Towards a comprehensive framework for enablers and inhibitors of bad news reporting on software projects in state universities in Zimbabwe

Melody Maseko
Department of Information and Marketing Sciences,
Midlands State University
Gweru, Zimbabwe
mmaseko@staff.msu.ac.zw,
https://orcid.org/0000-0002-4402-0471

Theo Tsokota
Department of Information and Marketing Sciences,
Midlands State University
Gweru, Zimbabwe
tsokotat@staff.msu.ac.zw
https://orcid.org/0000-0002-7347-515X

***Abstract* — Software project status reporting is critical in software project management, yet team members often find it easier to report positive news than negative progress. This research investigates the 'mum effect', which refers to the reluctance to report bad news on software projects by project team members. Silence on project bad news has remained a major contributor to project failure in higher learning institutions. This study, therefore, aimed to come up with a framework for the inhibitors and enablers of bad news reporting by project team members on specific academic and administrative software projects within state universities in Zimbabwe. Naturally, it is easier to report positive news than negative progress encountered during the software project life cycle. Following a qualitative, multiple-holistic case study approach, this research employed focus group discussions and key informant interviews with project managers, team members, and system users from three state universities in Zimbabwe. The findings indicate that the main enablers of bad news reporting include open communication, a positive organisational culture, and feedback and motivation to achieve. On the other hand, the results suggest that the main inhibitors of bad news reporting include a lack of communication, fear of punishment, an unfair distribution of work, and a lack of skills. The findings of this study can help institutions understand the dynamics at play in status reporting for software projects. Results from this study contribute to the body of knowledge theoretically and, practically, to status reporting on software project development in institutions of higher learning. This reduces the chances of software project failure and escalation.**

***Keywords*— Bad news reporting, Software projects, State universities, Zimbabwe, "mum effect", Education 5.0**

I. INTRODUCTION

The fourth industrial revolution and COVID-19 accelerated the digital transformation of many institutions and organisations, including state universities in Zimbabwe [1]. This evolution is driven by Zimbabwe's Education 5.0 policy framework, which calls for a transition from traditional teaching to innovation and industrialisation in Zimbabwe's state universities. Consequently, universities are investing in sophisticated academic and administrative systems, such as eLearning systems, student management systems, human resource management systems, accounting systems, and many others deemed necessary by the institutions. The highlighted softwares though different follow the same development life cycle. Reference [2] software project status reporting involves the regular, structured, and concise documentation of a software development project's progress, health, and risks, shared with stakeholders to ensure transparency, accountability, and alignment. However, failures in software projects remain a major challenge in the Information Technology (IT) world. Software projects usually fall behind schedule and go significantly over budget [3]. Software development products are mostly intangible, thus making project monitoring somewhat difficult. State universities are investing heavily in sophisticated software projects ranging from enterprise resource planning (ERP) and learning management systems (LMS) to high-tech prototypes in innovation hubs. Correct reporting of the project's status becomes a linchpin of institutional success. While there are several factors behind software projects' potential failure, [3] have identified project team members' reluctance (mum effect) to disclose negative updates on the progress of an IT project as a significant factor that could lead to project failure. This phenomenon usually leads to 'runaway' projects that utilise resources without delivering value. It is against this background that the writers seek to understand the enablers and inhibitors of bad news reporting by project team members in software projects.

II. THE PROBLEM AND THE GAP

A. Problem statement

The reluctance of project team members to report negative issues during project development, implementation, and post-implementation of software projects can lead to project failure. Consequently, reluctance to project status reporting can lead to risks such as project failure, loss of trust in software project teams, missed opportunities, reputational risk of universities and project overruns, which may result in loss of capital investment. The main objective of this study, therefore, is to develop a data-informed framework for the enablers and inhibitors of bad news reporting in a software development environment.

B. The Gap

Software projects usually fall behind schedule and go beyond budget. Reference [4] found that Bad News Reporting (BNR) has been neglected by Information Systems researchers, leaving ample room for further research

in this area. While general project failure is documented, there is a significant information gap regarding the enablers and inhibitors of bad news. In the context of Zimbabwean universities, unfamiliarity with these factors leads to the "mum effect," creating a disconnect between developers and decision-makers. There, however, seems to be little or no literature on the drivers and inhibitors of bad news reporting in software projects in Zimbabwe, leaving a gap for research.

