

CHAPTER

7

IMPROVING USER ACCEPTANCE TESTING IN RESEARCH SYSTEMS

*Suraya Miskon, Nani Amalina Zulkanain,
Zuridah Saadan, and Noor Fazilah Zakaria*

7.1 INTRODUCTION

User Acceptance Testing (UAT) is a test conducted for end users to accept or reject the system when all the development works have been completed (Alsoub et al., 2018). It is conducted by end users—typically staff or company employees who interact directly with the system to verify that all implemented functions operate as intended (Vanesha et al., 2024). The testing process focuses on acceptance of the outcome of the system (Loss et al., 2019). Testing the end-product does not indicate that the product functions properly under all possible conditions but can only show that it does not function improperly under specified conditions. Hence, the purpose of UAT is to examine and execute the developed system to detect the faults as early as possible during the system development process to ensure the requirements are met within the expected delivery, schedule time and budget.

This UAT practise is necessary to identify any error as an intense preparation prior the launch of the system to end users (Suman & Sahibuddin, 2019). As development teams nowadays apply Agile methods, the requirements which are changed during the project may or may not be communicated effectively to the developers and main

users thus, affect the system outcome. Therefore, this study sought to gather user feedback on challenges experienced by developers' team and main users during acceptance testing sessions and discover the solutions to overcome these challenges. All these experiences from both developers' team and main users become the main outcome for this research in finding suggestions of solutions from common challenges found during UAT based on their experience.

7.2 SOFTWARE TESTING AND USER ACCEPTANCE

Software testing methods are procedures used to test the system efficiently by the developers' team and main users. Main UAT will involve the main user and all comments or problems will be fixed by the developer before the release. However, before the main UAT, the developer team also performed an acceptance test known as Alpha test to improve the quality of code and system respectively before showing it to the main user during the main UAT which is called Beta test.

7.2.1 Alpha Testing

Alpha Testing is a type of acceptance testing performed by the developers under a controlled environment (Goel & Gupta, 2014). All developer teams will consistently be involved in the development phase and consistently test the effectiveness of the system before main UAT is held. Only the developer team is involved in this testing to improve the quality of the system. Error, system crashes and plenty of bugs are expected in Alpha Testing.

7.2.2 Beta Testing

User Acceptance Testing (UAT) is also familiar with Beta Testing which is available to test by the owner of the system and main users, developers then solve the problem before the release. The developer only involves just to be prepared with any problem happening with the system and assist main users during Beta Testing. This process helps identify and

remove defects. Beta Testing is also known as pre-release testing (Goel & Gupta, 2014). Fewer system crashes, some bugs and requests for new requirements are expected in this Beta Testing.

7.3 INVESTIGATING THE RESEARCH MANAGEMENT SYSTEM IMPLEMENTATION

This study focuses on the implementation of a Research Management System (RMS) in one of the research universities in Malaysia. The RMS is a web-based system that has internal users from various level management and department and external users. Therefore, there is complexity in the process of application and approval which include various users and approval levels.

The survey is divided into three sections which are general information, challenges of UAT and suggestions of improvement. Developer team which involves in Alpha Testing experience received Survey 1 which focuses on questions regarding their experience in Alpha Testing. Meanwhile, Survey 2 involved by main users only focuses on experience during Beta Testing since they are only involved in main UAT.

Overall, 10 responses received for Survey 1. All targeted respondents are from developer teams such as programmers and system analysts which are involved in the development phase of the system. Meanwhile, seven responses received from Survey 2 and all targeted respondents are main users of the system in the respective research university which consist of research manager, assistant research manager, and appointment clerk.

7.4 ANALYSING USER FEEDBACK FROM ALPHA AND BETA TESTING

This section summarises the responses of both Survey 1 and Survey 2. Discussions of the results are stated in this section to justify the detail of responses. All results are divided into challenges and suggestions of improvement.

7.4.1 Challenges of Alpha Testing

As mentioned in Section 7.3, Alpha Testing is only involved by the developers' team. Survey questions are categorised into two challenges which are requirement and system error as shown in Table 7.1.

