proposed risk management process can already be formalized and automated with the existing toolbox and it can actually assist decision making in industrially relevant settings [9]. Garc´ ıa et al. study focuses on the difficulties that small and medium-sized businesses (SMEs) encounter while applying globally recognized standards and frameworks, such as CMMI, ISO, and PMBOK, due to a lack of financial resources, human resources, skills, and technologies. To fit their particular size and type of organization,

SMEs may need to modify these best practices. Findings from a case study on a tool for fundamental risk management used in two businesses show that both viewed the tool favorably and as a quick way to put software engineering best practices into use [10].
Torchiano, M. et al. constructed studies on Productivity gains, portability, maintainability, and interoperability the advantages of model-driven approaches and software modelling. A sample of 155 Italian software professionals took part in an online based survey and shared exploratory personal opinions. According to the responses model driven approaches make it simpler to achieve improved standards, increased productivity, and platform independence. Better design support, better documentation, better maintenance, and higher software quality were mentioned as benefits of adopting basic modelling. They also mentioned some issues that might be discouraging its adaptation [11]. Ragnhild Van Der Straeten et al. research about the major problems faced when using model-driven software engineering.15 submitted research in a conference, from various countries and institutions, were used as a base to figure out the major problems being faced when trying to apply MDE (Model Driven Engineering). They identified 14 problems, 4 of which, the authors suggested as important Industrial Adoption, Formal Foundations, Scalability Issues, Model consistency and co-evolution [12].
Antonio Bucchiarone et al. outlines what has changed in research in MDE over the last 10 years, what challenges remain and what new challenges have arisen. In 2017 and 2018, two events were held that included experts from industry, academia and the open-source community, focusing on an analysis of the state of research, state of practice, and state of the art in model-driven engineering (MDE). This article reports on the results of those meetings, and presents a set of grand challenges that emerged[13].Mahmood, N. et al analyses that to help companies in SDLC for more precise and defined requirements for secure software development RESMM can play a huge part. A systematic literature review as well as case study in two companies was demonstrated in order to evaluate RESMM. As a result, it found out that RESMM is simple to understand and can able to determine the levels of maturity which will be vital for security requirement engineering in software companies [14]. Oluwafemi O., et al Offers supported standards for extracting software requirements from approved texts. Also offers services and products evaluated by the author in the financial technology sector while checking for compliance with information driven communication protocols and extracting many security standards [15].
Kalle R., et al suggests agile practices were consistently used, and they were seen to be used in conjunction with security practices. The phases of requirement and implementation saw the highest frequency of security-related activity. In general, it was thought that the most significant activities occurred early in the life cycle. The amount of utilization and the perceived security impact of numerous security actions were found to differ to better integrate security engineering operations into software development processes [16]. Manish S. et al Proposes an architectural layer named Fog computing to

address a variety of weaknesses. This architectural layer tends to help to assess security vulnerabilities and to predict the changes of the new paradigm in near future. Here the author elaborates the current research and intense vulnerability analysis focusing fully on scalability, performance and robustness [17]. YUSUF. M., et al proposes SRERM a model that organizations can use to get guideline from to measure their security requirements engineering (SRE) level of readiness. From the available resources, only 12 security requirements categories were created and 76 best practices were identified to build a SR-ERM. Two case studies were conducted to evaluate the proposed model in a real-world environment and few modifications were done to identify the readiness levels of SRE in the software industry [18].

Malik. N. A. M., et al conducted a systematic literature review from the available resources. 20 different SRE approaches were selected and that were compared to different technical parameters. The study results based on the comparative analysis and analytical evaluation shows that most of the SRE approaches do not meet the security standards and incorporate formal methods [19]. Hugo V., et al presents the findings of a systematic mapping study on security requirement engineering in the frame of reference of agile methods. Authors grouped those case studies and examined the limitations identified in the studies when dealing with security requirements in agile contexts. The results demonstrate that far more initiative should be put into empirically evaluating existing approaches, as well as room for future study to address the recognized limitations [20].

I. Lee proposed a cyber risk management framework with a focus on the cyber ecosystem and cyber risk quantification. The proposed framework which is believed to complement the existing frameworks categories factors affecting cyber risk into four layers: the cyber ecosystem layer, the cyber infrastructure layer, the cyber risk assessment layer, and the cyber performance layer that can be expanded to the qualitative risk assessment in the absence of quantitative historical data[21]. A.K. Sangaiah, O.W. Samuel et al. conducted a study that aims at hybridizing fuzzy multi criteria decision making approaches for the development of an assessment framework. It can be used to efficiently identify and rank notable software project risks that contribute to the decision making. The effectiveness of the proposed approach results was compared with earlier methods and the assessment shows that the integration of fuzzy approaches could be effective and as well accurate compared to existing approaches for evaluating software project risks [22].

Breno Gontijo Tavares et al. evaluates failures of software projects that are caused by improper risk management. Agile methodologies, which are now quite popular, lack specific risk management tactics. A tool was designed to address this issue by providing risk management techniques within an iterative life cycle. An experiment with agile professionals was done to test the tool's effectiveness, and the results showed improved risk response planning without additional time investment [23]. Wan Husin et al. acknowledges the dangers associated with distributed software development (DSD), including

communication breakdowns, time zone difficulties, and cultural obstacles. By highlighting communication, emphasizing component identification, and encouraging stakeholder cooperation, the article seeks to improve the risk management framework for DSD. The findings point out communication as a significant concern and is incorporated into the framework for better risk management across remote contexts [24].

Bj¨ ornsdottir et al. focuses on the progress made in benchmarking risk management, with a specific emphasis on ISO risk management systems. This study proposes a two-step benchmarking technique for evaluating the efficacy of ISO systems, with the objective of identifying latent risk factors and potential areas for development. The methodology was successfully employed in six firms that have obtained ISO certification, demonstrating its efficacy in the identification of risk factors and concerns that were previously overlooked[25]. Xingyuan Chen and Yong Deng et al. talks about the value of software risk management and the difficulties in performing software risk assessments. They provide a brand new data-driven software risk assessment approach called DDERM that measures uncertainty. The methodology can be used to solve additional risk management issues that are proven with a case study[26].

Masso et al. evaluate the current level of software risk management in the developing software industry, by conducting a systematic literature review (SLR). The results show a focus shift from integrated risk management processes to particular process activities. The analysis also reveals a lack of standardization in the use of risk management models or frameworks. These findings highlight the need for more thorough and rigorous research in software risk management to address the industry's increasing complexity and problems [27]. Abioye et al., address the problem of lack of effective and reusable risk management techniques for software projects. For software initiatives, the proposed solution is a life-cycle approach employing an ontology-based risk management framework. To identify, model, and rank risks information was collected from the literature, domain specialists, and practitioners. The implementation results demonstrate reduced costs, on-time delivery of high-quality projects, and positive feedback from software developers who find the framework to be extremely beneficial for their daily software development activities [28].

Jolak, R. et al. conducted a case study by observing two Model Based Engineering (MBE) projects for two months that focused on the distribution of the total software development effort over different development activities. After observing the challenges during the execution of projects reported by the developers it was found that collaboration and communication activities required the majority of the efforts spent. Tool-related challenges were also mentioned and the study found that in one case 22 percent and in another case 30 percent effort was spent on tools [29]. Akdur, D. et al. have done a world-wide survey, designed and conducted to understand the state-of-the-practice in software modelling practices in the embedded software industry. The survey received 627

responses from 27 different countries. The paper aims to shed light on the helpful effect of understanding and characterizing the cross-factor analysis of software modelling versus practitioner demographics [30].

Mohamed, M.A. et al. have done Systematic Literature Review (SLR), a study is presented that focuses on identifying and classifying the recent research practices used in Cyber physical Systems (CPS) development by applying MDE approaches. 140 research papers (2010–2018) were evaluated in order to provide a guidance for researchers and practitioners to identify appropriate tools and languages according to the system requirements filling in the gap of having no standard guideline available [31]. Ameller, D. et al. survey which is interview-based, findings are presented to understand how the Non-Functional Requirements (NFRs) are handled in Model-Driven Development (MDD). The survey includes practitioners from 18 different companies in 6 European countries. The results show that MDD is considered to be a complex process with almost no tool support for NFRs which leads to generating manual support compromising software development maintainability [32].

Badreddin, O. et al. identify trends in software design practice as well as adoption trends for modelling languages like UML. They designed a survey with two stages of data collection, split ten years apart, to achieve this. The survey analysis describes practice trends, such as upward, downward, positive, negative, and unexpected trends. According to the survey, both the usage of formal and domain-specific modeling as well as the broad practice of modeling is increasing to some extent [33]. Troya, J. et al. reviewed and analyzed defining uncertainty in software models thoroughly. The article offers a classification framework that makes it possible to compare and categories current proposals, analysis their state of research, and identify emerging trends. The concepts, notations, and formalisms used to express uncertainty in software models, as well as the application domains and methodologies constantly being proposed, were compiled in this survey [34].