Therefore, the primary objective of this research is to identify:

a) The enablers for bad news reporting among software project team members.

b) The inhibitors of bad news reporting by software project team members during software project development

III. Review of Literature

Software development is critical to organisational success in light of rapid technological advancement, particularly within Zimbabwean state universities. The move towards innovation and industrialisation (Education 5.0) by Zimbabwean state universities has greatly increased the complexity of the software projects they are implementing. Despite this investment, project failures remain a major challenge, with initiatives frequently falling behind schedule and exceeding budgets. The drive towards digitisation under the 'Education 5.0' mandate increases the risk of suppressing, delaying, or distorting project information. The reluctance to report bad news in software projects is linked to the "mum effect," which can be singled out as a major contributor to software project failures [5]. The 'mum effect' describes an individual's reluctance to report the correct status of especially troubled software projects.

Grounded in whistle-blowing theory, prior research in BNR has investigated various factors that can influence one's perception of reporting the actual status of a project, the duty to report, and the will to report bad news [6]. It is highlighted that organisational, personal, and situational factors influence this phenomenon.

A. Theoretical Framework of the "Mum Fffect*"*

For almost a decade, reluctance among team members to report bad news on software projects has been a well-documented problem in the project management literature, often highlighted as a primary contributor to project failure [7]. This phenomenon is mainly driven by the human tendency to avoid being the messenger of bad news, especially when such reporting is perceived to cause personal or professional risks. This reluctance creates a critical information gap between the operational level, where project team members observe emerging failures, and the strategic level, where decision-makers possess the authority to redirect or terminate troubled software projects.

B. The Smith and Keil Model of Reluctance to BNR

Reference [5] propounded the primary theoretical framework for understanding bad news reporting. This model synthesises literature from organisational behaviour, communications, ethics, and information systems.

TABLE I. BNR RELUCTANCE MODEL [5]

Theoretical Component	Mechanism of Influence	Impact on Reporting Behavior
Risk Perception	Assess the probability of failure versus the personal cost of reporting.	High perceived personal risk (e.g., job loss, social stigma) inhibits reporting at a greater extend.
Information Asymmetry	The gap in knowledge between project subordinates and managers.	Allows technical staff to conceal the true state of "intangible" software components if the gap do exist.
Ethical Climate	The organization's norms regarding rules, law, and social responsibility.	A "caring" or "rules-based" climate promotes reporting; an "instrumental" climate focused on self-interest discourages it.
Individual Moral Norms	Internalized standards of honesty and professional duty.	Individuals with high moral obligation are more likely to report bad news.

Table 1 suggests that the decision to report is a cognitive process influenced by four primary dimensions: risk perception, personal responsibility, organisational climate, and individual moral norms. Research has focused on macro aspects such as organisational culture and information blind spots [8], risk perception, time management [2], the perceived effects of not reporting bad news, and the challenges linked to software projects [4].

IV. Methodology

Employees at Zimbabwean State Universities comprised the core population for this study. This multiple-holistic case study covered three state universities in Zimbabwe, including institutions A, B, and C. Key informant interviews with project team members were conducted for this study. A focus group discussion was conducted with five system users in a selected state university. Custom-designed interviews were conducted for each participant group, including project managers, project team members, and some system users. The study employed the interview instrument because it is particularly useful in revealing the story behind a participant's encounters. In interviewing participants, the researcher used probing to allow contributors to elaborate on their views, provide details, and offer further insights into their reluctance to report bad news. A recording machine was used to capture the interviews with participants' consent, and transcription followed before analysis. A focus group discussion was conducted with five system users to gather more information. Group interaction among system users during focus group discussions helped participants make connections to several concepts that were not captured during individual interviews.

The study population included the software project team members, project managers and system users. In this study, intensity sampling was employed.

Intensity sampling comprises information-rich scenarios that strongly manifest the phenomenon being targeted. It bunches contributors based on preselected standards applicable to a specific research question. This study targeted software project team members and users; hence, the sample structure was based on that criterion. The sample consisted of 7 software project team members, 3 project managers, and

5 system users from three state universities in Zimbabwe. The sample distribution included one project manager, two project team members, and five system users from institution A. Institution B's target sample included one project manager and two project team members. Finally, Institution C had one project manager and two software project team members. The sample for this study was drawn from three state universities in Zimbabwe, which were anonymised as Institutions A, B, and C.