Table 7.1 Category of challenges of Alpha Testing

Tester	Category of Challenges	Description
Developers	Requirement	I realise new changes/requirements needed to improve system function and system interface.
		I receive unexpected changes to fix during the final test system.
	System Error	I realised a new error suddenly appeared during the final test by the technical team.

Figure 7.1 shows that 50% of developers agree, and 10% of developers strongly agree that new changes or requirements are needed to improve system function and interface in the development phase. This result shows that developers are consistently improving systems once systems are updated with new functions to ensure the system meets the overall requirement as needed.

Half of the developers (50%) agreed that new, unexpected internal errors emerged during the final test prior to Beta Testing. This aligns with the 60% of developers who reported receiving unexpected system changes to fix during this same testing phase.

Overall, all developers provided an “agree” response to all the challenges stated. This indicates that such challenges are always present before the Beta Testing phase. These issues need to be addressed first before releasing the system to the main users.

However, in any system development process, challenges are inevitable and may vary according to the level of system complexity, technical limitations, facility constraints, or other unexpected situations that occur during the testing phase. Therefore, several actions and suggestions can be implemented to overcome or minimise these

challenges faced by the developers. For further details, refer to Section 7.4.3, which discusses the developers' suggested improvements to address these challenges.

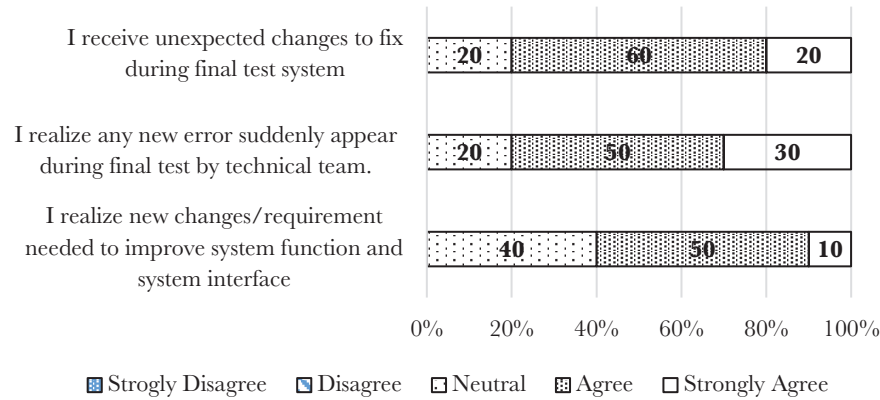


Figure 7.1 Developers' responses on challenges of Alpha Testing

7.4.2 Challenges of Beta Testing

Table 7.2 is the category of Beta Testing challenges rated in the survey. Another category is the added challenges stated by respondent and categorised into groups based on description stated by respondent.

Figure 7.2 shows results of Beta Testing challenges rated by main users. In terms of requirement, 43% rated and strongly agree respectively by main users that new changes or requirements need to improve system function and system interface, however overall, 42% agree that the system follows the standard and requirement as needed by the owner.

This result indicates that all requirements were well collected and effectively performed during the development phase. However, it is evident that users have different perspectives. In terms of the time provided, 57% of the main users gave a neutral response, suggesting that the perception of time may be subjective and varies among users.

For the technical support category, 57% of the main users strongly agreed that the technical staff provided sufficient assistance during the

UAT session. Similarly, 57% strongly agreed that the briefing given was clear and easy to understand. These results indicate that the developers' active involvement during the Beta Testing phase effectively guided and supported the main users.

Table 7.2 Category of challenges of Beta Testing

Tester	Category of Challenges	Description
Main Users	Requirement	I realise new changes/ requirements needed to improve system function and system interface during UAT.
		The system follows the standard and requirement as needed by the owner.
	Technical Help	Technical staff will assist you if there is any problem happening during the session.
		All briefing about system function and manual is clear and understandable.
	UAT Script	The script is helpful to test the system.
		It is easy to manage between system testing and writing comments in the script.
		The UAT script is easy to understand.
	Facilities	The computer provided is in good condition.
	Time	Time provided is enough to complete all test cases provided.