Feichtinger, K. et al. described in their work transforming variability models, including potential utilization scenarios, necessary capabilities and difficulties . Their development is based on observations, research, and input from variability modelling experts. This study gave an overview of the essential elements of our envisioned transformation strategy for variability models, which would be based on meta-models and generic transformation operations [35].Khan, R.A. et al. develop a Software Security Assurance Model (SSAM) to assist Global Software development (GSD) vendor organizations in measuring their readiness towards the development of secure software. Systematic literature review (SLR) and empirical study in the industry, the software security contributions, security challenges and their practices for global software development (GSD) vendors. Identifying the security contributions, risks, and practices in the context of SDLC [36].

Marc I. Kellner et al. provide an overview of work being conducted in the MDE field. It shows why software modelling is needed, in what situations it can be used and how to

efficiently apply the various techniques to a project. The authors have used the papers submitted in a special issue of the Journal of Systems and Software, The authors conclude by saying the approach for simulation modelling (or MDE) is always situational depending on the kind of project [37].van der Linden, D. et al. have constructed a mixed-method study that was conducted to understand the practitioners requirements. After analyzing the responses received from 104 participants, statistical and factorial analysis was used to explore potential correlations between the importance of different requirements as perceived by practitioners and the demographic factors (domain, purpose, topics). The results show the practitioners' requirements have been evaluated and shifted from the previous decade [38]. Basha, N.M.J. et al. explore the benefits, drawbacks, challenges, and resources of model-driven development, a significant area of model-based software design. The techniques for the difficult problems in the MBSD are discussed in this study. This study will consider this MBSD as a method for creating domain-specific components that will lead to the Generic MDD. The concerns and difficulties discussed in this work can help the research initiators conduct additional MBSD research [39].
Robert France et al. gives an overview of current research in MDE and discuss some of the major challenges that must be tackled in order to realize the MDE vision of software development. Using their own expertise on the topic they explained the current state of MDE and possible futures. They encourage more research and use of MDE in our current systems to achieve the vision they have [40].

3 Research Methodology

The research is designed to identify the practices of the software firms in Bangladesh in order to mitigate risks, the challenges present while at it and the key factors influencing those. The research also sheds light on the use of software modelling in software development and the approach towards it by the Bangladesh based software firms. Several surveys were conducted that included 03-25 software firms in Bangladesh. The surveys were based mostly online through questionnaires via google forms with an exception of few surveys that were conducted via interviews. A qualitative analysis of the gathered data was done in order to identify the practices along challenges mentioned and provide a novel guideline to mitigate risks in the overall software development and project management of software development life cycle.

3.1 Purpose of the Study

The software development life cycle (SDLC) consists of multiple stages each partially or fully depending on the other. It is important to complete each phase as precisely as possible before stepping to the next one in order to avoid revisiting and redoing the work that has already been done. The study focuses on considering security measurements at the requirement gathering stage, implementing risk management techniques for software project development and increasing the use of software modeling for developing software with the aim of mitigating risks in software development and making the SDLC time and cost effective. Based on the determined current practices and challenges from literature and survey, the study provides novel guidelines for the Bangladeshi software firms to encourage incorporating the risk mitigation approaches for improving the software development along with future opportunities.

3.2 Data Collection

Survey: Data were collected from several (3-25) Bangladesh based software firms mostly located in Dhaka. Multiple (3) questionnaires were prepared for the three separate studies based on the literature findings which also included some open questions targeting certain areas of scope.

Software developers, Software Project Managers, Software Test Engineers, SAQ Engineers, Software Architect, Junior Software Engineer, Cyber Security Analyst and Lead ML Engineer and Solutions Architect took their time to give their valuable input on the surveys on their respective area of expertise.

Data Sampling: Purposive sampling method was used for conducting the survey as the studies required insights and inputs from experts in specific criteria and roles. For security requirements and challenges the security analyst/support engineer , quality assurance

engineer, solution architect were surveyed. For the risk management practices project managers shared their perspective through the survey. For practitioners' challenges in software modeling software developers, test engineers, solution architects, project managers were targeted for the survey.

Sample Participants demographic

Requirement Engineering Challenges for Security Intense System Development

Job Title	Level of Expertise	Year(s) of Experience
QA Support Engineer	Intermediate	2 Years
Quality Assurance Engineer	Intermediate	2 Years
ML Engineer and Solution Architect	Intermediate	1 Year

Table 1: Sample Participant Demographic of Security Requirements Gathering Survey

Implementing Risk Management for Software Project Development in Bangladesh based Software Firms

Year(s) of Experience	Number of Project Manager(s)
1-3 Years	7
4-5 Years	6
6-8 Years	5
9-10 Years	2

Table 2: Sample Participant Demographic of Risk Management Practices Survey

Practitioners' Challenges in Software Modeling

Job Title	Year(s) of Experience	Type of Product Developed
Software Developer	Less than 5 years	Enterprise Applications
Project Manager	Less than 5 years	Mobile Applications
Solution Architect	5-10 years	Web Applications
Software Test Engineer	5-10 years	IoT (Internet of Things) Applications

Table 3: Sample Participant Demographic of Practitioners' Challenges in Software Modeling Survey

3.3 Measurement and Analysis

The surveys were intended to understand the current scenario of the software development firms in Bangladesh. The surveys specifically focused on software security requirements implementation, incorporating risk management in software project management and the use of software modeling. The questionnaires were designed with a focus to identify the following:

Present Practices: The common and used practices in the software development by the Bangladesh based software firms.

Issues and Challenges: The problems with the running practices and the kind of challenges they are being faced because of them.
Trends and Patterns: The trends and patterns observed from the survey responses will be used to identify the practices and their impact on the SDLC.
Opinions and Suggestions for enhancement: The industry experts point of view and suggestion to address the issues.
Implementation Barriers/ Challenges: The challenges/ barriers to implement the suggested improvements according to the industry experts.

3.4 Research Method

The data collected through the surveys and interviews were thoroughly studied to gather information and analyze trends and patterns of the responses. The questionnaires for each study were prepared with the focus to identify the current practices and challenges in the respective fields and their impact in the software development life cycle in Bangladesh. The open questions on the questionnaire lead the way to understanding the perspectives of the professionals working in several roles such as requirement engineer, project manager, and software developer. The responses were studied to perceive the scenario and their inputs and suggestions were kept in mind while preparing the guidelines. The guidelines address the issues mentioned in the survey responses and intend to mitigate the risks by making the timely use of the resources.

4 Result Analysis

4.1 Result and Discussion

For each study certain research questions were determined. The questionnaires were designed including other questions surrounding those research questions with the intention of having a broad perspective of the respective areas. The combination of the wider perspectives of each area will eventually lead to understanding the overall impact of the practices and challenges in the SDLC resulting in a better opportunity to address them in order to mitigate risk in the SDLC.
Some research questions and responses are discussed below.

- **Requirement Engineering Challenges for Security Intense System Development**

A survey was conducted that included project managers, system analysts, tester/ technical lead, security advisor from three different companies (Bangladesh). The survey focused on the following areas of concern along with other relevant research questions.
RQ1: Do software companies apply Security Requirements Engineering at the beginning of the software development life cycle?
RQ2: Does Security Requirements Engineering help to develop a quality product?
The responses were analyzed and based on that certain challenging areas were identified for security requirements and implementation, the main concern being not prioritizing security requirements, not conducting any risk assessment at the initial stage and not following any framework or policies for gathering security requirements, failing to meet deadlines, budget issues and changes in requirements.

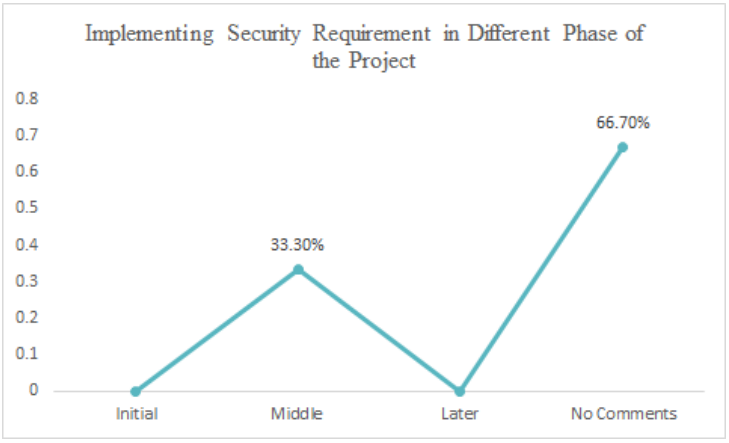

Figure 3: Stages of Security Requirement Implementation

The responses shown in figure 1 represent the stage of performing security analysis. It can be seen that none of the companies do security requirements at the initial phase, contradictory to the literature that recommends conducting security requirements analysis in the initial phase to overcome requirement engineering challenges for security intense system development [1].