The interviews and focus group discussions were transcribed and analysed qualitatively as part of the data analysis. The second step in the data analysis process involved immersion analysis. The researcher conducted a repeat, delayed and untargeted reading of all written text responses to familiarise with the narratives and allow for the discovery of unexpected patterns. The transcribed focus group discussions and interviews were subjected to manifest qualitative content analysis, an approach oriented towards summarising the informational contents of data with minimal interference [9]. Data-driven codes that emerged from recurrent words and phrases were assigned to the text. Similar codes were then grouped into thematic categories.

V. Findings and Discussions

This study sought to identify the enablers and inhibitors of bad news reporting (BNR) on software projects within Zimbabwean state universities. Drivers of bad news reporting (BNR) refer to factors that can be operationalised to intrinsically and extrinsically motivate team members to report bad news in software projects. Reference [10] alludes to team members' reluctance to report and to the selective reporting of bad news, which leads to the failure of many major software projects. The drivers of BNR can be positive or negative, and they are applied not only to the whole organisation but also to those specifically involved in software project development. The inhibitors to BNR are factors that make it difficult for project team members to report bad news on software projects before, during, and after development. The findings reveal a tension between a deeply ingrained culture of fear and a strong, aspirational desire for psychological safety and professional growth. This section, therefore, discusses the findings of this study.

A. The enablers of Bad News Reporting

Enablers of BNR are the factors that positively motivate a member to report on a project's status. The following factors have been noted from the collected data:

1) Understanding the dynamics of teamwork

Working with teams requires knowledge, skills, and aptitudes to develop effective, motivated, and engaged project team members. Responses from the project team indicate that the unfair distribution of workload assignments undermines the drive to report bad news. Five participants highlighted concerns about this issue as an environmental factor that can either promote or stifle the reporting of bad news about software projects. Reference [11] found that useful feedback, communication, and listening skills were flagged as key themes in group dynamics during software development. Participants believe that there is a strong relationship between how the team is handled and BNR: be listed in columns, not grouped by affiliation.

"I think the ability and the understanding of how a team works. For example, if someone reports, it does not necessarily mean that they are 'sell-outs' or those who are supposed to carry the burden of doing the thing. Instead, it is supposed to be shared or given to someone who is idle or does not have anything else to do." (Participant 2, Project Team Member)

Participants from all three state universities showed that reporting especially bad news on a software project will be easy if team dynamics are well handled. Reference [11] notes the great value of team-based settings in software development, underscoring the need for project leaders to understand team dynamics and manage them appropriately. Reference [12] found that their positive team dynamics fostered inclusivity in the working environment and strengthened interpersonal relationships among team members. Attaining this leads to high motivation among team members, thereby enhancing communication. Reference [13] views teamwork dynamics from a different angle, arguing that the size of the project plays a major role in its success or failure.

2) Use of Reward systems

Reference [14] state that the reward system encompasses monetary, non-monetary, and psychological compensations that an organisation offers its employees for services rendered. Various types of rewards may be included in reward schemes, namely extrinsic and intrinsic rewards. Extrinsic rewards consist of financial remuneration and working conditions provided to employees as part of their employment. On the other hand, intrinsic rewards pertain to the gratification derived from the actual job execution, such as personal fulfilment and a feeling of making a meaningful contribution to society. Reference [15] argues that one objective of rewarding employees is to align the risk preferences of managers and employees with those of the organisation. If the project members' preferences align with the organisation's, adverse effects on projects can be avoided. Project teams will have a greater sense of responsibility, leading them to report any mishaps on the projects they will be working on.

One participant said:

"I think mostly it is about the consequences that follow. Is the reporting benefiting, or maybe it is negative to me, such that if I report more or less all those tasks that I would have reported, they will come to me again?" (Participant 3, Project Team Member)

Most project team members showed that reporting bad news should not breed negative consequences but rather contribute positively to both the project team member and the project at large. Participants had different interpretations of what could be classified as a benefit. Some referred to the supervisor's reaction or response as a critical factor influencing the desire to report. Another participant said:

"What motivates me mostly is maybe if action is going to be taken, obviously you would be motivated to go on and report more, but if nothing is done, l will just be t fed up and leave it like that." (Participant 14, Project Team Member)

