J. Electrical Systems 20-7s (2024): 2280-2293

[1] Akhil M V *
[2] Jagathy Raj V P

Causes of Reworks in IoT Product Development in Startup Companies: A Delphi Study

Journal of Electrical Systems

Abstract: - Demand for Internet of Things (IoT) products is increasing worldwide, and many startups are involved in IoT product development. Reworks in product development are significant problems these startups face, impacting their productivity and long-term survivability. The repercussions of reworks and the need to avoid or minimize them in IoT product development in startups make it crucial to find the factors leading to reworks. This paper presents a Delphi study conducted among thirty-three experts working in IoT product development in startup companies to identify the factors causing reworks in IoT product development in startups. This study identified forty factors that lead to reworks, and categorized them under nine groups. This study sheds light on the important factors that lead to reworks in IoT product development in startups. The findings of this study will be helpful to startups working in IoT product development to be aware and mindful of major rework-causing factors.

Keywords: Delphi Method, Delphi Study, IoT Product Development, Reworks, Startups, Product Development.

I. Introduction

Internet of Things (IoT) based products find wide application in different sectors like healthcare, agriculture, education, automobiles [1]–[7]. Startup companies have many opportunities in IoT product development due to the scope for further innovation due to rapid technological advancements and the rise in demand for niche IoT products that large companies may not be attending. Startup companies that work on developing IoT products not only have many opportunities but also face many challenges. With the fast technological advances, the life cycle of IoT products is decreasing, necessitating rapid product development. Only companies that can cycle through product development quickly and deliver quality products that are more advanced and useful than their competition can hope to survive in this era of Industry 4.0. Product development teams will have to be more flexible and accommodating to late change requests from customers and management. These change requirements and other complications bring about one of the most critical but under-addressed challenges in product development: the reworks.

Rework is defined as "the unnecessary effort of re-doing a process or activity that was incorrectly implemented at the first time" [8]. Reworks present considerable challenges to product development and can even have catastrophic consequences [9], [10]. Reworks usually lead to significant cost, quality, and schedule problems in product development projects and negatively affect employee morale [11]–[14]. Foreseeing rework is challenging because of the complexity and interdependent nature of events that precede it [15], [16] and the uniqueness of each project [17]. Even though many studies have tried to uncover the causes and effects of reworks and on practices and policies aimed at reducing reworks, it remains one of the most enigmatic problems in product development projects.

Rework is a critical factor that affects product development companies and can significantly influence the performance and even the survival of startup companies [18], [19]. The effect reworks have on companies' performance has interested practitioners and researchers alike. Previous studies have tried to find the factors that lead to reworks and develop policies and practices that can help reduce reworks in projects [20]. However, rework remains a big problem for companies developing complex products like IoT devices, and it is a factor that can cause the failure of a project, especially for startups where their access to resources is very limited. There is a need for more studies to unearth the causes of reworks, especially in the context of startups, to mitigate its adverse impacts on product development projects. This research endeavours to find the causes of reworks in startup companies developing IoT products by conducting a Delhi study among experts who have experience in and are also working in IoT product development in startup companies.

The remainder of this paper is systematized as follows. Section II discusses the significance of reworks in projects. Section III describes the methodology, the Delphi method adopted for conducting this study. Section IV details how

[1] School of Management Studies, Cochin University of Science and Technology, Kochi, Kerala, India. akhilmaliekkal@gmail.com
[2] School of Management Studies, Cochin University of Science and Technology, Kochi, Kerala, India. jagathyraj@gmail.com
* Corresponding Author Email: akhilmaliekkal@gmail.com

the study was conducted, and section V explains the results and discusses their significance in relation to the existing body of knowledge. Section VI concludes the paper and discusses the scope for future work.

II. Literature Review

IoT products are usually a network of interconnected devices with sensors, processing capability, software, and user interface components that communicate with and exchange information with other devices that are part of its system. The IoT product development projects are usually complex, and as in the case of the any complex product development projects, they are also prone to reworks. The need to rework exists in almost all projects due to various reasons like changes in requirements, errors in work done, and obsolescence. Reworks can trigger unexpected and problematic dynamics like the formation of a vicious cycle of rework that can jeopardize the projects [10]. The reasons for reworks might be internal or external to the project system. External causes can be changes in the external environment that demand changes in the work already done [11]. Internal causes may be work done wrongly and other quality issues [8]. Incorrect understanding or lack of common understanding among stakeholders in charge of ensuring the completion of the project about its various aspects, like the project objectives, scope, and design, can lead to mistakes that require rework [8], [11], [12], [21]. These faulty understandings are usually the results of problems in planning, coordination, and communication, which are major factors that result in reworks, as they often lead to eleventh-hour changes [22]–[24].

Concurrent engineering or design, where tasks are performed parallelly, is practiced across various sectors in the product development process [25]. A high degree of concurrence is a factor capable of reducing the lead time but has high risks of giving rise to reworks. As the degree of concurrence increases and there is high schedule pressure, the effort devoted to quality assurance will be less. Thus, there is a higher probability for errors and quality issues to go unnoticed, which, if and when detected in a later phase, would have amplified the damages and rework requirements, thus compromising the project performance. Delays, inaccuracies, and incompleteness of information transfer within a project reduce the probability of detecting errors and rework requirements [25]–[27].

Misinterpretation of customer requirements and communication barriers with customers can be a significant source of faulty work [28]. Mistakes and the need for changes in the initial design stage of a project, if not discovered early on, can lead to significant deterioration of project performance due to rework because of the dynamic complexity created by the interaction between different phases in the project [29]. Projects are also prone to the fallibility of the people who work on them, who change the rules and regulations to suit them, and who fail to learn from past mistakes. Research has shown that project personnel tend to hide unfavourable news of errors or violations from senior management to avoid getting blamed. Thus, the errors will go unattended for a long time and cause severe losses [30], [31]. The discovery of rework requirements can induce firefighting in organizations where an unplanned allocation of resources might be necessary to address the rework requirements identified in product development. This phenomenon is common in product development organizations, especially if they are involved in multiple product development [32].

Project flexibility also increases the chances of reworks, especially in the later stages [33]. Similarly, frequent changes made in the specification of the product are identified as a persistent issue in development projects that can lead to significant reworks [34]–[37]. Studies have also reported that reworks due to specification changes can cause performance issues, conflicts, and dissatisfaction [34], [36]. Change requests from customers can cause multiple reworks and significantly delay product development, especially if the change request comes in during the later stages of product development [38]. It has been recommended that the customer be involved in product development only if the organization can deal with the associated challenges [39].