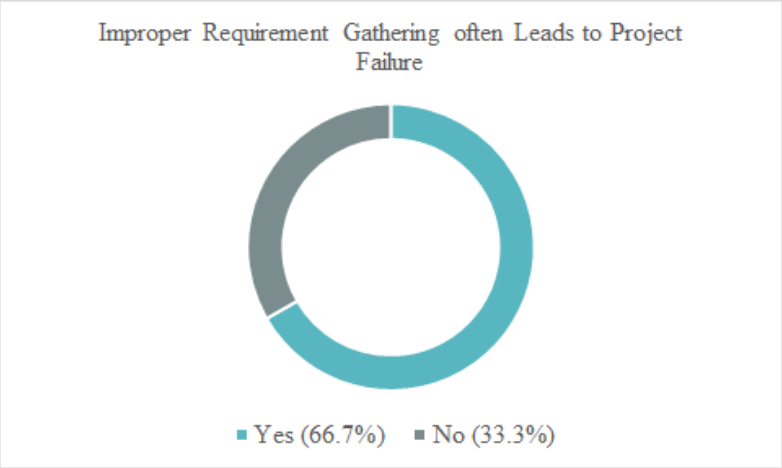

Figure 4: Impact of Improper Requirement Gathering

The respondents were asked about their opinion on the importance of requirement gathering. According to the survey responses 66.7 percent of the respondents think improper requirement development often leads to project failure. As failing to do a proper requirement analysis leads to implementations that might require changes at the later stage which is often not time and cost effective.
Based on the study's findings, the surveyed companies do not follow any sort of guideline or framework and they do not conduct security requirements at the beginning stage of the project. They mostly follow an ad hoc process which is not beneficiary to their security requirement engineering. According to the participants, they do not follow any particular framework or any policies and some of them do not cross-check operational and functional requirements against safety requirements resulting in poor quality and less secured software.

- **Implementing Risk Management for Software Project Development in Bangladesh based Software Firms**

In order to understand the current scenario of the software firms based in Bangladesh in respect to risk management implementation in software project management, a survey (online) was conducted. From the responses it can be seen that even though risk management is not widely practiced in project management, the majority of the respondents

are in favor of including risk management as it can help with reducing cost and utilizing time efficiently. The study tries to find answers to the following research questions (RQ) to explore the challenges and opportunities along with the current practices of risk management.

RQ1: What are the current practices and challenges of implementing risk management in software project management?

RQ2: How is implementing risk management beneficial for software development?

The respondents were asked about the risk management practices that are currently being used by them.

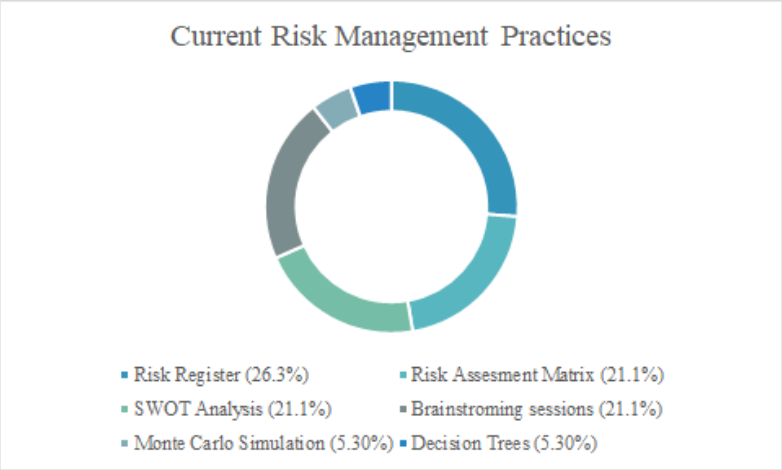

Figure 5: Current Risk Management Practices

The responses included Risk Register (26.3 percent), Risk Assessment Matrix (21.1 percent), SWOT Analysis (21.1 percent), Brainstorming Sessions (21.1 percent), Monte Carlo Simulation, and Decision Trees. While talking about the implementation challenges of risk management practices 38.1 percent of the respondents mentioned lack of awareness or understanding risk management concepts as a major reason. The challenges also included difficulty in identifying and prioritizing project risks (19 percent), resistance from team members or stakeholders (19 percent), Limited availability of resources such as financial, human resources, tools (9.5 percent), inadequate risk management training (9.5 percent, time constraints.

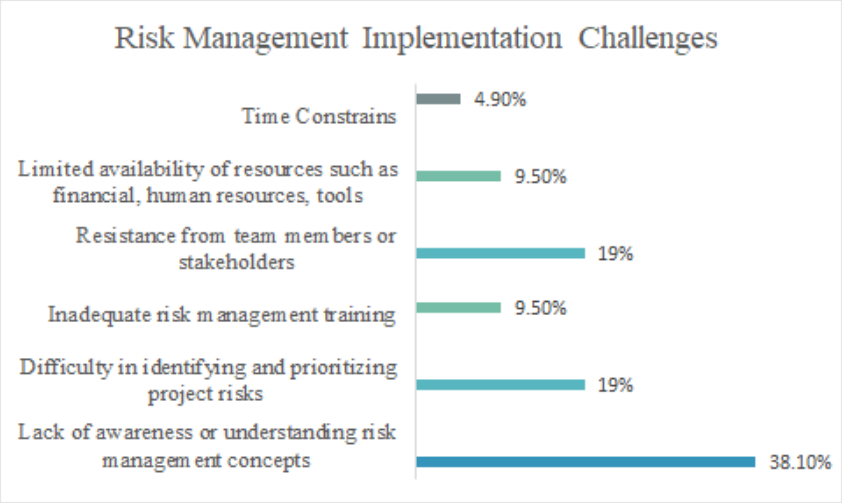

Figure 6: Risk Management Implementation Challenges

The challenges also included difficulty in identifying and prioritizing project risks (19 percent), resistance from team members or stakeholders (19 percent), Limited availability of resources such as financial, human resources, tools (9.5 percent), inadequate risk management training (9.5 percent, time constraints. The following figure represents the frequency of conducting risk assessment during the SDLC.

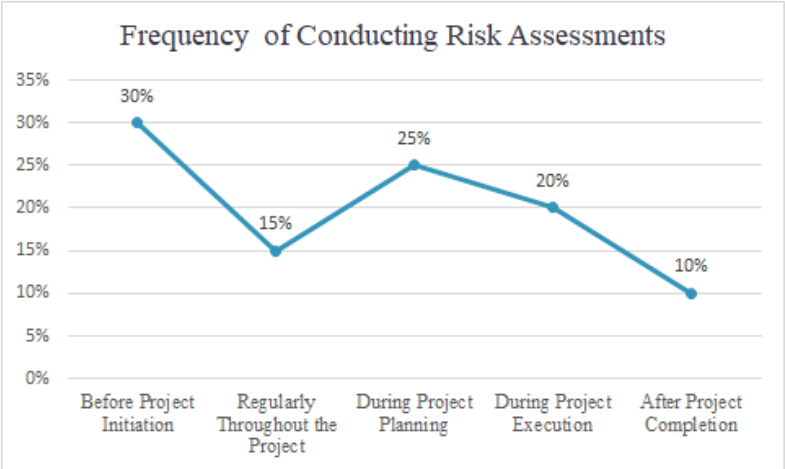

Figure 7: Frequency of Conducting Risk Assessments

Only 15 percent responded said they regularly do risk assessment throughout the project and 30 percent said that they do a risk assessment before initiating the project. 20 percent responded that they conduct a risk assessment during the project planning and during project execution while the rest said they perform risk assessment after the project execution.

The study surveyed the perspectives of project managers about the benefits of implementing risk management in software development project management. Among the respondents 31.8 percent strongly agreed and 36.4 percent agreed that implementing risk

management practices helps to identify and address potential security vulnerabilities in software development.

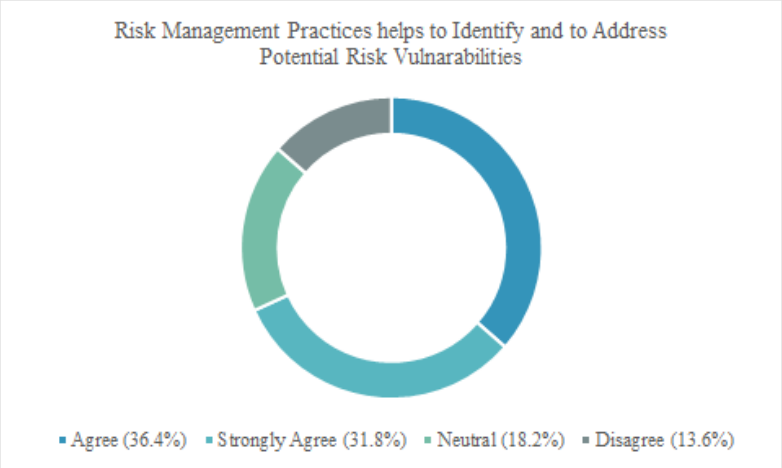

Figure 8: Impact of Risk Management Practices in identifying risk vulnerabilities

Identification of risk vulnerabilities at the early stage of software development can be beneficial for project management as it will help to take measurements to address the risk vulnerabilities in a timely manner to build a secure software which will have a positive impact on the success of the project.