Some participants proposed offering extrinsic rewards quarterly or annually to motivate them to report bad news on software projects. Two project managers also highlighted the

importance of rewards for status reporting on projects as a motivator, so that bad news reporting will not be seen as a sign of failure or incompetence, but rather as a control measure towards project success. One project manager said,

"I think we need to develop a central project management tool where everyone updates project status and makes it visible to all. Rewards should be offered to those who are up to date in updating the good and bad issues on a project, especially those who highlight bugs in a system." (Participant 8, Project Manager)

Most participants highlighted the importance of rewards in fostering a culture of reporting bad news in software projects. Reference [16], however, offer a different perspective on reward systems as drivers of bad news reporting. The author argues that corporations might fall victim to false claims by team members who may force institutions into financial settlements to avoid adverse public reputations. On the contrary, reference [17] argue that team-based rewards can have an intrinsic effect, increasing the likelihood of reluctance to report bad news when other team members engage in unethical behaviour.

3) Achievement Motivation

One important driver for a person to report bad news about a project is the sense of achievement that comes with the project's success. Almost all project team members noted that everyone enjoys being a part of success; hence, keeping quiet when a project deviates from the planned course of action no longer makes sense. A project team member said,

"The more you do not expose or the more you do not give feedback on the challenges or issues that you are facing, it can lead to a project failure, of which no one would want to be associated with something that has failed and ultimately be classed as a failure." (Participant 12, Project Team Member)

Conversely, a project manager and some system users showed that most project team members do not care about the project outcomes. According to the demographic data, across all institutions chosen for this study, several young project team members are under 35. System users raised the concern that these young members are constantly changing organisations and are still exploring; hence, they will not wait to see the project through to completion. One key system user stated that these young employees are always searching for greener pastures, so they do not see any benefit in bringing success to projects.

"Frankly, some of the project members and developers are rolling stones. They just do not care much about the success or failure of the project. They are just indifferent." (Participant 6, System User)

Another participant stated the following regarding personal achievement.

"What we do not want as management in employing someone is to lose touch with the general skills set for a post. For example, if we employ you as a webmaster, we expect you to be able to code using PHP and embed a video in HTML. So, those are the key competencies for the post. We do not want a webmaster to come and say, "I cannot install Joomla," but if it is a new technology, we will recommend training, or, for a particular project, we would need to improve our skill set. ICT is about doing what you can do best. So these guys really want to shine through. Once in a while, we release an application, and users are happy about it, then you have what we call self-actualisation." (Participant 11, Project Manager)

In light of the above comments, it is quite evident that project team members need to understand the task at hand to deliver as per user expectations. If users gain confidence in the development team, smooth communication about good or bad news is more likely. For users to gain confidence in their development team, the deliverables must be achieved as per user expectations. Reference [18] state that the job is the most cited motivator, but there are no clear indicators of which aspects of the software engineer's job trigger motivation. Reference [19] highlight that opportunities for achievement motivate software engineers working on a project. If the project team is achieving the project goals, they become motivated and embrace the project as their own.

4) Effective Group Communication

Taking note of the apprehensions surrounding status reporting, especially bad news reporting, a project manager highlighted the need to always strive to encourage one another at an individual level during software development. In this respect, some project team members emphasised the importance of speaking one-on-one with the project leader at a personal level before engaging higher authorities on any issues arising during project development. This is said to bring a sense of security to project team members, and most participants admitted that this strategy really works. One project manager noted that failure by project team members to report undesirable issues in the system impacts the whole team; hence, it is effective to build personal relationships with team members at the individual level. At one institution, the project manager schedules morning meetings where each member briefs her on what she is working on, what she needs to complete, and any challenges she is facing.

"One-on-one encounters done once or twice a week, such that when regularly done, the person who always gives excuses will ask him/herself what is wrong with me, and positive change can be achieved." (Participant 5, Project Manager)

Several project team members are afraid to speak and express themselves in meetings. Nevertheless, they prefer a face-to-face dialogue. This implies that a company must establish multiple communication channels to cultivate the nature of reporting. Reference [20] shows that reporting channels and effective communication positively influence project team members' desire to report bad news on software projects. Reference [21] stressed that effective and honest communication is a foundation for sharing information.