The type of product developed by startups needs to be considered when selecting apt methods for minimizing reworks. The organization's prioritization of quality will differ for startups focusing more on hardware or software development [40]. Reworks scenario will be more complex for startups that develop products that have hardware and software components; the simultaneous hardware and software development and their interaction and integration can increase the complexity of the challenges faced by these startups [40], [41].

Recently, there has been a rise in research on IoT product development [42]–[45]. The issue of reworks in IoT product development in startup companies is usually not given the necessary attention by the research community. The significance of reworks in IoT product development and its impacts on a startup company is substantial; thus, it warrants focused studies on the rework phenomenon, especially its causes in startups doing IoT product development. This study will address this issue by furthering the knowledge about the causes of reworks focusing on IoT product development in startup companies by conducting a Delphi study among people working as part of product development teams in startup companies.

III. Methodology

The Delphi method was developed as an approach to arrive at a commonly shared opinion from a group of experts [46], [47]. It is considered a method that encourages individual thinking and supports the development of a shared opinion [47]. The Delphi method has certain advantages over other group-based decision-making approaches. The anonymity of participants in the Delphi method and the avoidance of direct communication between the participants ensure that the opinion of dominant members in the panel subverting other members' opinions will not happen [47]. The anonymity encourages all panel members' active and undaunted participation in the study.

The Delphi methods usually have multiple iterations where controlled information from the preceding stage is provided to all panel members. The panel members can use the information from previous stages of the Delphi to improve upon their ideas and opinions from the previous stage [46], [47], [48]. The number of experts in the panel for a Delphi study can range from single-digit numbers to hundreds, and it usually depends on the study's context and objectives [49]–[52]. Even though having a sizeable number of members in the Delphi panel is believed to improve the quality of the study, time and resource availability also need to be accounted for when fixing the number of participants in a Delphi panel [48], [53].

Usually, non-probability sampling methods are used to select panel members for Delphi studies [48], [49]. There are no pre-defined rules for panelists selection for Delphi studies, so it is usually based on specific criteria that the researchers fix [47]. The criteria are developed based on the study's objectives and context. First, a list of experts who qualify for specific criteria created by the researcher and are willing to participate in the study is created. It is advisable to include panelists from different expertise, therefore having different perspectives to improve the study's relevance. Panel members for the study are selected equitably from the list of experts [53]. The number of iterations or Delphi sessions varies from study to study; a large number of iterations is usually avoided as it can cause exhaustion and thus lead to attrition [46], [54]–[56].

The first stage of a Delphi study is usually conducted by presenting open-ended questions to the panel members, for listing their opinion or perspective about the topic being studied/problem being addressed. The responses from the first iteration are pooled together and sent to the panel members for the second iteration. In the second iteration of the Delphi study, Likert-like scales are used for the panel members to rate each item in the list sent to them based on their importance to the topic being studied/ problem being addressed [46], [57]. From the third iteration onwards, along with the list of opinions/concepts, the mean or median score they received in the previous iteration and the standard deviation are shared with the panel members for rating again. Sometimes when the rating provided by an expert falls outside a specific limit of the standard deviation score, the expert may be requested to justify the score they have assigned to the particular opinion/concept. This procedure is repeated till the final iteration of the study, and only the elements that fall within a specific fixed range in the score are taken forward to the final list [46], [49], [58].

IV. The Delphi Process

A Delphi study was conducted to identify the factors that cause reworks in IoT product development projects in startups; Figure 1 depicts the Delphi process followed in this study. The following were the criteria used in selecting members for the expert panel.

1. Should have at least five years of experience in IoT product development in a startup company.
2. Should still be working in product development teams in startups in some capacity.

Various startups involved in IoT product development were contacted directly to identify experts who fulfilled the criteria fixed for the expert panel member in this study and to solicit their cooperation. The expert panel was formed with eleven members who satisfied the criteria and were willing to participate in this study. The average age of the participants was 30.91; eight were male, and three were female. The average experience of the panel members in IoT product development was 8.18 years, and their average experience in product development in startup companies was 5.73 years. Among the expert panel, 82.82 percent of the members were leading the projects they were currently being involved in. Five experts were from software development backgrounds, four were from hardware development backgrounds, and two were from user interface/user experience (UI/UX) development backgrounds. The members of the expert panel remained anonymous to each other during the study.