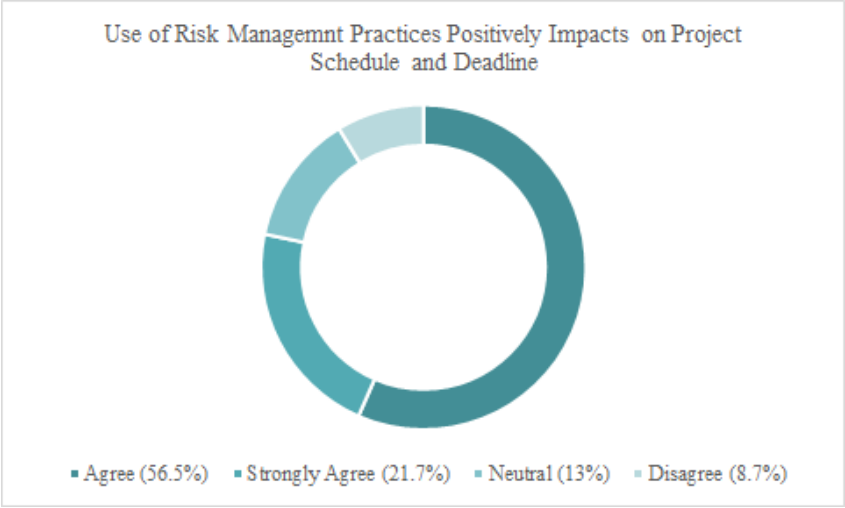

Figure 9: Impact of Risk Management Practices on Project Schedule

According to 78.2 percent (21.7 percent strongly agree, 56.5 percent agree) respondents the overall schedule and deadline adherence is positively impacted by the use of risk management practices.

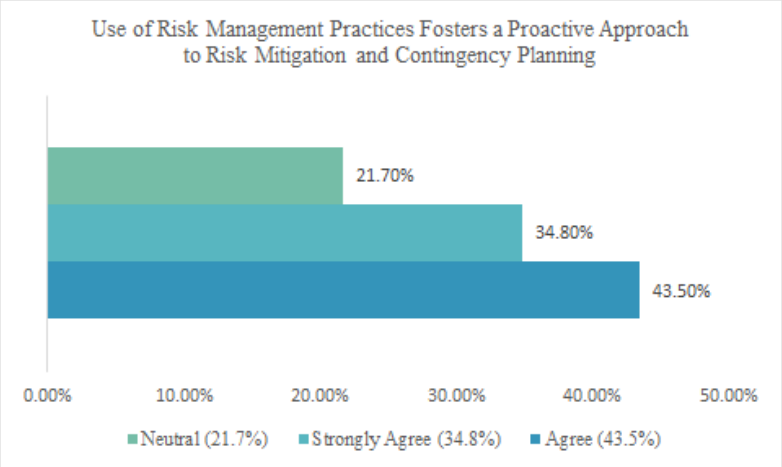

Figure 10: Impact of Risk Management Practices in Risk Mitigation and Contingency Planning

These responses in the figure indicate that incorporating risk management practices in the software development project management fosters a proactive approach to risk mitigation and contingency planning which to which 34.8 percent of respondents strongly agree and 43.5 percent of the respondents agree.
Overall, it can be concluded that most of the project managers taking part in the survey acknowledge that implementing a risk management plan in the software development project management plan can mitigate risk and have a positive impact on the success of the project life cycle.

- **Practitioners' Challenges in Software Modeling**

A survey was conducted to understand the perspective of the practitioners about software modeling in model driven engineering (MDE) and challenges that included software developers and software project managers working in different software firms in Bangladesh. The survey was based on both questionnaire and interviews focusing on the following areas of concern along with other relevant research questions.

RQ1- What levels of difficulties do practitioners encounter with the various types of software modeling challenges?

RQ2- What unique challenges might practitioners go through when modeling software systems?

The survey responses reflect different levels of difficulties for practitioners. For Modeling language 50 percent said that it is too complicated to understand while 75 percent mentioned lack of proper documentation/ tutorial/ support as a difficulty. Managing language complexity increases the level of challenges for practitioners according to 33.3 percent respondents. Extending Modeling languages adds another level of difficulty for 35.7 percent. Domain-specific modeling environments and developing formal modeling languages were mentioned as difficulties by 42.9 percent.

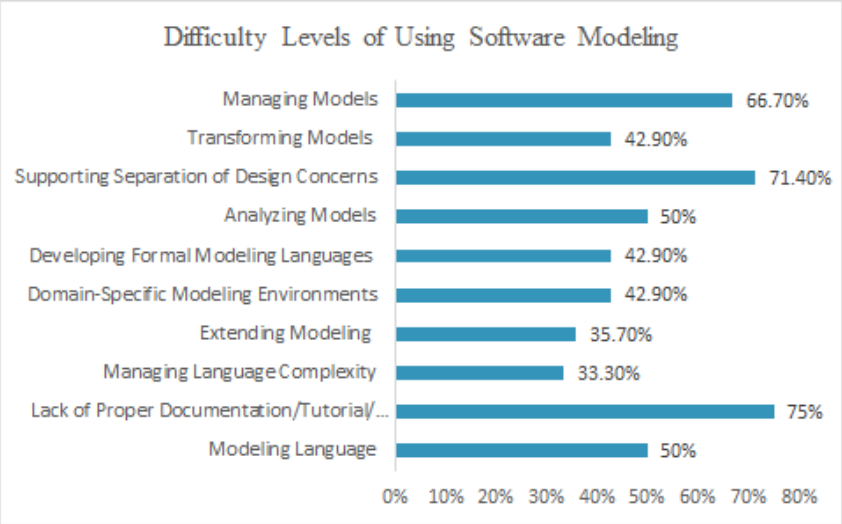

Figure 11: Difficulty Levels of Using Software Modeling

Among the respondents 50 percent said Analyzing models, 71.4 percent said Supporting separation of design concerns, and 42.9 percent mentioned Transforming models and 66.7 percent said Managing models were difficult.

The survey results show that there are several difficulties for practitioners in software modeling. The difficulty of creating a model that is both accurate and efficient, was the biggest issue noted. Additionally, practitioners believed that a significant barrier was

the lack of knowledge and experience in software modeling. The task's complexity, the requirement to coordinate several stakeholders, and the lack of resources were additional difficulties.

According to the survey results in the following figure, software modeling for software organizations is fraught with a number of difficulties for practitioners. Different modules of the model couldn't be merged due to inconsistency in design, No prior use cases or data to help develop the model seem to be the two major issues.

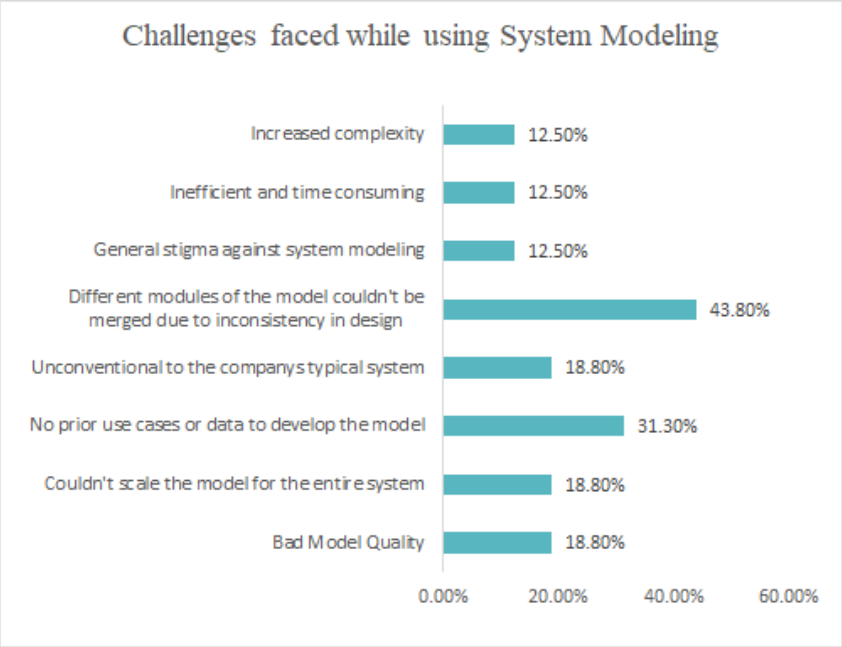

Figure 12: Challenges of using System Modeling

Bad Model Quality, Couldn't scale the model for the entire system and Unconventional to the company's typical system are frequently mentioned factors. General stigma against system modeling, Inefficient to use. Takes more time and increased complexity are some of the other problems that were mentioned. As per the literature, these identified challenges are not unique to practitioners based in Bangladesh.