For the facilities category, 42% of the main users responded neutral, while another 42% strongly agreed that all facilities were in good condition and ready for use. This finding suggests that not all users encountered issues related to facilities. This observation aligns with the findings of Suman and Sahibuddin (2019), who noted that limited hardware or device capabilities could lead to difficulties during UAT sessions.

For challenges about the UAT script, 57% main users rated agree and strongly agree respectively regarding script that is helpful to test the system. 42% of main users rated strongly agree that it is easy to manage testing and writing comments in the script while 57% rated agree that the script is easy to understand. These results supported by Suman and Sahibuddin (2019) that most users during UAT lack the user's knowledge

thus, testing should be conducted in a proper way with instruction and right method.

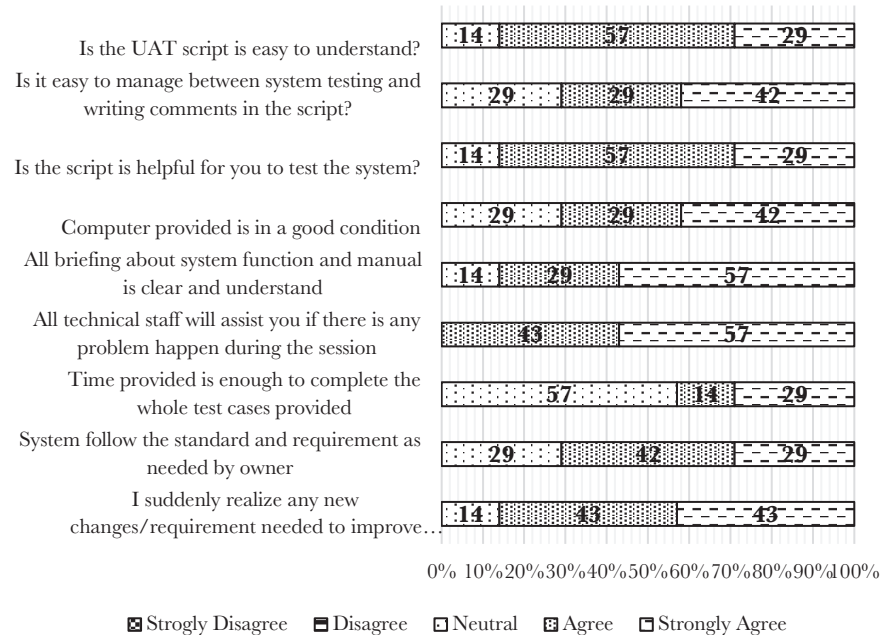


Figure 7.2 Main users' responses on challenges of Beta Testing

7.4.3 Suggestions of Improvement for Alpha Testing

Based on responses from Survey 1, Table 7.3 shows the suggestions of improvement for Alpha Testing suggested by developers' team. All suggestions are categorised into two categories which are communication and progress monitoring.

In summary, developers focus on communication and progress monitoring platforms. A communication platform is defined as the channel of information given that eventually affects the output and progress of the current development. Developers' need to be informed about the current plan or any changes made instantly since all development applied an Agile framework that is flexible of any changes anytime as requested. This is supported by Loss et al., (2019) that

criticism and complaints indicate that customers are engaged with the new system thus, action must be taken at an early stage.

Along the development phase, any error needs to be figured out for debugging and correction. This action needs to be monitored to ensure that different developers with different tasks achieve the goal in each task aligned with time given. A centralised and organised communication platform will help developers to receive information quickly and easily for monitoring.

Table 7.3 Suggestions of improvement for Alpha Testing

Tester	Category of Improvement	Description
Developers	Communication	Provide platform for communication plan and progress
	Progress Monitoring	Programmers have to re-check error
		System analyst needs to monitor and manage progress

7.4.4 Suggestions of Improvement for Beta Testing

Based on responses from Survey 2, Table 7.4 shows the suggestions of improvement for Beta Testing suggested by main users. All suggestions are categorized into three categories.