5) Feedback

Feedback is a component of two-way communication in which a report is sent to the responsible recipient, who then replies to the sender. Participants from all three state universities highlighted feedback as a major driver for reporting on project status. The discussion clearly articulated that feedback is lacking in state universities, as management gives the development team's reports a deaf ear. To highlight the severity of this issue, three-quarters of the study's participants reported that feedback is lacking across all three institutions. One System user said,

"We also need feedback from end users and project team members. Follow-ups are necessary to ensure constant communication between project teams, end users and the executives who are finance owners." (Participant 9, System User)

Similarly, another participant stated that:

"I think there is some need for feedback. If there is constant feedback on what you are bringing up, it would help me communicate my issues or ideas. If we get feedback, we will most likely report; if we do not, people will not feel the need to report. The more feedback we get, the more we report." (Participant 15, Project Team Member)

Feedback from all project stakeholders is critical. Users should give feedback to the development team, the project manager should give feedback to developers, and vice versa, as well as feedback from management to the project manager. Reference [22] highlight the importance of feedback in software evolution. Keil & Robey (1999, 2000) coined the term "Deaf Effect," which describes how management's refusal to hear bad news from team members about software projects leads to escalation and project failure [23]. If bad news reporters do not receive feedback, they give up, leading to the "mum effect". Acting on reported news will always encourage the reporter to come back with more news, whether it is good or bad. However, [24] believe that not everyone is expected to report bad news, except those who are role-prescribed to do so. It is not perceived as credible; hence, the message is ignored.

B. Inhibitors of Bad News Reporting

The study also sought to identify the factors behind individuals' reluctance to report bad news in software projects. Findings from this study identified issues discussed below:

1) Fear of Punishment and Stigmatisation

Reference [25] describe fear as a fundamental emotion comprising four elements: subjective feeling, physiological arousal, behavioural expression, and cognitive interpretation. One project manager highlighted the fears of project team members when reporting bad news and highlighted the importance of employing strategies to help project teams report news without fear. Positive and negative measures can be put in place to encourage reporting of bad news. This participant highlighted that punishment for those project team members who fail to report bad news should be employed to show other team members the consequences of not reporting in the future. In the same vein, punishments for mistakes can deter the reporting of unpleasant news in software projects. The following quotation is illustrative:

"You find out that something, let us say a user has come, maybe he says this is not working, or a programmer has done me wrong or something pinpointed to someone, you will find out that person will not come to you and say a user has come and said this, this may be because that person will think, what will my manager or my supervisor say about me? What are the consequences when it comes to it ...?. So sometimes, because of that, we will never get to have that information if users do not report. We sometimes hear such issues through surveys, and that is when you will get that negative information." (Participant 8, Project Manager)

In line with this, one of the participants highlighted that the project team members should know that there are consequences if improper reporting is done or if reporting is not done at all. This calls for a code of conduct that ensures project team members are not left in the dark or tormented by fear of the unknown, unsure what could befall them. However, one project manager believed that punishment worsens the situation when negative news is not reported. The participant fears that using punishment will worsen the situation because project team members will be unsure whether to report, out of fear of reprisals. The participant said,

"Honestly, if we try to use punishments, we will confuse the project team members more. Why am I saying this? If they report bad news, we managers usually, if not always, attribute that to the reporter and at the end, they are labelled incompetent as compared to those who do not report." (Participant 11, Project Manager)

Similarly, project team members opined that punishment is not effective, but instead, they encouraged the use of rewards rather than encouraging BNR:

"People should not really be punished for making mistakes unless it is proven that it's deliberate or incompetence. Mistakes cultivate innovative minds. However, rigidity in our institutions makes the environment tense such that people decide to talk flowery things and hide bad news." (Participant 10, System User)

Being punished for making mistakes or failing to implement things correctly always stifles innovation. Reference [26], in line with this user perception, states that fear of failure is a huge psychological barrier that stifles innovation and risk-taking in modern organisations. An organisation's culture and individual psychology greatly influence how people perceive things. There are cases where punishments apply, for example, where a project member becomes known for derailing team progress through mistakes. In some organisational cultures, making mistakes is considered a sign of incompetence or negligence [26]. This stigma can dampen employees from trying new approaches or thinking outside the box for fear of consequences.