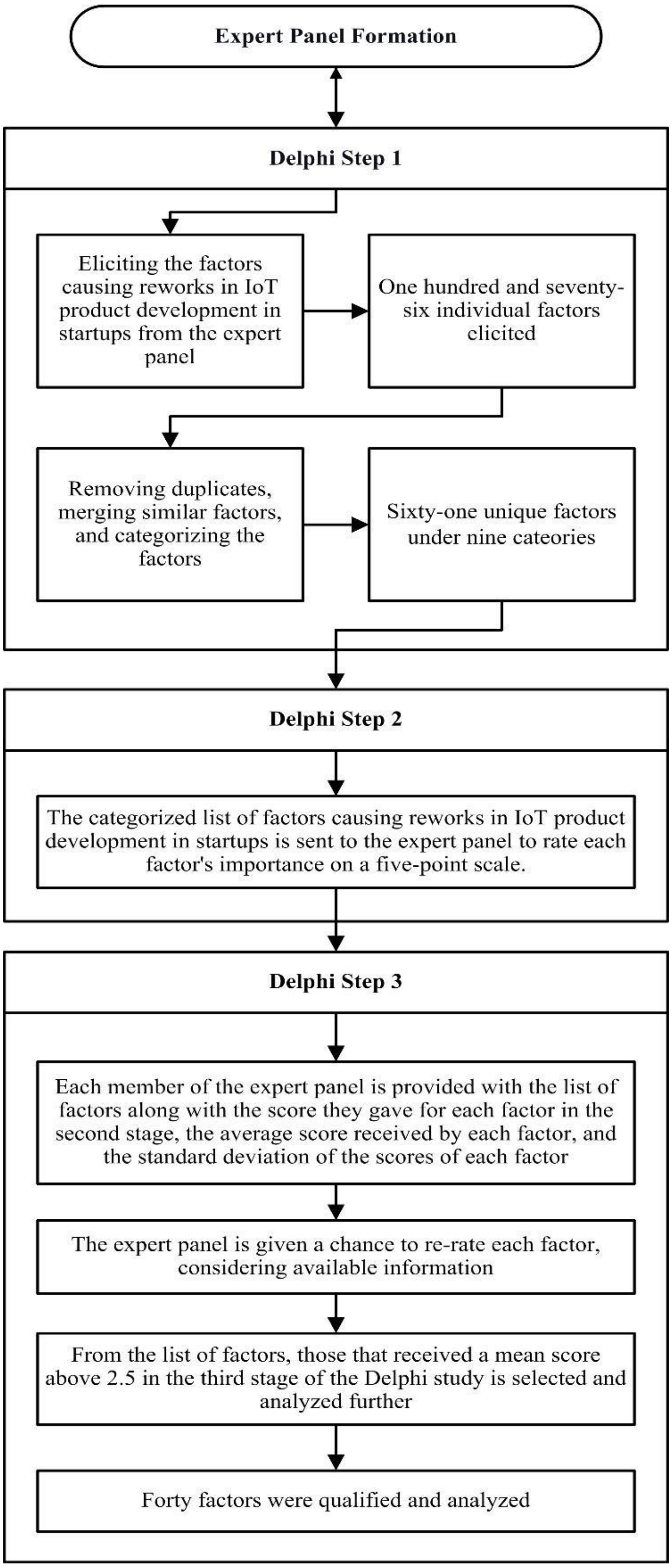

Figure 1. The Delphi Process followed in this study to identify the factors that cause reworks in IoT product development projects in startups

In the first stage of the Delphi study, the expert panel was asked to individually list the factors they perceive as the causes of reworks in IoT product development in startup companies in single sentences. The total number of factors listed by the expert panel was one hundred and seventy-six. The research team combined the responses from the expert panel, the responses with similar meanings were merged, and a single list of factors that cause reworks in IoT product development in startup companies was created. After the first stage of the Delphi study, the list of factors that influence reworks in startup product development contained sixty-one unique factors. Based on the nature of the factors, the researchers categorized the factors under nine heads, namely: (1) Factors related to requirement elicitation, (2) Factors related to changes, (3) Factors related to information issues, (4) Factors related to planning, estimating, and scheduling, (5) Factors related to inadvertent errors and issues, (6) Factors related to communication and coordination, (7) Factors related to quality management, (8) Skill and training issues, and (9) Other relevant factors.

For the second stage of the Delphi study, the categorized list of unique factors was sent back to the expert panel. The panel members were asked to rate the importance of each factor listed in a five-point scale. In the rating scale, a score of one implies that the respondent thinks the corresponding factor is not an important cause of reworks in startup product development projects. A score of two means that the respondent thinks the corresponding factor is somewhat important, a score of three means it is important, a score of four means it is very important, and a score of five means it is an extremely important cause of reworks in startup product development projects.

For the third stage of the Delphi Study, the list was again sent to the panel members to rate each factor's importance on the five-point scale. In this stage, the score they gave for each factor in the second stage, along with the average score received by each factor and the standard deviation of the scores of each factor, were sent to the panel members. The panel members were given the chance to re-rate each factor, considering the mean score each factor received and the standard deviation. The response received from the panel members will be discussed in section V.

V. Analysis And Interpretation

The categorized list of sixty-one factors that can cause reworks in IoT product development in startup companies, along with the mean score each factor received in the second and third stage of the Delphi study and the standard deviation score, is depicted in Table 1. It was seen that the standard deviation of the score each factor received in the third stage of the Delphi study was smaller than in the second stage, indicating the gradual convergence of opinion among the experts in the panel.

From the listed factors, only factors that received a mean score above 2.5 in the third stage of the Delphi study were considered for further analysis, and forty out of sixty-one factors qualified. The list of qualified factors arranged in the descending order of the mean score they received in the third stage of the Delphi study is listed in Table 2. Three of the listed factors received a mean score of 4.00 and topped the list. They are (1) The team/organization not conducting proper requirement elicitation, (2) Design and scope changes initiated by the customers, and (3) Lack of proper planning. Four factors received mean scores between 3.50 and 3.99; they were (1) Improper design, (2) Inadequate testing of the product being developed, (3) Not finding the root cause of errors/mistakes and not learning from previous mistakes, and (4) Inaccurate and non-reliable information. Fourteen factors received a score between 3.00 and 3.49, and nineteen factors received a score between 2.50 and 2.99.

Lack of proper planning and the team/organization not conducting proper requirement elicitation were among the top causes of reworks, and the team/organization not having any procedure for updating requirements when environmental conditions or customer requirements change was also found to be a related and important cause of reworks. This finding is consistent with other studies on reworks in projects in other sectors that found that lack of planning is a major cause of reworks in projects [23], [24]. This means that proper planning, requirement elicitation, and procedures for updating those requirements when needed are important for the smooth running of the project and also in reducing reworks, and it is relevant not just in IoT product development in startups but also in projects in other sectors. Lack of clarity regarding the project’s objectives was also an important source of rework identified in this study, similar to the findings of Hwang et al. [22] and Johns [28].

Design and scope changes initiated by the customers were another top source of reworks in IoT product development in startups. A similar observation has been made in other studies where customer-initiated changes in the scope and design of the projects were found to be a significant cause of reworks [22], [38]. Design and scope changes initiated by team members were also important causes of reworks in IoT product development, even though its score was less than that of customer-initiated changes. Improper design was also reported to be an important cause of rework; it is especially significant in the case of IoT product development as the product being developed will have diverse components that need to work harmoniously, and erroneous designs can impede the development of the product with intended functionalities. Inadequate testing of the product being developed was also reported to be a major issue causing reworks along with not detecting errors early on, not finding the root cause of errors/mistakes, and not learning from previous mistakes. This finding is similar to that of studies conducted on projects in other domains that found that not conducting proper testing leads to errors going unnoticed and, when they are detected later on, causes significant reworks which could have been minimized if they were identified in earlier stages of the project [29], [32], [59], [60].

Studies on causes of reworks in projects across sectors have identified the importance of proper communication and coordination among all stakeholders and team members and how the lack of proper communication and coordination can lead to various problems, including reworks [28], [61]–[64]. This study has found that factors related to communication and coordination, such as (1) Lack of proper coordination and communication among team members, (2) Lack of proper coordination and communication with customers/clients, (3) Misinterpretation of instructions and requirements, (4) Not taking feedback from all stakeholders, and (5) Delays in communicating necessary information and requirements are important causes of reworks in IoT product development projects in startup companies.

Table 1.

