

CHAPTER

8

STREAMLINING SOFTWARE DEVELOPMENT THROUGH TICKETING TOOLS

*Noor Aklima Harun, Farizie Abdul Razak,
Noor Fazilah Zakaria, and Sharin Hazlin Huspi*

8.1 INTRODUCTION

High-performing software development relies not only on tools but, more importantly, on people. Its success depends on how analysts, programmers, and system owners collaborate, make collective decisions, and share knowledge across team boundaries.

In many organisations, the greatest challenges in software development are not purely technical but organisational, that is on how to ensure clear communication, consistent task ownership, and timely coordination among multiple contributors. When these social and procedural elements are weak, projects experience delays, redundant work, and reduced quality.

Common issues include unclear task allocation, communication breakdowns between developers and system owners, overlapping responsibilities, inconsistent documentation, and limited traceability of decisions. These problems are amplified in environments where multiple systems and units interact, as seen in large institutions such as universities. Without an integrated platform to manage requests, discussions, and

progress tracking, essential issues often get lost or delayed, ultimately affecting the delivery and sustainability of developed systems.

To address these coordination challenges, structured collaboration tools, such as helpdesk ticketing systems, play a crucial role. A ticketing system provides a centralised environment to collect, organise, and track requests or issues, allowing teams to manage work transparently and systematically. Beyond mere technical support, such systems strengthen accountability, clarify ownership, and promote a culture of responsiveness within development teams.

Thus, the Helpdesk Ticketing System (HTS) was introduced to the development team at Universiti Teknologi Malaysia (UTM), where it serves as an integrated communication and coordination platform between system owners, developers, and researchers. Researchers can report technical issues or suggest improvements directly through the system, ensuring that their requests reach the right personnel.

Before HTS was introduced, communication relied heavily on phone calls, WhatsApp messages, and emails. These informal channels were difficult to monitor, prone to miscommunication, and often resulted in delayed responses or lost information. The absence of a standardised process created uncertainty over responsibilities, made tracking progress difficult, and hindered knowledge sharing among development teams.

Since the implementation of HTS, communication and coordination within UTM's system development environment have become significantly more organised. Records are systematically tracked, issues are assigned to the appropriate personnel, and notifications help ensure that no requests are overlooked. The system has also improved handovers and reduced turnaround times for resolving issues.

This chapter focuses on the team perspective of system development by highlighting the common hurdles faced by developers, such as unclear requests, mixed priorities, dependencies on other units, and loss of contextual information. It also explains how the HTS helps mitigate these issues by standardising request intake, clarifying ownership,

improving inter-team communication, and enhancing overall development efficiency. Through UTM's experience, this chapter demonstrates how a structured ticketing approach can transform fragmented workflows into coordinated, accountable, and high-performing development practises.

8.2 THE EVOLUTION OF TICKETING SYSTEMS FOR TEAM COORDINATION

Effective software development depends on more than technical proficiency, as it relies also on how teams communicate, coordinate, and manage shared responsibilities. As development projects grow in complexity, team members often face difficulties maintaining visibility of tasks, prioritising issues, and ensuring accountability across multiple roles. These challenges can lead to duplicated work, unresolved problems, and reduced efficiency in both development and maintenance phases.

In this context, helpdesk or ticketing systems have evolved beyond traditional customer service functions to become critical coordination tools within software teams (Wang et al., 2011; Paramesh & Shreedhara, 2019). They enable structured communication between analysts, programmers, and system owners by centralising issue tracking, assigning responsibilities, and documenting decision trails. Such systems provide transparency, allowing every team member to monitor progress, understand dependencies, and contribute to faster problem resolution.

Recognising the importance of systematic collaboration, the Deputy Vice Chancellor's Department (Research and Innovation) in UTM explored various tools to improve communication and workflow management among its development teams. Several platforms, including Spiceworks (Spicework Documentations, 2021), Trello (Trello Documentations, 2021), and osTicket, were evaluated to determine the most suitable framework for managing technical requests and system support (Rist & Haslam, 2020). These evaluations included analysing team requirements, understanding existing manual coordination processes, and identifying gaps that hindered efficient teamwork. The goal was to adopt a

solution that not only addressed user support but also strengthened collaboration and productivity among developers and system owners involved in UTM's research management systems.

8.2.1 The Role and Structure of UTM Research Computing

The Deputy Vice Chancellor's Department (Research and Innovation) or known as the DVCRI Department is the department that manages innovation research for Universiti Teknologi Malaysia (UTM) researchers. The main services related to information technology are developed and maintained by the Research Computing Section or known as UTMRC.

UTMRC's main portfolio is as a unit that provides Information Technology (IT) related support services to UTM staff, mainly researchers. Services provided include research-related system development, innovation, commercialisation, community and networking, as well as other ICT support. Among the services provided by UTMRC are the development of RADIS 4.0, INNOCOMMS 2.0, UTMLAB Phase 1, e-Press, eActivity, UTMScholars, UTMExperts, and the TNCPI Official Web Portal. Other services include providing support data through Research Support (MyRA data or MyMohes).