Table 7.4 Suggestions of improvement of Beta Testing

Tester	Category of Improvement	Description
Main Users	UAT Script	Online script instead of paper-based script
		Short and simple test case
	Time	Short time session
	Briefing session	Firm briefing session of system introduction

For main users, these questions are rated in selection of choices, 100% rated for short and simple test cases while 43% of main users' rate for online script instead of paper-based script. 29% of main users rated short time sessions and 57% rated for firm briefing sessions, respectively.

The results indicate that the use of online test scripts and shorter session durations are the preferred suggestions for improvement. All main users involved in the UAT were university staff who participated over several days, which included an introduction, UAT activities, and a Question and Answer (Q&A) session. Therefore, adopting an online test script is reasonable, as it can reduce both the overall time required and the need for face-to-face meetings.

This finding is supported by Suman and Sahibuddin (2019), who stated that extended UAT durations can be challenging for users, especially when they have multiple work commitments as staff members. Furthermore, conducting UAT sessions online can help minimise paper usage, contributing to a more efficient and sustainable testing process.

7.5 CONCLUSION

This study examined the challenges and improvement strategies of UAT within a research university context, focusing on both the Alpha and Beta Testing phases of a RMS. Findings from the perspectives of developers and end users revealed recurring issues such as communication gaps, unclear requirements, unexpected system errors, time constraints, and limited accessibility to testing materials and facilities. Additionally, an extension in the time required to complete tasks may negatively impact the reputation of the developing companies from the client's perspective (Raja Pillai & Hemamalina, 2022). These challenges often stem from the dynamic nature of system requirements, varying levels of user understanding, and the absence of structured communication and documentation during the development process.

From the developers' perspective, frequent requirement changes during the Agile-based development process led to difficulties in maintaining consistent progress tracking and synchronising updates.

Without a centralised communication channel, information delays caused rework and inconsistencies across testing cycles. Meanwhile, the Beta Testing phase revealed users' challenges in understanding testing scripts, managing limited time, and balancing UAT responsibilities with daily administrative duties. Although technical support and system briefings were helpful, users emphasised the need for clearer instructions, concise test cases, and more accessible testing formats.

To overcome these issues, this study proposes several strategies: establishing structured communication and monitoring platforms, introducing online UAT scripts to enhance flexibility and reduce paper dependency, and organizing shorter, more focused testing sessions accompanied by clear briefing materials. Implementing these improvements can strengthen collaboration, streamline feedback loops, and enhance the overall quality and efficiency of the UAT process.

Ultimately, successful UAT relies not only on technical precision but also on strong collaboration and continuous engagement between developers and users. Effective UAT practises serve as a bridge between technology and people, ensuring that systems are not only functional but also aligned with user needs and institutional goals. In conclusion, a well-structured, communicative, and user centred UAT framework is vital in ensuring the success and sustainability of system implementation in higher education environments.

REFERENCES

- Alsoub, K. R., Alrawashdeh, A. T., & Althunibat, A. (2018). User acceptance criteria for Enterprise Resource Planning software system. *International Journal of Innovative Computing, Information and Control*, 14(1), 297–307.
- Goel, R., & Gupta, N. (2014). Survey on acceptance testing technique. *International Journal of Software and Web Science*, 1(7), 20–23.

- Loss, S., Ciriello, F. R., & Cito, J. (2019). *Beware of disengaged user acceptance in testing Software-as-a-Service* [Conference session]. In 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion) (pp. 298–299). IEEE. <https://doi.org/10.1109/ICSE-Companion.2019.00123>
- Raja Pillai, N. S., & Hemamalina, R. R. (2022). Hybrid user acceptance test procedure to improve the software quality. *The International Arab Journal of Information Technology*, 19(6), 956–964. <https://doi.org/10.34028/iajit/19/6/16>
- Suman, R., & Sahibuddin, S. (2019). *User acceptance testing in mobile health applications: An overview and the challenges* [Conference session]. In Proceedings of the 2nd International Conference on Information Science and Systems (pp. 16–19). Association for Computing Machinery. <https://doi.org/10.1145/3322645.3322681>
- Vanesha, N. A., Rizky, R., & Purwanto, A. (2024). Comparison between usability and user acceptance testing on educational game assessment. *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, 13(2), 210–215. <https://doi.org/10.32736/sisfokom.v13i2.2099>