2) Unfair distribution of work

Reference [28] highlight how the high labour turnover for developers has become widespread in organisations, leading to knowledge loss, reduced productivity, and increased defects. The increase in the hardworking developer's workload is one of the triggers of high labour turnover. If project leaders fail to distribute work fairly, high labour turnover becomes the norm. To avoid overload, project team members would rather remain silent about a failing course of action when working on a software project, lest the bug they report becomes the developer's burden. This factor is highly related to understanding team dynamics in software project development. Fairness in workload distribution by project managers is critical. All participants (project team members) from the three institutions raised concerns about workload. The participants highlighted that:

"As a developer, you need to be careful because you will give yourself unnecessary tasks and pressure through reporting bad news. Usually, if you raise a matter, it becomes your baby, so it depends on how this is handled after reporting. If it is going to be dealt with fairly, then it will be better, and it motivates me to report again next time on negative news." (Participant 15, Project Team Member)

Two project managers felt they were executing their duties quite well, with professionalism, in terms of work distribution. One project manager acknowledged that, as humans, fairness in work distribution is a challenge, since by default, hard workers are usually assigned more work. Overload is said to affect project timelines and software quality. One project manager acknowledged that, if a project team member raises a query, the boomerang effect is to give the problem to that person to fix automatically.

"Sometimes you find out that, let's say, there is a component which is not working, and maybe I, as a project leader, did not even notice it. The developer who worked on that issue can keep silent about it because they may know it will be an additional task or, let me say, a burden added to them. So they would rather pretend as if all is well. Usually, it will come up as some request from system users who will be testing or using the system, and the burden will lie on me as a project manager to whom I can assign the task because maybe I would have forgotten who was working on that particular job." (Participant 5, Project Manager)

The responses from participants require urgent attention from managers to boost the team members' reporting confidence. Project managers must evenly distribute the workload among team members or somewhat incentivise the hard workers. This study has identified the unfair distribution of workload as an inhibitor of bad news reporting by software project team members. On the contrary, team members who are not assigned many tasks may lose their morale and self-esteem. In turn, this triggers silence during meetings. Team leaders must therefore understand every team member and recognise that strengths and weaknesses are not uniform; hence, work with them at a more personal level.

3) Lack of knowledge and skills

Almost all project managers posited that enhancing personal skills is important in any work environment, as most of their project team members lack the expertise to perform their duties. Appropriate training on the technologies and platforms used increases the chances of software project success. If a project team member lacks knowledge on how to approach a task, it will be difficult to report any divergences or issues affecting the project. One participant highlighted that:

"Also, if someone knows that he/she cannot do the task, disinclination can come in, and one would say, 'Let a sleeping dog lie'; if it comes back to me, I will take it up from there. Looking at it, I can say they fear exposure of incompetence." (Participant 11, Project Manager)

About half of the project team members highlighted the need for skills development. These participants felt that programming software without a full understanding of the programming language was like being thrown into the deep end. The project manager and some system users noted that the people involved in a project need thorough training to perform their duties effectively.

"We lack skills for conducting IS audits, so little has been done. The things we normally talk about are not necessarily the software itself but the output, so it is still a challenge. However, when we have the capacity, the developers can take it from this department. We also conduct follow-up audits to check that the things that we have recommended have been implemented." (Participant 10, System User)

Due to a lack of skills, many project team members reluctantly avoid reporting any bugs in the software. In this case, the project team members showed that a lack of expertise on how to fix the bug automatically hinders one from reporting the project's state. Hence, they fear being labelled incompetent and resort to silence.

VI. Proposed Framework for the Drivers and Inhibitors of Bad News Reporting on Software Projects

The results for this study were used to come up with a suitable framework shown in Fig. 1. below. The framework was derived from the study's findings and clearly shows the drivers and inhibitors of Bad news reporting, as well as the implications of each. When enablers are enhanced, members are more motivated to report bad news in software projects. If status reporting is done correctly, costs are minimised, deadlines are met, bug-free software is rolled out, and, finally, the software is successfully implemented. On the contrary, inhibitors of BNR lead to project failure, brain drain, prolonged project life cycles, and escalation.