43.8 percent of the respondents agree that the merging issue of different models due to inconsistency in design is a major challenge. When it comes to the scalability issue of the model, the area of scope needs to be explored and a proper investigation needs to be done. The numbers from the results in the next figure show that 31.3 percent of the respondents strongly agree that system modeling does help in development while 25 percent agree.

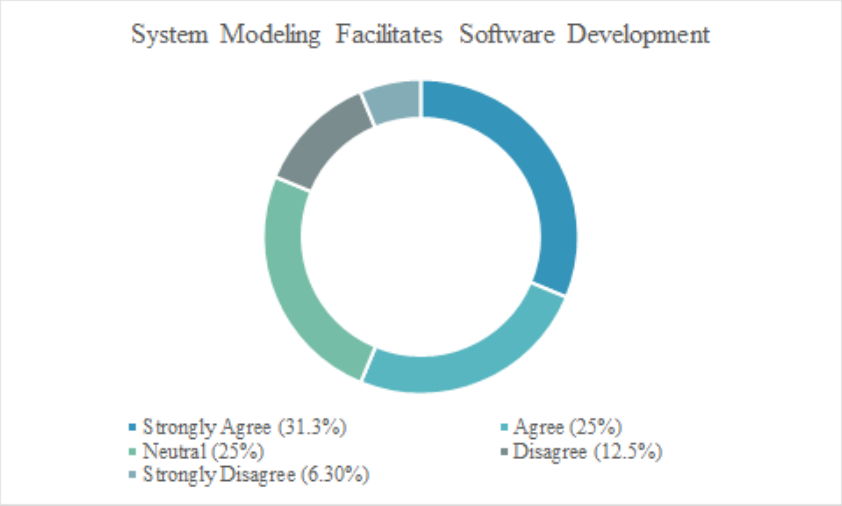

Figure 13: System Modeling Facilitates Software Development

This is a positive fact for the future of MDE in Bangladesh. The problems being faced now can be overcome with research and development. But, before all that there needs to be a drive towards that goal. This response paints an optimistic future where the current challenges will be replaced by benefits and opportunities.
As observed from the survey responses below in the figure, most of the developers now try to use system modeling in their projects. 20 percent developers said they implement system modeling while 60 percent said that they sometimes include system modeling in their projects.

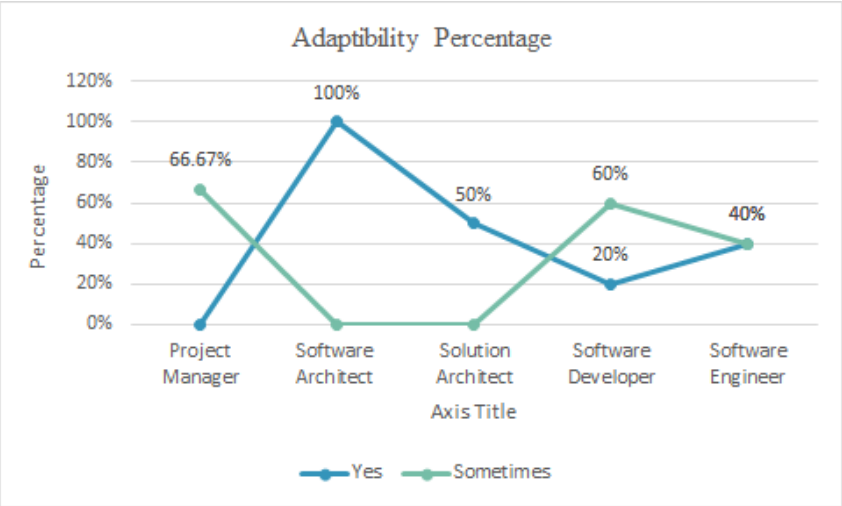

Figure 14: System Modeling Adaptability Percentage

This is a sign of positive change. Practitioners are getting familiar with the advantages of system modeling and so it is being tried in projects. If not for the whole system, developers try to use MDE in individual modules to properly build them and later merge with the overall project. Even though the usage percentage of implementing system mod-

eling is promising, the percentage of the system modeling not being used is not neglect able.

Practitioners believe that although software modeling is a crucial activity, doing it well is challenging. Practitioners suggested that the organization provide their software modeling teams' greater resources and training, as well as better cooperation across stakeholders, to overcome these issues. Additionally, experts urged the development of improved technologies to streamline the procedure. Better tools were also requested by practitioners to make the procedure simpler and more effective.

Incorporating practices that contribute to the betterment of different phrases of the SDLC will benefit the whole life cycle of the software development project. The three studies focused on identifying the practices and challenges of security requirement analysis in requirement gathering, risk management plan in project management and use of software modeling in software development; three different and important stages. All three studies reflect the answers to having a successful project and how including risk assessment at the initial stage, having a risk management plan and using software modeling can not only make the SDLC time and cost effective but also paves the way to developing a secure and high-quality software.

Being able to gather those and scan them through the lens of risk mitigation in the overall software development can lead to significant improvement and opportunities for the software firms in Bangladesh.

5 Proposed Guideline

Mitigating risk in the software development life cycle is a crucial but essential task that is directly related to the success of the software project development.A framework to integrate risk management practices in different phrases throughout SDLC can make the pivotal task of risk mitigating feasible. Security requirements should be integrated with software development life cycle (SDLC) at the early stage as developers need to be well aware of the requirements as the implementer of those. A risk management plan should be included in the software project management plan to have strategies sorted should any potential risk arises. The software development plan should also include secure development and security testing as an additional part of the SDLC risk mitigation plan. The main objective is to suggest an effective guideline based on the survey responses on how the organization in Bangladesh can gain insights and improve their secure software development life cycle. The aim is to explore the risk and security management practices and propose a risk mitigation model/ guideline for software development which can be followed by the Bangladesh based software firms. It should be noted that the proposed guideline is a general guideline based on the literature findings and survey responses that might not adhere to the principles of all software firms in Bangladesh. Proposed below is a five step guideline that could significantly contribute to mitigate risks in software project development-

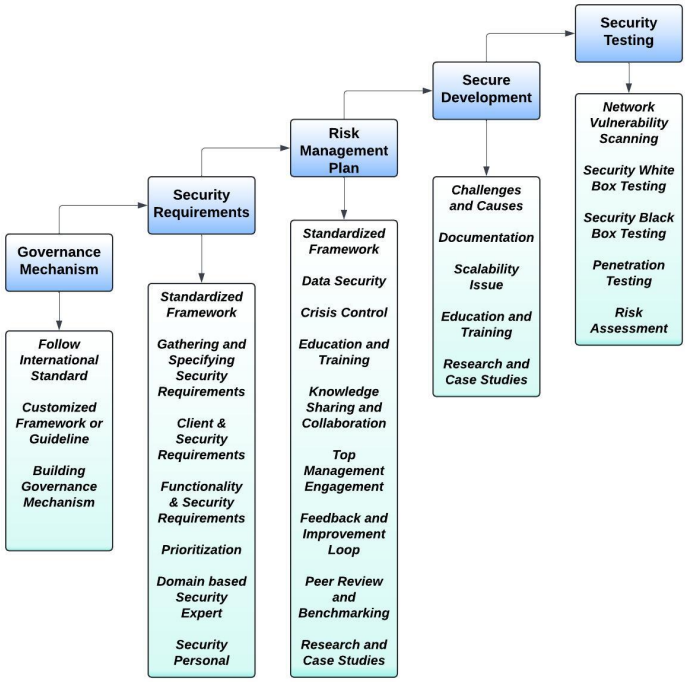

Figure 15: Proposed Guideline to Mitigate Risks in SDLC

The five steps of the guideline is discussed in details below.

Governance Mechanism

Several organizations got benefit by practicing a well-established framework such as NIST's Secure Software Development Framework (SSDF). Bangladesh based software firms can also be benefited by the following

- **Follow International Standard:** Bangladesh based software firms can gain knowledge from OWASP (Open Worldwide Application Security Project), SAFEcode etc as they discuss the software vulnerability in detail, provide security engineering training and regularly update them.

- **Customized Framework or Guideline:** Based on those frameworks and guidelines the organizations in Bangladesh can customize their own secure development policy to establish the security requirements.

- **Building Governance Mechanism:** The standards such as ISO 27001, SOC 2 which provide specific security requirements by which organizations can build their governance mechanism.

Security Requirements

The basic requirements are: functional, nonfunctional, and derived/domain specific. Software security is widely considered as a non-functional requirement. According to the survey responses, software security requirements can be considered into these three categories and not just non-functional. Security requirements specify what a system must do and be in order to perform securely, just as performance requirements specify what a system must do and be in order to perform according to specifications. User access control, data integrity, authentication, session time-out and secure password lockouts are examples of functional security requirements. In other hand Audit logs lie to non-functional security requirements. User ID, PIN number, account lockout, prevention against SQL injection etc. should fall under derived security requirements. According to the literature and surveys it is beneficial to integrate security requirements with every part of development but currently organizations mostly treat them as non-functional requirements only and this mindset might be affecting the schedule, budget and the overall quality of the software; impacting the success of the software development project.