The categorized list of sixty-one factors that can cause reworks in IoT product development in startup companies, along with the mean score received in the second and third stages of the Delphi study and the standard deviation score

Sl. No.	Factors	Delphi Stage 2		Delphi Stage 3	
		Mean Score	Standard Deviation	Mean Score	Standard Deviation
	Factors related to Requirement Elicitation				
1	The team/organization not conducting proper requirement elicitation.	3.82	0.95	4.00	0.87
2	The team/organization not having any procedure for updating requirements when environmental conditions or customer requirements change.	2.91	0.91	3.00	0.87
	Factors related to Changes				
3	Design and scope changes initiated by the customers.	3.82	1.36	4.00	1.06
4	Design and scope changes initiated by team members.	2.73	0.76	2.82	0.58
5	Developments affecting cyber security and data privacy necessitating reworks to improve the safety.	2.91	1.18	2.91	0.91
6	Advancements in IoT sensors and actuators technology necessitating reworks to adapt to the latest technology.	2.73	1.07	2.73	0.98
7	Advancements in IoT communication platforms and technologies necessitating reworks to adapt to the latest advancements.	2.36	0.99	2.55	0.79
8	Advancements in IoT related IT technologies necessitating reworks to adapt to the latest advancements.	2.45	1.18	2.82	0.85
9	Advancements in IoT data storage and management system technologies necessitating reworks to adapt to the latest advancements.	2.27	0.98	2.27	0.88
	Factors related to Information Issues				
10	Inaccurate and non-reliable information.	3.18	1.55	3.55	1.25
11	Not getting adequate information at the right time.	2.91	1.10	3.27	1.07
	Factors related to Planning, Estimating and Scheduling				
12	Lack of proper planning.	3.64	1.32	4.00	0.87
13	Not conducting brainstorming.	2.55	0.90	2.36	0.65
14	Lack of clarity regarding the project’s objectives.	3.00	1.06	3.36	0.78

15	Project objectives not synchronized with corporate objectives.	1.91	1.01	1.73	0.76
16	Lack of proper project scheduling.	2.09	1.01	2.18	0.95
17	Lack of proper resource allocation.	2.27	1.15	2.45	1.09
18	Improper work allocation/assigning the wrong task to the wrong person.	2.45	1.39	2.91	1.01
19	Lack of proper estimating of the project's cost.	2.36	1.39	2.18	1.13
20	Not setting proper milestones for the project.	1.82	0.85	1.91	0.80
	Factors related to Inadvertent Errors and Issues				
21	Errors/mistakes in instructions.	2.45	1.25	3.09	0.91
22	Errors/mistakes in documentation.	2.45	1.39	2.55	1.33
23	Errors and mistakes in outsourced works.	3.18	1.49	3.45	1.18
24	Improper design.	3.45	1.33	3.64	0.90
25	Accidental and other unpredictable problems.	2.73	0.98	2.91	0.80
26	Errors and mistakes made by team members.	2.73	1.07	2.55	0.79
27	Equipment failures.	2.18	1.29	1.91	0.80
28	Software failure.	3.00	1.22	2.45	0.90
	Factors related to Communication and Coordination				
29	Lack of proper coordination and communication among team members.	2.73	0.76	3.18	0.58
30	Lack of proper coordination and communication with suppliers.	2.18	1.04	2.45	0.67
31	Lack of proper coordination and communication with customers/clients.	2.73	0.98	3.00	0.75
32	Lack of consensus regarding the project's objectives.	2.00	1.15	2.09	0.91
33	Lack of consensus regarding the project's scope.	2.36	1.25	2.36	1.08
34	Lack of consensus regarding the project's schedule.	2.36	1.39	2.36	1.25
35	Misinterpretation of instructions and requirements.	2.73	0.76	2.91	0.68
36	Not taking feedback from all stakeholders.	2.64	0.90	2.82	0.73
37	Delays in communicating necessary information and requirements.	2.55	0.90	2.82	0.73
	Factors related to Quality management				
38	Lack of adequate quality management practices and procedures.	3.09	1.01	3.36	0.78
39	Not detecting errors early on.	3.18	1.21	3.18	0.95
40	The project team do not address the errors/mistakes early on.	2.91	0.91	2.91	0.80
41	Non-compliance to instructions.	2.73	0.98	2.91	0.68
42	Use of inadequate-quality components.	3.09	1.26	3.27	0.98
43	Use of inadequate-quality tools and equipment.	2.82	1.36	3.09	1.01
44	Inadequate testing of the product being developed.	3.45	1.00	3.64	0.90

45	Not maintaining tools and equipment.	2.45	1.25	3.00	0.87
46	Accepting reworks as normal and not finding their sources.	3.27	1.44	3.36	1.32
47	Not finding the root cause of errors/mistakes and not learning from previous mistakes.	3.55	1.00	3.64	0.78
	Skill and Training Issues				
48	Lack of knowledge of the team members about the tasks.	2.18	0.85	2.18	0.58
49	Lack of skill of team members.	2.45	1.00	2.55	0.90
50	Lack of management skills of the team leader.	2.91	1.47	2.82	1.36
51	Not providing adequate training to the project team.	3.00	1.15	2.82	0.85
	Other relevant factors				
52	Team member attrition during the project.	2.45	1.00	2.45	0.79
53	Team member burnout due to overworking.	2.36	0.90	2.45	0.79
54	The project team hiding/not reporting errors/mistakes.	2.64	1.17	2.36	0.99
55	Non-compatibility of system components.	2.36	1.39	2.45	1.18
56	Lack of proper documentation.	3.09	1.47	2.55	1.25
57	Errors caused due to the complexity of the project.	3.09	1.33	3.09	1.10
58	Changes necessary due to developments in environment like new government policy, wars, pandemic etc.	1.73	0.88	1.55	0.67
59	Changes that are required due to issues in the technical feasibility of the product being developed.	2.45	1.18	2.27	0.88
60	Changes necessitated by budgetary constraints.	2.64	0.90	2.73	0.88
61	Competing Priorities of different stakeholders.	2.18	0.95	2.27	0.76

Table 2. Significant factors that can cause reworks in the IoT product development projects in startups

Sl. No.	Factors	Mean Score
1	The team/organization not conducting proper requirement elicitation.	4.00
2	Design and scope changes initiated by the customers.	4.00
3	Lack of proper planning.	4.00
4	Improper design.	3.64
5	Inadequate testing of the product being developed.	3.64
6	Not finding the root cause of errors/mistakes and not learning from previous mistakes.	3.64
7	Inaccurate and non-reliable information.	3.55
8	Errors and mistakes in outsourced works.	3.45
9	Lack of clarity regarding the project’s objectives.	3.36
10	Lack of adequate quality management practices and procedures.	3.36
11	Accepting reworks as normal and not finding their sources.	3.36
12	Not getting adequate information at the right time.	3.27
13	Use of inadequate-quality components.	3.27
14	Lack of proper coordination and communication among team members.	3.18
15	Not detecting errors early on.	3.18