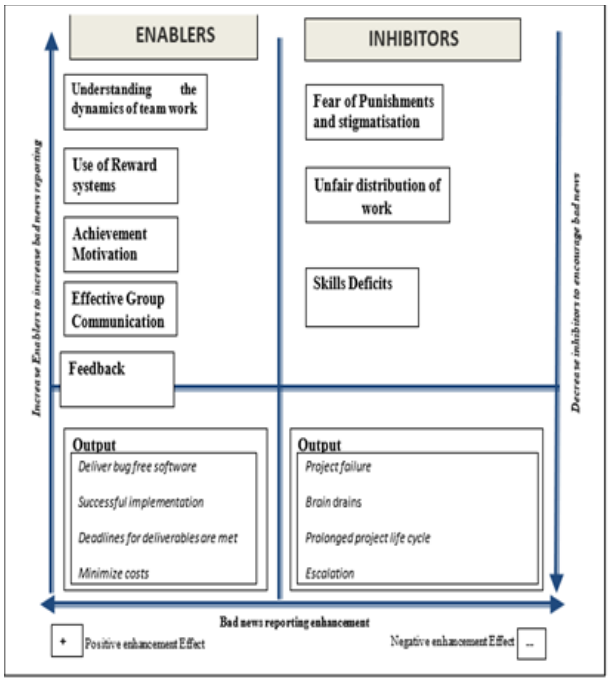

Fig. 1. A framework for enablers and inhibitors of bad news reporting

VII. Areas for Future Research

While this study provides significant insights into the reporting behaviours within Zimbabwean state universities, several limitations must be acknowledged:

A. Geographic and Institutional Scope

This study was limited to three specific state universities in Zimbabwe. While these provide a representative view of the Education 5.0 landscape, the findings may not be directly generalisable to private universities or institutions in different regions.

B. Sample Size and Qualitative Nature

The research followed a qualitative approach with a targeted sample of 15 participants. Although intensity sampling provided information-rich scenarios on bad news

reporting, a statistically representative sample of all IT professionals in the country would be more valuable through a quantitative research approach.

VIII. Conclusions and Implications

This study sought to identify and develop a framework for the enablers and inhibitors of bad news reporting (BNR) in software projects at Zimbabwean state universities. The findings unravel a tension between a deeply ingrained culture of fear and a strong, aspirational desire for psychological safety and professional growth. Reference [27] state that modern organisations need to instil a culture that supports and nurtures innovation among employees. Project team members should have the liberty, with their seniors' approval, to try new things and make mistakes to cultivate a spirit of innovation, for the betterment of the institutions and the success of the projects. Reference [26] notes that in a world where change is the only constant, adaptability and a proactive approach to challenges are the keys to sustained success and innovation. The study results highlighted more enablers of bad news reporting than inhibitors, suggesting that institutions can employ strategies to curb the 'mum effect' on software project development

References

[1] C. M. Toquero, "Challenges and opportunities for higher education amid the COVID-19 pandemic: The Philippine context," Pedagogical Research, vol. 5, no. 4, 2020.

[2] M. Keil and C. Park, "Bad news reporting on troubled IT projects: Reassessing the mediating role of responsibility in the basic whistle-blowing model," J. Syst. Softw., vol. 83, no. 11, pp. 2305–2316, 2010.

[3] Miro, "What is project status reporting?" Miro.com, 2025.

[4] J. Natovich, Z. Derzy, and R. Natovich, "Positive psychology in information technology project management: the case of bad news reporting," Int. J. Inf. Technol. Proj. Manag., vol. 4, no. 4, pp. 35–50, 2013.

[5] H. J. Smith and M. Keil, "The reluctance to report bad news on troubled software projects: a theoretical model," Information Systems Journal, vol. 13, no. 1, pp. 69–95, Jan. 2003.

[6] [6] S. Sarkar, M. Keil, and H. K. Lee, "Context and Timing in Bad News Reporting: An Exploratory study in IS projects," 2015.

[7] M. Keil, G. P. Im, and M. Mähring, "Reporting bad news on software projects: the effects of culturally constituted views of face-saving," Inf. Syst. J., vol. 17, no. 1, pp. 59–87, 2007, doi: 10.1111/j.1365-2575.2006.00235.x.

[8] M. Keil and C. Park, "Bad news reporting on troubled IT projects: Reassessing the mediating role of responsibility in the basic whistleblowing model," Journal of Systems and Software, vol. 83, no. 11, pp. 2305–2316, Nov. 2010.

[9] M. Sandelowski, "Whatever happened to qualitative description?" Res. Nurs. Health, vol. 23, no. 4, pp. 334–340, 2000.

[10] Z. A. I. R. der Prüfungen, "Status Reporting in IT Projects," 2013.