The practices that could enhance and solidify the software security requirements (but not limited to these) are-

- **Standardized Security Requirements Framework:** A standard requirements framework, policy or guideline needs to be followed to streamline the elicitation of security requirements.

- **Gathering Security Requirements:** The security requirements need to be collected and specified before initiating the software development.

- **Specific Security Requirements:** Security requirements need to be as specific as possible so that there is no confusion afterwards.

- **Client and Security Requirements:** The customers should be made aware of and asked about their security requirements from the beginning of the project.

- **Functionality and Security Requirements:** Basic Security requirement related to functionality of the software should be maintained.

- **Security Requirements Prioritization:** Prioritization needs to be defined for the different type of security requirements.

- **Domain based Security Expert:** Relevant security domain experts should be involved in the requirement collection phase.

- **Security Personal:** A separate security personal can be delicately involve to review the security requirements during the development phase.

Risk Management Plan

Risk management is crucial for any company or organization, whether it's dealing with workplace accidents, staying current on regulatory compliance issues, or resolving unemployment claims. While implementing risk management might seem like a complex task and often considered nonessential, it has the potential to be a powerful instrument for project management. Risk management programs offer an opportunity to cut expenses, save time, and acquire a competitive edge in addition to eliminating risks. Employing and training substitute workers, modifying work schedules, looking into incidents, and putting in place safety protocols, and corrective actions, hardware and software failures,

human mistakes, viruses are all examples of potential risks while developing a project. The following suggestions to address the concerns with risk management techniques in software project development inside Bangladesh-based software firms could be used:

- **Standardized Risk Management Framework:** A standardized methodology or framework for risk management suited for Bangladesh based software firms that includes best practices, guidelines, and templates for risk identification, assessment, mitigation, and monitoring should prove beneficial and work as an encouragement.

- **Data Security:** The security of the data should be ensured and plans such as the Information Technology Risk Management (ITRM) program that aims to identify, foresee, and eliminate risks before they become issues should be created in case of an IT emergency.

- **Crisis Control:** Crises such as cyber-attacks, regulatory scrutiny, workplace harassment, compliance issues, litigation which arise without prior notice should be included in a crisis management plan (having available consultants, lawyers, IT specialists, and other experts available) along with strategies to handle the anticipated risks.

- **Education and Training:** Plan risk management workshops, seminars, and training sessions, risk management courses by academic institutions and business organizations for software project teams, project managers, and senior management that includes skill development and certification will encourage participation.

- **Knowledge Sharing and Collaboration:** A network or a platform where software companies might exchange case studies, lessons gained, and experiences in risk management will promote corporate cooperation and knowledge sharing to advance the adoption of successful strategies and tactics.

- **Top Management Engagement:** As senior management's knowledge of risk management plays a crucial part in a project's success, they should be updated with the statistics and case studies that demonstrate the advantages of effective risk management.

- **Feedback and Improvement:** Risk management strategies and policies should be regularly evaluated with the help of feedback from stakeholders and software experts to ensure that the standardized risk management framework and training programmers are kept current and effective to iteratively enhance them.

- **Peer Review and Benchmarking:** Software companies should be encouraged to take part in peer evaluations and benchmarking activities to compare their risk management procedures to industry norms by acknowledging and rewarding companies that succeed at risk management and ensure optimal practices.

- **Research and Case Studies:** Researching on projects that explore the difficulties and possibilities unique to Bangladesh's software sector, Documenting case studies that showcase the success tales of businesses that have successfully implemented risk management techniques and show observable effects will motivate the inclusion risk management plan.

Secure Development

It is tough to implement the whole idea of secure development and practice it in reality as the developer's focus might shift to write the code more accurately rather from the security point of view. Due to the fact most of the modern software is not developed from scratch and the developers also can't limit their focus on code writing only. Model driven engineering (MDE) can play a significant role in this case. Software modeling can ensure smooth implementation of requirements, help make important decisions about components and develop secure software. Software modeling is mostly used for creating secure software as they are created and tested before implementation. Using system modeling as a tool through different stages of SDLC can be beneficial for making necessary modifications from time to time which is required to develop a secured software.

While software modeling is not widely used in Bangladesh based software firms there is a positive changing trend that can be observed from the survey responses. To encourage the use of Model Driven Engineering/ Software modeling to utilize its benefits for software development certain steps can be taken such as-

- **Challenges and Causes:** Determine the challenges of implementing MDE and identify their root causes through extensive research and take steps to address the challenges.

- **Documentation:** Ensure proper step by step documentation of implementations by the practitioners for comparative studies.

- **Scalability Issue:** Techniques such as modular engineering principles, incremental processing could be utilized in order to address the scalability issues of MDE.

- **Education and Training:** Education events such as seminars, workshops, and training sessions can promote the use of MDE with a clear understanding of its utility and success.

- **Research and Case Studies:** Real life use cases and supporting research data can also encourage the implementation of software modeling in certain settings.

Security Testing Security testing is a type of Software testing that checks the vulnerabilities of the overall software/ application and ensures that the software is secured from unauthorized access, malicious attacks etc. by providing a forewarning to the developers to mitigate those before launch. Some major goals of Security testing are to identify threats and vulnerabilities, to detect the security risk and to help developers to fix codes that might have potential security threat.
Confidentiality, Integrity, availability, Authentication, Authorization and Non-repudiation etc. are the basic principles of security testing. Some of the common type of security testing which can be followed by the organization in Bangladesh are-

- **Network Vulnerability Scanning:** Security scanning for the network is a way to check and identify potential vulnerabilities and threats which allows the organization to take action in order to mitigate the potential risks before they become harmful for the system.

- **Security White Box Testing:** The white box testing allows the tester to do an in-depth inspection of the codes along with source codes and infrastructure designs and documents to identify any internal issues.

- **Security Black Box Testing:** Through a black box testing the tester can be sure of the systems external behavior and ensure the systems safety from cyber-attacks.

- **Penetration Testing:** Penetration testing is executed to identify any security loopholes that the system might have and can be taken advantage of by any invader or attacker.

- **Risk Assessment:** Risk assessment is used to identify potential risks along with case studies and scenarios in order to prepare the team should such risks arise.

- **IT Security Posture Assessment:** Security posture assessment helps to identify the security threats of the software firms' security policies and strategies by performing extensive evaluation.

- **Ethical Hacking:** Ethical hacking such as the Read Teaming concept is used to identify a system's weakness by attacking the system intentionally and assessing the security level of the system or software.

Governance Mechanism, Security Requirements, Risk Management Plan, Secure Development and Security Testing each plan an important role in the success of developing a secure software system. Each phrase works as a step in a staircase which leads the way to the landing that is a successful and secure software project development. Being able to combine them together in a framework and implementing them in a software development project can increase the chances of climbing up the staircase by facing the least amount of threats. While it is not practically possible to predict and eliminate all kinds of risks in a software development project, a plan or a guideline to mitigate them can work wonders by having the team prepared with the knowledge of possible risks and strategies to handle them should they arise.

6 Conclusion

The potential for risk mitigation in Bangladesh based software companies appears to be promising. The respondents participating in the survey strongly agreed that integrating risk management techniques into software project development positively influences the project success, decision-making, meeting deadlines and overall efficiency. This provides a strong ground for encouraging the risk management practices to mitigate the risks in software project development for the management.
The responses on the survey also helped to identify the areas of possible development and certain areas might require specialized approaches to investigate further in order to identify potential challenges. To mitigate the risks as much as possible the strategies should be proactive. Risk identification and assessment,risk management and response planning,educating and training the team and including the people from all stages involved in the software development project would encourage the risk management practices and increase the chances of figuring out the most efficient risk management practices and techniques for the software development project and the software firms.

6.1 Sustainability

Sustainability requires developing projects that are of high quality in order to make them long lasting. The goal of this project is exactly that. Incorporating risk mitigation practices throughout the project development increases the systems sustainability by building a software or systems which are highly secured and meets the quality standards. It also ensures less waste as risk mitigation practices in each phrase most likely paves the way to a successful development of software projects.

6.1.1 Social Impact

The impact of a software in the societal context is an essential criteria of concern. How the software development project affects the people related to it and what kind of impact the complete project will have on the society is a sensitive area that should be taken under careful consideration. A framework or policy that exclusively deals with the social impacts of a software development project and educating the people related to the project will help determining the social impacts effectively.

Security Requirements for Security Intense Systems
The demand for secured systems is increasing and security requirement consideration has become essential. According to the survey conducted it leads the production of a software that is secure and efficient to use.

Risk Management Practices in Project Management

The survey responses indicate that a risk management plan ensures safety of the individuals associated with the software development project. Having a guideline to follow, ensuring step by step implementation of the risk management practices will result in less trouble for the people involved in the development project and produce a system that adheres to the requirements and requires less corrections in the later stages.

Practitioners' Challenges in Software Modeling

Even though there are challenges but with proper training and guideline Practitioners will find the implementation of systems favorable for them with the use of software modeling according to survey responses.