Sl. No.	Factors	Mean Score
16	Errors/mistakes in instructions.	3.09
17	Use of inadequate-quality tools and equipment.	3.09
18	Errors caused due to the complexity of the project.	3.09
19	The team/organization not having any procedure for updating requirements when environmental conditions or customer requirements change.	3.00
20	Lack of proper coordination and communication with customers/clients.	3.00
21	Not maintaining tools and equipment.	3.00
22	Developments affecting cyber security and data privacy necessitating reworks to improve the safety.	2.91
23	Improper work allocation/assigning the wrong task to the wrong person.	2.91
24	Accidental and other unpredictable problems.	2.91
25	Misinterpretation of instructions and requirements.	2.91
26	The project team do not address the errors/mistakes early on.	2.91
27	Non-compliance to instructions.	2.91
28	Design and scope changes initiated by team members.	2.82
29	Advancements in IoT related IT technologies necessitating reworks to adapt to the latest advancements.	2.82
30	Not taking feedback from all stakeholders.	2.82
31	Delays in communicating necessary information and requirements.	2.82
32	Lack of management skills of the team leader.	2.82
33	Not providing adequate training to the project team.	2.82
34	Advancements in IoT sensors and actuators technology necessitating reworks to adapt to the latest technology.	2.73
35	Changes necessitated by budgetary constraints.	2.73
36	Advancements in IoT communication platforms and technologies necessitating reworks to adapt to the latest advancements.	2.55
37	Errors/mistakes in documentation.	2.55
38	Errors and mistakes made by team members.	2.55
39	Lack of skill of team members.	2.55
40	Lack of proper documentation.	2.55

Information-related issues like Inaccurate and non-reliable information and not getting adequate information at the right time were also found to cause reworks in IoT product development. Previous studies in other sectors have already reported that Proper management of information in projects is extremely important in complex projects and more so when multiple parties are involved in the development process [61], [65].

Some of the causes of reworks found in this study that are more relevant to IoT product development than other projects were: (1) Developments affecting cyber security and data privacy necessitating reworks to improve the safety, (2) Advancements in IoT related IT technologies necessitating reworks to adapt to the latest advancements, (3) Advancements in IoT sensors and actuators technology necessitating reworks to adapt to the latest technology, and (4) Advancements in IoT communication platforms and technologies necessitating reworks to adapt to the latest advancements. The rapid advancement in IoT-related technologies, and legal and ethical issues pertaining to data privacy, and the rise in cyber security issues heighten the importance of these factors.

Quality-related factors like (1) Lack of adequate quality management practices and procedures, (2) Use of inadequate-quality components, (3) Use of inadequate-quality tools and equipment, (4) Not maintaining tools and equipment, and (5) Non-compliance to instructions were found to cause reworks. Studies on projects in other sectors have also come up with similar findings related to reworks caused by quality-related issues [61], [66], [67]. Accepting reworks as normal and not finding their sources was also identified as an important source of rework. Similar observation has been made in projects in other sectors signifying the need to avoid this mindset among the project team [30], [68].

Lack of management skills of the team leader and the lack of skill of team members were identified as factors that can cause rework. Not providing adequate training to the project team, and improper work allocation/assigning the wrong task to the wrong person are also rework-causing factors. These findings match previous studies on other sectors that have found inadequate training and engaging participants with low skills can lead to reworks [69], [70].

Factors related to inadvertent errors and issues and other relevant factors that were identified to cause reworks in IoT product development in startups were (1) Errors and mistakes in outsourced works, (2) Errors/mistakes in instructions, (3) Accidental and other unpredictable problems, (4) Errors/mistakes in documentation, (5) Errors and mistakes made by team members, (6) Errors caused due to the complexity of the project, (7) Changes necessitated by budgetary constraints, and (8) Lack of proper documentation. Among them, Errors and mistakes in outsourced works, Errors/mistakes in instructions, and Errors caused due to the complexity of the project had a mean score above 3.00, indicating that they are important causes of reworks.

By calculating the mean score received for each head in the categorization considering only the factors that qualified for further analysis in the third stage of the Delphi study, it was found that factors related to requirement elicitation were the most important factors that can cause reworks in IoT product development projects. The factors related to planning, estimating, and scheduling were second in importance, followed closely by factors related to information issues. Factors related to quality management and factors related to inadvertent errors and issues were fourth and fifth in importance, respectively. The mean score received for each head of the categorization is depicted in Table 3.

Table 3. Mean score received for each head of the categorization of factors that cause reworks in IoT startup product development projects.

Sl. No.	Categorization of factors	Mean Score
1	Factors related to Requirement Elicitation	3.50
2	Factors related to Planning, Estimating and Scheduling	3.42
3	Factors related to Information Issues	3.41
4	Factors related to Quality management	3.24
5	Factors related to Inadvertent Errors and Issues	3.03
6	Factors related to Changes	2.97
7	Factors related to Communication and Coordination	2.95
8	Other relevant factors	2.79
9	Skill and Training Issues	2.73

This work presents an attempt to discover the factors that cause reworks in IoT product development in startup companies. In this study, the Delphi method was employed to gather information regarding rework-causing factors from experts working in IoT product development in startups. This study addressed an important gap in knowledge regarding the causes of reworks in IoT product development in startup companies. The information from this study can help IoT product development startup teams reduce reworks by avoiding, minimizing, or mitigating the major factors that can cause reworks.

VI. Conclusions And Directions For Future Research

In the Industry 4.0 age, IoT products are finding applications in diverse sectors, and many startups are engaged in developing IoT products. IoT product development involves the simultaneous development and integration of various components of the IoT product or system, including the sensors, hardware, software, UI/UX, communication channels, and data storage systems, making the product development complex and requiring the participation of team members with diverse experience. The chances for reworks are higher in IoT product development than in most other fields due to the fast-paced technological changes, the complexity of products, and the involvement of people from different fields. Reworks can cause severe problems for startups working on IoT product development, and it is vital to identify the major sources of reworks in IoT product development. This research identified the factors that cause reworks in IoT product development projects in startups by conducting a Delphi study among eleven experienced experts working in IoT product development in startups.

A list of sixty-one factors that cause reworks in IoT product development were identified in the first stage, and they were classified under nine heads by the researchers. In the subsequent steps of the Delphi study, forty out of the sixty-one factors were found to be significant factors that can cause reworks in the IoT product development projects. The most important factors among them were (1) The team/organization not conducting proper requirement elicitation, (2) Design and scope changes initiated by the customers, and (3) Lack of proper planning. Some factors identified in this study as causes of reworks were consistent with the findings of previous studies in other fields, and some other factors were more relevant in the case of IoT product development.