[11] S. A. Licorish, "Understanding the Effect of Agile Practice Quality on Software Product Quality," IEEE Trans. Softw. Eng., 2025.

[12] R. O'Connor and S. Basri, "The effect of team dynamics on software development process improvement," International Journal of Human Capital and Information Technology Professionals (IJHCITP), vol. 3, no. 3, pp. 13–26, Jul. 2012.

[13] T. Dingsøyr, P. Schneider, G. R. Bergersen, and Y. Lindsjørn, "Challenges in understanding the relationship between teamwork quality and project success in large-scale agile projects," in Proc. 17th Int. Conf. Cooperative and Human Aspects of Software Engineering (CHASE), 2024, pp. 51–56.

[14] J. Bratton and J. Gold, Towards critical human resource management education (CHRME): a sociological imagination approach, Work, Employment and Society, vol. 29, no. 3, pp. 496–507, 2015.

[15] A. Thompson and J. Ryan, Employee Reward, 2014.

[16] P. Ayagre and J. Aidoo-Buameh, "Whistleblower reward system implementation effects on whistle-blowing in organisations," Eur. J. Account. Audit. Finance Res., vol. 2, no. 1, pp. 80–90, 2014.

[17] Q. Hu, H. Adam, S. Desai, and S. Mo, "Turning a blind eye to team members' unethical behaviour: The role of reward systems," J. Bus. Ethics, vol. 194, no. 2, pp. 297–316, 2024.

[18] S. Beecham, N. Baddoo, T. Hall, H. Robinson, and H. Sharp, "Motivation in Software Engineering: A systematic literature review," Inf. Softw. Technol., vol. 50, no. 9-10, pp. 860–878, 2008.

[19] N. Baddoo, T. Hall, and D. Jagielska, "Software developer motivation in a high maturity organisation: a mixed methods case study," Softw. Process Improv. Pract., vol. 11, no. 3, pp. 219–228, 2006.

[20] D. Mrowiec, "Factors influencing internal whistle-blowing. A systematic review of the literature," J. Econ. Manag., vol. 44, pp. 142–186, 2022.

[21] J. A. Fulkerson et al., "Family home food environment and nutrition-related parent and child personal and behavioural outcomes... plus program: a randomised controlled trial," J. Acad. Nutr. Dietetics, vol. 118, no. 2, pp. 240–251, 2018.

[22] [22] S. Van Oordt and E. Guzman, "On the role of user feedback in software evolution: A practitioners' perspective," in 2021 IEEE 29th International Requirements Engineering Conference (RE), 2021, pp. 221–232.

[23] M. J. Cuellar, "The nature of project success," Inf. Syst. Faculty Presentations, Georgia Southern Univ., 2013.

[24] Y. K. Lee, C. T. Chang, Y. Lin, and Z. H. Cheng, "The dark side of smartphone usage: Psychological traits, compulsive behaviour, and technostress," Comput. Human Behav., vol. 31, pp. 373–383, 2014.

[25] M. Bugdol and K. Nagody-Mrozowicz, Management, Organisation and Fear: Causes, Consequences and Strategies, Routledge, 2020.

[26] D. Robey, "Blowing the whistle on troubled software projects," Commun. ACM, vol. 44, no. 4, pp. 87–93, 2001.

[27] M. Shirinzad and K. Dixon, "Assessing Corridor Safety Performance at Approaches to Freeway Interchanges Using Lane Change Data," Transp. Res. Rec., vol. 2678, no. 11, pp. 1768–1776, 2024.

[28] M. O. F. Munshi and S. Barai, "IP rights to AI-generated works: Barriers presented by existing law and reforms needed," Can. J. Bus. Inf. Stud., vol. 4, no. 2, pp. 37–46, 2022.

[29] E. Mirsaeedi and P. C. Rigby, "Mitigating turnover with code review recommendation: Balancing expertise, workload, and knowledge distribution," in Proc. ACM/IEEE 42nd Int. Conf. Softw. Eng. (ICSE), 2020, pp. 1183–1195.

[30] D. Robey, "Blowing the whistle on troubled software projects," Communications of the ACM, vol. 44, no. 4, pp. 87–93, Apr. 2001.

[31] M. Keil, J. Mann, and A. Rai, "Why software projects escalate: An empirical analysis and test of four theoretical models," MIS Quarterly, vol. 24, no. 4, pp. 631–647, Dec. 2000.