6.1.2 Environmental Impact

Developing projects and systems that are sustainable has become essential. Incorporating green coding, green programming, and mindful use of resources and the concept of energy efficiency will add up to the sustainability factors. Reducing e-waste by utilizing the resources to the fullest and developing systems that are secure, easy to update and perform efficiently is crucial to environmental sustainability. Adopting a formal environmental policy or procedure and training staff on environmental considerations will help to achieve the sustainability of the environmental impacts of software development projects.

Security Requirements for Security Intense Systems

Developing software that is secure and efficient to use will gain the trust of the users. They will likely start using them to the full potential, reducing the use of paper. The software will also be utilized fully and will not contribute much to the e-waste.

Risk Management Practices in Project Management

Risk management practices ensure the team is prepared to deal with the potential risks and ensure the factors vulnerable to risks are being covered. According to the survey, this leads to smooth implementations and development of software projects.

Practitioners' Challenges in Software Modeling

System modeling contributes to the sustainable factor as well as software development as it can use software models for similar kinds of projects that have been previously developed and implemented, as reflected through the survey. It ensures the sustainable use of the resources along with proper implementation.

6.2 Feasibility

The proposed guideline in this project is a combination of practices that are widely used globally and are considered beneficial for mitigating risks. The implementation of those practices in the software project development from the perspective of Bangladesh based software firms should be an uncomplicated and cost effective task. Software firms might

need some modification or customization to meet their policy or goals but the proposed general guideline is a compilation of practices that are presently being used worldwide.

6.2.1 Technical Feasibility

Technical feasibility is an important factor for a software development project. To evaluate the technical feasibility it is essential to ensure the access to necessary tools and platforms. The tools and technical support that are used for risk mitigation in different stages are commonly available and easy to access. The data collection method and the data analysis techniques are other major key factors that's feasibility should be ensured. This included surveys (both online and in person) and interviews. In order to keep the risk mitigation practices updated constant evaluation is required. To serve the purpose it is necessary to conduct these surveys from time to time and to gain an honest perspective participation of the people involved in the project development is essential. The lack of knowledge or unwillingness on using the tools for risk mitigation purposes could be a barrier. Encouragement on learning and utilizing the tools and resources will also contribute to the technical feasibility of the project.

Security Requirements for Security Intense Systems

The technical feasibility of gathering security requirements since the initial stage of SDLC is achievable. From the survey responses it has been reflected that incorporating security requirements from the early stage is technically more feasible than dealing with them at the later stage. The clear concept about the security requirements leads to implementations that meet the expectations and functionalities.

Risk Management Practices in Software Project Management

Including a risk management plan in order to encourage risk management practices might appear to be a complicated task but it is technically rather feasible. The survey responses show that the project managers actually believe that risk management practices are beneficial for software development projects and there are ready tools that are already in use. It is the willingness of the responsible individuals that needs to be ensured.

Practitioners' Challenges in Software Modeling

Software modeling or system modeling deals with abstract models of a system that provides the chance to decide on the final outcome before stating the actual implementation. Using a similar kind of model that already exists and making required modifications makes the task less complicated and also ensures the success of the final product. According to the survey responses, the majority of the practitioners believe that using system modeling facilitates software development.

6.2.2 Economic Feasibility

The economic feasibility of a project includes the economic viability of the project and the outcome related to it. Rough Order of Magnitude (ROM) contains the estimated cost of the project and its parts. The cost related to the tools, platforms and technical support that is required to implement risk mitigation practices is critical for economic feasibility. The appointment of human resources or security experts or personnel dedicated to ensuring implementation of risk mitigation practices and the cost related to it also needs to be considered. The cost of data collection methods (surveys and interviews), tools that are being used for data analysis and documentation are crucial factors of a project's economic feasibility.

Security Requirements for Security Intense Systems

As a non-functional requirement security is often dealt with at the later stage and sometimes at the final stage. Which most of the time leads to last minute modifications and redoing certain steps exceeding the budget and deadline. The survey responses indicate that such situations can be avoided by including security requirements since the beginning of the software development project.

Risk Management Practices in Software Project Management

A risk management plan includes a budget allocated for risk management practices according to the risk assessment of the software development project. This contributes to the economical feasibility of the project by mitigating risk and the additional cost related to it. The survey responses show that this proactive approach is beneficial to deal with incidental expenses.

Practitioners' Challenges in Software Modeling

Software modeling requires tools and individuals who are experts in using them. Acquiring the tools and training the people related to the software development project to use those might add some cost but the proper utilization of the concept will ensure that it is worthwhile. It contributes to reducing cost by reusing resources for the implementation purpose. According to the survey conducted the practitioners think that even though there are some difficulties using system modeling can be advantageous.

6.2.3 Operational Feasibility

In order to assess a project's successful implementation operational feasibility is necessary. Implementing risk management practices might seem like a complicated task but mostly these are simple steps and practices combination of which benefits the overall project. The proposed guideline which consists of risk mitigation practices and steps in different stages, is based on the perspectives of the experts involved in software development projects. So, the practical implementation is expected to contribute to the project's efficiency with minimal changes and additions to the project development environment

and practices. Ensuring the implementation of the risk mitigation practices will positively impact the software development project by developing a secure and efficient system.

Security Requirements for Security Intense Systems

According to the survey responses as well as literature findings, taking security requirements under consideration since the beginning of the project has proven to be beneficial for the overall project as it produces products that are accurate to the requirements and secure to use.

Risk Management Practices in Software Project Management

The survey participants who are experienced project managers responded that risk management practices help identifying risk vulnerabilities in a software development project and provide a chance to take necessary steps to mitigate those. Conducting risk assessments throughout the project on a regular basis has a positive impact on the project's successful implementation.

Practitioners' Challenges in Software Modeling

Software modeling could be a challenging task but with proper utilization of the resources it can contribute to secure development of the software. A lot of models are already prepared and tested to be correct as they have been implemented and using such models for similar kinds of projects requires minimal modifications and increases the chances of proper implementation of the system.

6.3 Ethics

It is essential to maintain ethical boundaries while conducting research. The participant gave their consent to use the data provided by them as a response to the survey questions. Their identity and other personal information are being kept confidential to ensure that the data analyzing was not influenced by the respondents' role or identity. The data that was collected through the survey were only used for the sake of this research and with the best intentions to understand the present condition of the software firms in order to determine the scopes of improvements.

Data Security and Privacy through Security Requirements

The development of the software project should be done keeping in mind the societal impact and individual well-being. With the advancement of technology it has become essential to ensure the safety of data and personal information. The world has become technology driven and it is now almost impossible to not use software for work or personal reasons. Developing a system that takes the security issues to the fullest consideration and implements the security requirements since the initial stage of the development produces systems that are well secured and less susceptible to cyber-attacks.

Risk Assessment through Risk Management Plan

Assessing the potential risk of the software development project and taking necessary

steps in order to mitigate the risks of ten serve as a safety measure for the project development team. It often saves the team from legal actions as all the criteria are taken under consideration in the risk assessment. It also helps the team to prepare for any potential critical situations and ensures their safety.

Well-built and User Friendly Secure System through Software Modeling

The software modeling ensures the system is well built and secured and delivers a product that meets the requirements of the client. Using system modeling and doing security testing it beforehand ensures proper implementations of the required system. This also saves time and cost along with reuse of resources. It helps to access the user friendliness of the system and provides a chance to make sure that it is easily accessible by the client and inclusive to the users.

6.4 Project Summary

Three separate surveys were done in order to find the current practices and the challenges the software firms in Bangladesh are facing. The surveys focused on security requirement gathering, risk management in software development project management and software modeling in software development. The project aims to provide a guideline to mitigate risks in the software project development by understanding the present scenario and determining the opportunities. People of significant roles such as project managers, software developers, software test engineers, software architects, solution architects etc. took part in the survey and shared their insights. Based on the responses analysis and literature finding a five step guideline has been proposed to mitigate risks in the software development.

6.5 Future Work

The project has a lot of scope for future work which could increase the potentiality and enhance the efficiency of risk mitigation in software development projects. The surveys that were done for three separate studies were based on literature findings. A second round of survey based on the responses could lead the way to a much concrete understanding and reasoning and opinion. Also, the surveys had only a few number of respondents due to time constraints. Conducting surveys on a wide range of people working in the industry will help to gather a large amount of data to analyze and significantly contribute to the literature and proposed framework. The guideline that has been proposed is only based on the analysis of the literature findings and survey responses. Practical implementation of the guideline is necessary to determine the efficiency and improvement rate and figure out the weak points or gaps and the opinion of the industry experts.

References

[1] K. Tuma, G. Calikli, and R. Scandariato, "Threat analysis of software systems: A systematic literature review," *Journal of Systems and Software*, vol. 144, pp. 275–294, 2018.

[2] S. Khanneh and V. Anu, "Security requirements prioritization techniques: A survey and classification framework," *Software*, vol. 1, no. 4, pp. 450–472, 2022.