There is considerable scope for future research in this area. For future works, field studies and case studies in IoT product development projects in startups can give a more in-depth understanding of the dynamics of reworks in those projects. This research can also be replicated in IoT product development in non-startup companies and study how the factors influencing reworks and their significance differ in startups and non-startups.

References

[1] A. Rejeb *et al.*, "The Internet of Things (IoT) in healthcare: Taking stock and moving forward," *Internet of Things*, vol. 22, 2023, doi: https://doi.org/10.1016/j.iot.2023.100721.

[2] S. Zeadally and O. Bello, "Harnessing the power of Internet of Things based connectivity to improve healthcare," *Internet of Things*, vol. 14, 2021, doi: https://doi.org/10.1016/j.iot.2019.100074.

[3] A. Rejeb, K. Rejeb, A. Abdollahi, F. Al-Turjman, and H. Treiblmaier, "The Interplay between the Internet of Things and agriculture: A bibliometric analysis and research agenda," *Internet of Things*, vol. 19, 2022, doi: https://doi.org/10.1016/j.iot.2022.100580.

[4] H. Benyezza, M. Bouhedda, R. Kara, and S. Rebouh, "Smart platform based on IoT and WSN for monitoring and control of a greenhouse in the context of precision agriculture," *Internet of Things*, vol. 23, 2023, doi: https://doi.org/10.1016/j.iot.2023.100830.

[5] J. Henry, S. Tang, S. Mukhopadhyay, and H. M. Yap, "A randomised control trial for measuring student engagement through the Internet of Things and serious games," *Internet of Things*, vol. 13, 2021, doi: https://doi.org/10.1016/j.iot.2020.100332.

[6] H. Pourrahmani *et al.*, "The applications of Internet of Things in the automotive industry: A review of the batteries, fuel cells, and engines," *Internet of Things*, vol. 19, 2022, doi: https://doi.org/10.1016/j.iot.2022.100579.

[7] S. Caballé, F. Xhafa, and L. Barolli, "Using mobile devices to support online collaborative learning," *Mobile Information Systems*, vol. 6, no. 1, pp. 27–47, 2010, doi: 10.3233/MIS-2010-0091.

[8] P. E. D. Love and D. J. Edwards, "Forensic project management: The underlying causes of rework in construction projects," *Civil Engineering and Environmental Systems*, vol. 21, no. 3, pp. 207–228, Sep. 2004, doi: 10.1080/10286600412331295955.

[9] P. E. D. Love, F. Ackermann, B. Carey, J. Morrison, M. Ward, and A. Park, "Praxis of rework mitigation in construction," *Journal of Management in Engineering*, vol. 32, no. 5, Sep. 2016, doi: 10.1061/(ASCE)ME.1943-5479.0000442.

[10] J. M. Lyneis and D. N. Ford, "System dynamics applied to project management: A survey, assessment, and directions for future research," *System Dynamics Review*, vol. 23, no. 2–3, pp. 157–189, Jun. 2007, doi: 10.1002/SDR.377.

[11] K. G. Cooper, "Naval Ship Production: A Claim Settled and a Framework Built," *Interfaces (Providence)*, vol. 10, no. 6, pp. 20–36, Dec. 1980, doi: 10.1287/INTE.10.6.20.

[12] K. G. Cooper, "The $2,000 hour: How managers influence project performance through the rework cycle," *Project Management Journal*, vol. 25, no. 1, 1994, Accessed: Nov. 27, 2023.

[13] R. A. Rivas, J. D. Borcherding, V. González, and L. F. Alarcón, "Analysis of Factors Influencing Productivity Using Craftsmen Questionnaires: Case Study in a Chilean Construction Company," *Journal of construction engineering and management*, vol. 137, no. 4, pp. 312–320, Apr. 2011, doi: 10.1061/(ASCE)CO.1943-7862.0000274.

[14] J. B. H. Yap, M. Skitmore, J. Gray, and K. Shavarebi, "Systemic view to understanding design change causation and exploitation of communications and knowledge," *Project Management Journal*, vol. 50, no. 3, pp. 288–305, Jun. 2019, doi: 10.1177/8756972819829641.

[15] P. E. D. Love, F. Ackermann, J. Smith, Z. Irani, and D. J. Edwards, "Making sense of rework causation in offshore hydrocarbon projects," *Project Management Journal*, vol. 47, no. 4, pp. 16–28, Aug. 2016, doi: 10.1177/875697281604700403.

[16] P. Love, J. Smith, F. Ackermann, and Z. Irani, "The praxis of stupidity: an explanation to understand the barriers mitigating rework in construction," *Production Planning & Control*, vol. 29, no. 13, pp. 1112–1125, Oct. 2018, doi: 10.1080/09537287.2018.1518551.

[17] P. E. D. Love and P. Teo, "Statistical Analysis of Injury and Nonconformance Frequencies in Construction: Negative Binomial Regression Model," *Journal of construction engineering and management*, vol. 143, no. 8, Aug. 2017, doi: 10.1061/(ASCE)CO.1943-7862.0001326.

[18] V. Gupta, J. M. Fernandez-Crehuet, T. Hanne, and R. Telesko, "Fostering product innovations in software startups through freelancer supported requirement engineering," *Results in Engineering*, vol. 8, 2020, doi: https://doi.org/10.1016/j.rineng.2020.100175.

[19] V. Berg, J. Birkeland, A. Nguyen-Duc, I. O. Pappas, and L. Jaccheri, "Achieving agility and quality in product development-an empirical study of hardware startups," *Journal of Systems and Software*, vol. 167, 2020, doi: https://doi.org/10.1016/j.jss.2020.110599.

[20] C. Laporte and R. O'Connor, "Software process improvement standards and guides for very small organization: An overview of eight implementations," *CrossTalk, The Journal of Defense Software Engineering*, vol. 30, no. 3, pp. 23–27, 2017, Accessed: Nov. 27, 2023.

[21] H. Doloi, "Cost Overruns and Failure in Project Management: Understanding the Roles of Key Stakeholders in Construction Projects," *Journal of construction engineering and management*, vol. 139, no. 3, pp. 267–279, Mar. 2013, doi: 10.1061/(ASCE)CO.1943-7862.0000621.

[22] B. Hwang, X. Zhao, K. G.-I. J. of P. Management, and undefined 2014, "Investigating the client-related rework in building projects: The case of Singapore," *International Journal of Project Management*, vol. 32, no. 4, pp. 698–708, 2014, doi: https://doi.org/10.1016/j.ijproman.2013.08.009.