[3] S. T. Bulusu, R. Laborde, A. S. Wazan, F. Barrere, and A. Benzekri, "Applying a requirement engineering based approach to evaluate the security requirements engineering methodologies," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, pp. 1316–1318, 2018.

[4] M. T. J. Ansari, D. Pandey, and M. Alenezi, "Store: Security threat oriented requirements engineering methodology," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 2, pp. 191–203, 2022.

[5] R. Subha and A. Haldorai, "An efficient identification of security threats in requirement engineering methodology," *Computational Intelligence and Neuroscience*, vol. 2022, no. 1, p. 1872079, 2022.

[6] F. F. S. Flores and S. R. de Lemos Meira, "Ethical software engineering: a critical review about software engineering in face of security requirements in the iot/ioe society," in *2021 IEEE International Systems Conference (SysCon)*, pp. 1–8, IEEE, 2021.

[7] K. G. O. Valério, C. E. S. da Silva, and S. M. Neves, "Risk management in software development projects: systematic review of the state of the art literature," *International Journal of Open Source Software and Processes (IJOSSP)*, vol. 11, no. 1, pp. 1–22, 2020.

[8] M. A. Jarrah, J. Baker, and I. Altarawneh, "Toward successful project implementation: Integration between project management processes and project risk management," *Problems and Perspectives in Management*, vol. 20, no. 3, p. 258, 2022.

[9] C. Basile, B. De Sutter, D. Canavese, L. Regano, and B. Coppens, "Design, implementation, and automation of a risk management approach for man-at-the-end software protection," *Computers & Security*, vol. 132, p. 103321, 2023.

[10] Y.-M. García, M. Muñoz, J. Mejía, G.-P. Gasca, and A. Mireles, "Application of a risk management tool focused on helping to small and medium enterprises implementing the best practices in software development projects," in *World Conference on Information Systems and Technologies*, pp. 429–440, Springer, 2018.

[11] M. Torchiano, F. Tomassetti, F. Ricca, A. Tiso, and G. Reggio, "Relevance, benefits, and problems of software modelling and model driven techniques—a survey in the italian industry," *Journal of Systems and Software*, vol. 86, no. 8, pp. 2110–2126, 2013.

[12] R. Van Der Straeten, T. Mens, and S. Van Baelen, "Challenges in model-driven software engineering," in *International conference on model driven engineering languages and systems*, pp. 35–47, Springer, 2008.

[13] A. Bucchiarone, J. Cabot, R. F. Paige, and A. Pierantonio, "Grand challenges in model-driven engineering: an analysis of the state of the research," *Software and Systems Modeling*, vol. 19, pp. 5–13, 2020.

[14] M. Niazi, A. M. Saeed, M. Alshayeb, S. Mahmood, and S. Zafar, "A maturity model for secure requirements engineering," *Computers & Security*, vol. 95, p. 101852, 2020.

[15] O. Olukoya, "Assessing frameworks for eliciting privacy & security requirements from laws and regulations," *Computers & Security*, vol. 117, p. 102697, 2022.

[16] M. Zarour, M. Alenezi, and K. Alsarayrah, "Software security specifications and design: How software engineers and practitioners are mixing things up," in *Proceedings of the 24th International Conference on Evaluation and Assessment in Software Engineering*, pp. 451–456, 2020.

[17] M. Snehi and A. Bhandari, "Vulnerability retrospection of security solutions for software-defined cyber–physical system against ddos and iot-ddos attacks," *Computer Science Review*, vol. 40, p. 100371, 2021.

[18] Y. Mufti, M. Niazi, M. Alshayeb, and S. Mahmood, "A readiness model for security requirements engineering," *IEEE Access*, vol. 6, pp. 28611–28631, 2018.

[19] M. N. Anwar Mohammad, M. Nazir, and K. Mustafa, "A systematic review and analytical evaluation of security requirements engineering approaches," *Arabian Journal for Science and Engineering*, vol. 44, pp. 8963–8987, 2019.

[20] T. Li and Z. Chen, "An ontology-based learning approach for automatically classifying security requirements," *Journal of Systems and Software*, vol. 165, p. 110566, 2020.

[21] I. Lee, "Cybersecurity: Risk management framework and investment cost analysis," *Business Horizons*, vol. 64, no. 5, pp. 659–671, 2021.

[22] A. K. Sangaiah, O. W. Samuel, X. Li, M. Abdel-Basset, and H. Wang, "Towards an efficient risk assessment in software projects–fuzzy reinforcement paradigm," *Computers & Electrical Engineering*, vol. 71, pp. 833–846, 2018.

[23] B. G. Tavares, M. Keil, C. E. Sanches da Silva, and A. D. de Souza, "A risk management tool for agile software development," *Journal of Computer Information Systems*, vol. 61, no. 6, pp. 561–570, 2021.

[24] W. S. W. Husin, Y. Yahya, N. F. M. Azmi, N. N. A. Sjarif, S. Chuprat, and A. Azmi, "Risk management framework for distributed software team: A case study of telecommunication company," *Procedia Computer Science*, vol. 161, pp. 178–186, 2019.

[25] S. H. Björnsdottir, P. Jensson, S. E. Thorsteinsson, I. M. Dokas, and R. J. de Boer, "Benchmarking iso risk management systems to assess efficacy and help identify hidden organizational risk," *Sustainability*, vol. 14, no. 9, p. 4937, 2022.

[26] X. Chen and Y. Deng, "An evidential software risk evaluation model," *Mathematics*, vol. 10, no. 13, p. 2325, 2022.

[27] J. Masso, F. J. Pino, C. Pardo, F. García, and M. Piattini, "Risk management in the software life cycle: A systematic literature review," *Computer standards & interfaces*, vol. 71, p. 103431, 2020.

[28] T. E. Abioye, O. T. Arogundade, S. Misra, A. T. Akinwale, and O. J. Adeniran, "Toward ontology-based risk management framework for software projects: an empirical study," *Journal of software: Evolution and Process*, vol. 32, no. 12, p. e2269, 2020.

[29] R. Jolak, T. Ho-Quang, M. R. Chaudron, and R. R. Schiffelers, "Model-based software engineering: A multiple-case study on challenges and development efforts," in *Proceedings of the 21th ACM/IEEE international conference on model driven engineering languages and systems*, pp. 213–223, 2018.

[30] D. Akdur, V. Garousi, and O. Demirörs, "Cross-factor analysis of software modeling practices versus practitioner demographics in the embedded software industry," in *2017 6th Mediterranean Conference on Embedded Computing (MECO)*, pp. 1–5, IEEE, 2017.

[31] M. A. Mohamed, G. Kardas, and M. Challenger, "Model-driven engineering tools and languages for cyber-physical systems–a systematic literature review," *IEEE Access*, vol. 9, pp. 48605–48630, 2021.

[32] D. Ameller, X. Franch, C. Gómez, S. Martínez-Fernández, J. Araújo, S. Biffl, J. Cabot, V. Cortellessa, D. M. Fernández, A. Moreira, *et al.*, "Dealing with non-functional requirements in model-driven development: A survey," *IEEE Transactions on Software Engineering*, vol. 47, no. 4, pp. 818–835, 2019.

[33] O. Badreddin, R. Khandoker, A. Forward, O. Masmali, and T. C. Lethbridge, "A decade of software design and modeling: A survey to uncover trends of the practice," in *Proceedings of the 21th acm/ieee international conference on model driven engineering languages and systems*, pp. 245–255, 2018.

[34] J. Troya, N. Moreno, M. F. Bertoa, and A. Vallecillo, "Uncertainty representation in software models: a survey," *Software and Systems Modeling*, vol. 20, no. 4, pp. 1183–1213, 2021.

[35] K. Feichtinger and R. Rabiser, "Towards transforming variability models: Usage scenarios, required capabilities and challenges," in *Proceedings of the 24th ACM International Systems and Software Product Line Conference-Volume B*, pp. 44–51, 2020.

[36] R. A. Khan and S. U. Khan, "A preliminary structure of software security assurance model," in *Proceedings of the 13th International Conference on Global Software Engineering*, pp. 137–140, 2018.

[37] M. I. Kellner, R. J. Madachy, and D. M. Raffo, "Software process simulation modeling: why? what? how?," *Journal of Systems and Software*, vol. 46, no. 2-3, pp. 91–105, 1999.

[38] D. van der Linden, I. Hadar, and A. Zamansky, "What practitioners really want: requirements for visual notations in conceptual modeling," *Software & Systems Modeling*, vol. 18, pp. 1813–1831, 2019.

[39] N. M. J. Basha, S. A. Moiz, and M. Rizwanullah, "Model based software development: issues & challenges," *Special Issue of International Journal of Computer Science & Informatics (IJCSI), ISSN (PRINT)*, vol. 2, no. 1, p. 2, 2012.

[40] R. France and B. Rumpe, "Model-driven development of complex software: A research roadmap," in *Future of Software Engineering (FOSE'07)*, pp. 37–54, IEEE, 2007.