[23] P. E. D. Love, D. J. Edwards, H. Watson, and P. Davis, "Rework in Civil Infrastructure Projects: Determination of Cost Predictors," *Journal of construction engineering and management*, vol. 136, no. 3, pp. 275–282, Mar. 2010, doi: 10.1061/(ASCE)CO.1943-7862.0000136.

[24] P. E. D. Love and J. Smith, "Benchmarking, Benchaction, and Benchlearning: Rework Mitigation in Projects," *Journal of Management in Engineering*, vol. 19, no. 4, pp. 147–159, Oct. 2003, doi: 10.1061/(ASCE)0742-597X(2003)19:4(147).

[25] H. Rahmandad and D. M. Weiss, "Dynamics of concurrent software development," *System Dynamics Review*, vol. 25, no. 3, pp. 224–249, Jul. 2009, doi: 10.1002/SDR.425.

[26] H. Akkermans and K. E. Van Oorschot, "Pilot Error? Managerial Decision Biases as Explanation for Disruptions in Aircraft Development," *Project Management Journal*, vol. 47, no. 2, pp. 79–102, Apr. 2016, doi: 10.1002/PMJ.21585.

[27] D. N. Ford and J. D. Sterman, "Overcoming the 90% syndrome: Iteration management in concurrent development projects," *Concurrent Engineering*, vol. 11, no. 3, p. 177, Sep. 2003, doi: 10.1177/106329303038031.

[28] T. G. Johns, "Managing the behavior of people working in teams. Applying the project-management method," *International Journal of Project Management*, vol. 13, no. 1, pp. 33–38, 1995, doi: https://doi.org/10.1016/0263-7863(95)95701-E.

[29] Y. Li and T. R. B. Taylor, "Modeling the Impact of Design Rework on Transportation Infrastructure Construction Project Performance," *Journal of construction engineering and management*, vol. 140, no. 9, Sep. 2014, doi: 10.1061/(ASCE)CO.1943-7862.0000878.

[30] P. E. D. Love, J. Smith, F. Ackermann, and Z. Irani, "Making sense of rework and its unintended consequence in projects: The emergence of uncomfortable knowledge," *International Journal of Project Management*, vol. 37, no. 3, pp. 501–516, 2019, doi: https://doi.org/10.1016/j.ijproman.2019.02.004.

[31] D. N. Ford and J. D. Sterman, "The Liar's Club: concealing rework in concurrent development," *Concurrent Engineering*, vol. 11, no. 3, pp. 211–219, Sep. 2003, doi: 10.1177/106329303038028.

[32] N. P. Repenning, "Understanding fire fighting in new product development," *Journal of Product Innovation Management*, vol. 18, no. 5, pp. 285–300, Sep. 2001, doi: 10.1111/1540-5885.1850285.

[33] N. O. E. Olsson, "Management of flexibility in projects," *International Journal of Project Management*, vol. 24, pp. 66–74, 2006, doi: https://doi.org/10.1016/j.ijproman.2005.06.010.

[34] P. K. Lam and K. S. Chin, "Identifying and prioritizing critical success factors for conflict management in collaborative new product development," *Industrial Marketing Management*, vol. 34, no. 8, pp. 761–772, Nov. 2005, doi: 10.1016/J.INDMARMAN.2004.12.006.

[35] R. Nellore and K. Söderquist, "Strategic outsourcing through specifications," *Omega (Westport)*, vol. 28, no. 5, pp. 525–540, 2000, doi: https://doi.org/10.1016/S0305-0483(99)00078-X.

[36] C. Karlsson, R. Nellore, and K. Söderquist, "Black Box Engineering: Redefining the Role of Product Specifications," *Journal of Product Innovation Management*, vol. 15, no. 6, pp. 534–549, 1998, doi: 10.1111/1540-5885.1560534.

[37] D. G. Smith and R. G. Rhodes, "Specification Formulation—an Approach that Works," *Journal of Engineering Design*, vol. 3, no. 4, pp. 275–289, Jan. 1992, doi: 10.1080/09544829208914762.

[38] X. Zhang, Y. Tan, and Z. Yang, "Analysis of impact of requirement change on product development progress based on system dynamics," *IEEE Access*, vol. 9, pp. 445–457, 2020, Accessed: Nov. 27, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9305213/

[39] A. S. Cui and F. Wu, "The Impact of Customer Involvement on New Product Development: Contingent and Substitutive Effects," *Journal of Product Innovation Management*, vol. 34, no. 1, pp. 60–80, Jan. 2017, doi: 10.1111/JPIM.12326.

[40] J. Ronkainen and P. Abrahamsson, "Software development under stringent hardware constraints: Do agile methods have a chance?," *XP 2003: Extreme Programming and Agile Processes in Software Engineering*, pp. 73–79, 2003, doi: 10.1007/3-540-44870-5_10.

[41] J. Ronkainen, J. Taramaa, and A. Savuoja, "Characteristics of process improvement of hardware-related SW," *PROFES 2002: Product Focused Software Process Improvement*, pp. 247–257, 2002, doi: 10.1007/3-540-36209-6_22.

[42] L. C. B. C. Ferreira, P. R. Chaves, R. M. Assumpção, O. C. Branquinho, F. Fabiano, and P. Cardieri, "The three-phase methodology for iot project development," *Internet of Things*, vol. 20, 2022, doi: https://doi.org/10.1016/j.iot.2022.100624.

[43] A. Collaguazo, M. Villavicencio, and A. Abran, "An activity-based approach for the early identification and resolution of problems in the development of IoT systems in academic projects," *Internet of Things*, vol. 24, 2023, doi: https://doi.org/10.1016/j.iot.2023.100929.

[44] F. Sanchez-Sutil and A. Cano-Ortega, "Development and implementation of a PQ analyser to monitoring public lighting installations with a LoRa wireless system," *Internet of Things*, vol. 22, 2023, doi: https://doi.org/10.1016/j.iot.2023.100711.

[45] A. Chakraborty, R. Das Gupta, Md. Z. Kabir, and S. Dhar, "Development of an IoT-enabled cost-effective asthma patient monitoring system: Integrating health and indoor environment data with statistical analysis and data visualization," *Internet of Things*, vol. 24, 2023, doi: https://doi.org/10.1016/j.iot.2023.100942.

[46] S. Thangaratinam and C. W. Redman, "The Delphi technique," *The Obstetrician & Gynaecologist*, vol. 7, no. 2, pp. 120–125, Apr. 2005, doi: 10.1576/TOAG.7.2.120.27071.

[47] N. Dalkey and O. Helmer, "An Experimental Application of the DELPHI Method to the Use of Experts," *Manage Sci*, vol. 9, no. 3, pp. 458–467, Apr. 1963, doi: 10.1287/MNSC.9.3.458.

[48] C. Powell, "The Delphi technique: Myths and realities," *Journal of Advanced Nursing*, vol. 41, no. 4, pp. 376–382, Feb. 2003, doi: 10.1046/J.1365-2648.2003.02537.X.

[49] F. Hasson, S. Keeney, and H. Mckenna, "Research guidelines for the Delphi survey technique," *Journal of Advanced Nursing*, vol. 32, no. 4, pp. 1008–1015, 2000, doi: 10.1046/j.1365-2648.2000.t01-1-01567.x.

[50] A. Fink, J. Kosecoff, M. Chassin, and R. H. Brook, "Consensus methods: Characteristics and guidelines for use," *American Journal Public Health*, vol. 74, no. 9, pp. 979–983, 1984, doi: 10.2105/AJPH.74.9.979.

[51] N. Reid, "The Delphi technique: its contribution to the evaluation of professional practice," *Professional competence and quality assurance in the caring professions*, 1998.

[52] A. Delbecq, A. Van de Ven, and D. Gustafson, "Group techniques for program planning: A guide to nominal group and Delphi processes," 1975, Accessed: Nov. 27, 2023.

[53] M. Murphy *et al.*, "Consensus development methods, and their use in clinical guideline development," *Health Technology Assessment (Winchester, England)*, vol. 2, no. 3, 1998, Accessed: Nov. 27, 2023.

[54] B. Green, M. Jones, D. Hughes, and A. Williams, "Applying the Delphi technique in a study of GPs' information requirements," *Health & social care in the community*, vol. 7, no. 3, pp. 198–205, 1999, doi: 10.1046/j.1365-2524.1999.00176.x.

[55] B. Beech, "Studying the future: A Delphi survey of how multi-disciplinary clinical staff view the likely development of two community mental health centres over the course of the next two years," *Journal of Advanced Nursing*, vol. 25, no. 2, pp. 331–338, 1997, doi: 10.1046/J.1365-2648.1997.1997025331.X.

[56] S. Procter and M. Hunt, "Using the Delphi survey technique to develop a professional definition of nursing for analysing nursing workload," *Journal of Advanced Nursing*, vol. 19, no. 5, pp. 1003–1014, 1994, doi: 10.1111/J.1365-2648.1994.TB01180.X.

[57] S. Drumm, C. Bradley, and F. Moriarty, "'More of an art than a science'? The development, design and mechanics of the Delphi technique," *Research in Social and Administrative Pharmacy*, vol. 18, no. 1, pp. 2230–2236, 2022, doi: https://doi.org/10.1016/j.sapharm.2021.06.027.

[58] J. W. Zartha Sossa, W. Halal, and R. Hernandez Zarta, "Delphi method: analysis of rounds, stakeholder and statistical indicators," *Foresight*, vol. 21, no. 5, pp. 525–544, Sep. 2019, doi: 10.1108/FS-11-2018-0095/FULL/HTML.

[59] F. Huq, "Testing in the software development life-cycle: now or later," *International Journal of Project Management*, vol. 18, no. 4, pp. 243–250, 2000, doi: https://doi.org/10.1016/S0263-7863(99)00024-1.

[60] A. Rodrigues and J. Bowers, "The role of system dynamics in project management," *International Journal of Project Management*, vol. 14, no. 4, pp. 213–220, 1996, doi: https://doi.org/10.1016/0263-7863(95)00075-5.

[61] F. Pargar, J. Kujala, K. Aaltonen, and sampsa Ruutu, "Value creation dynamics in a project alliance," *International Journal of Project Management*, vol. 37, no. 5, pp. 716–730, 2019, doi: https://doi.org/10.1016/j.ijproman.2018.12.006.

[62] Y. Lu, L. Luo, H. Wang, Y. Le, and Q. Shi, "Measurement model of project complexity for large-scale projects from task and organization perspective," *International Journal of Project Management*, vol. 33, no. 3, pp. 610–622, 2015, doi: https://doi.org/10.1016/j.ijproman.2014.12.005.

[63] K. C. Iyer, N. B. Chaphalkar, and G. A. Joshi, "Understanding time delay disputes in construction contracts," *International Journal of Project Management*, vol. 26, no. 2, pp. 174–184, 2008, doi: https://doi.org/10.1016/j.ijproman.2007.05.002.

[64] P. Barber, C. Tomkins, and A. Graves, "Decentralised site management—a case study," *International Journal of Project Management*, vol. 17, no. 2, pp. 113–120, 1999, doi: https://doi.org/10.1016/S0263-7863(98)00014-3.

[65] D. K. H. Chua and Hossain Md. Aslam, "A simulation model to study the impact of early information on design duration and redesign," *International Journal of Project Management*, vol. 29, no. 3, pp. 246–257, 2011, doi: https://doi.org/10.1016/j.ijproman.2010.02.012.

[66] X. Meng and B. Gallagher, "The impact of incentive mechanisms on project performance," *International journal of project management*, vol. 30, no. 3, pp. 352–362, 2012, doi: https://doi.org/10.1016/j.ijproman.2011.08.006.

[67] Andi and T. Minato, "Design documents quality in the Japanese construction industry: factors influencing and impacts on construction process," *International Journal of Project Management*, vol. 21, no. 7, pp. 537–546, 2003, doi: https://doi.org/10.1016/S0263-7863(02)00083-2.

[68] P. E. D. Love, P. Teo, M. Davidson, S. Cumming, and J. Morrison, "Building absorptive capacity in an alliance: Process improvement through lessons learned," *International Journal of Project Management*, vol. 34, no. 7, pp. 1123–1137, 2016, doi: https://doi.org/10.1016/j.ijproman.2016.05.010.

[69] F. Y. Y. Ling, S. P. Low, S. Q. Wang, and H. H. Lim, "Key project management practices affecting Singaporean firms' project performance in China," *International Journal of Project Management*, vol. 27, no. 1, pp. 59–71, 2009, doi: https://doi.org/10.1016/j.ijproman.2007.10.004.

[70] P. E. D. Love, G. D. Holt, L. Y. Shen, H. Li, and Z. Irani, "Using systems dynamics to better understand change and rework in construction project management systems," *International Journal of Project Management*, vol. 20, no. 6, pp. 425–436, 2002, doi: https://doi.org/10.1016/S0263-7863(01)00039-4.