For all the developed systems, the system users are divided into two; (1) System owners and (2) Researchers (e.g.: Academic staff, Research Officers/Assistant). The system owner is defined as the department or unit that is responsible for providing user requirements to the developer for system development and subsequently perform the operation, maintenance, and coordination of the overall data on the system. Whereas researchers are individuals who provide input data into the system and communicate directly with the system owner and the system developer.

The collaborative structure within UTMRC is inherently complex due to its multi-layered interactions among system owners, developers, and end-users. Each project often involves several research units with overlapping responsibilities, varied technical competencies, and differing priorities, making coordination and communication a continual challenge. Managing this diversity requires clear workflows, transparent issue tracking, and

structured mechanisms to ensure that decisions, feedback, and technical resolutions are consistently shared across teams.

8.2.2 Research Management Centre Virtual Helpdesk System

Since 2016, UTM has begun to explore tools to facilitate the support system to users. Exploration of these different tools to gain experience and knowledge on how to manage support services is crucial to understand the whole process. Figure 8.1 shows one of the early initiatives to computerise the helpdesk support system (Research Management Centre [RMC], 2016).

Figure 8.1 shows separate systems to support the helpdesk to handle complaints and inquiry regarding all related research systems. RMC Virtual Helpdesk System serves as an online counter for researchers to lodge complaints.

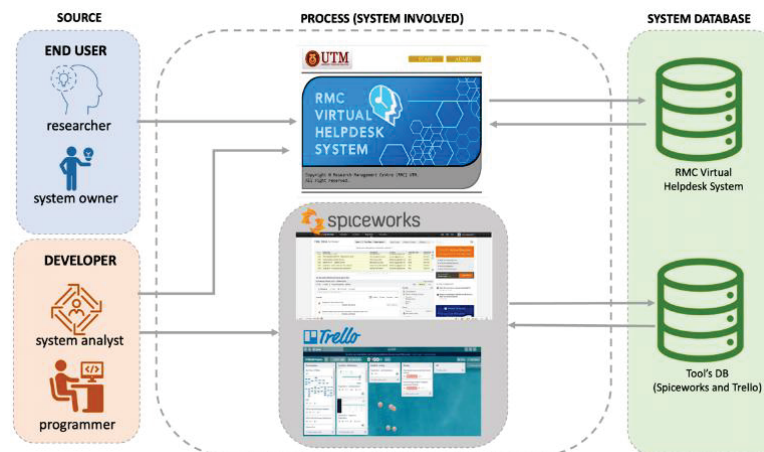


Figure 8.1 Research Management Centre Virtual Helpdesk System flow

Once the complaints and inquiries are recorded, a Customer Service Officer will assign the task to respective personnel: Either the system owner or the development team. However, it was identified that having three different systems to manage the complaints have major weaknesses as specified in Table 8.1.

As seen in Figure 8.1, three different systems were used, and this has made difficulties to all people involved in managing the complaints. There were also redundant tasks and information when using the RMC Virtual Helpdesk System and the two-supporting system (Spiceworks and Trello). This has made the process and management of complaints and inquiries inefficient, and thus risks of lost complaints are higher.

Table 8.1 Comparison of different system or tools

Tools/System	Weaknesses
RMC Virtual Helpdesk System	The customer service officer task is to manually assign the complaints to assigned units.
Spicework IT Desktop (by Spicework)	- The system is used for communication and management of projects between developers only. - Need to re-enter information/ solution or inform the end users manually.
Trello (by Atlassian)	Not an ideal method for managing all types of projects (e.g.: a project with a great number of tasks is not suitable because it is hard to visualise).

These limitations made the RMC Virtual Helpdesk System less favourable among researchers. Most researchers prefer to be in touch with the developers or system owners via email, phone call or various messaging platforms (Figure 8.2).

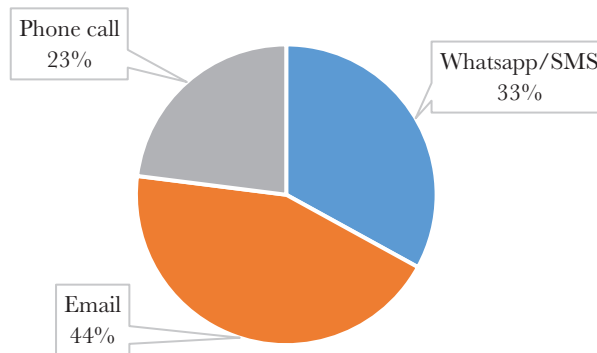


Figure 8.2 Communication method for complaints and inquiries (before HTS)

Emails to personal staff in-charge are considered as the official medium to file complaints or post any inquiries regarding the system. However, email responses are considered slow, and the risk of missed complaints are high. Phone calls and messaging platforms such as Whatsapp, are considered a more personal medium, as it is if the users can get the phone numbers of the personnel to ask.

It was also observed that the response to fix the problem varies from one to more than three days, with 1–2 days having the highest response rate (48%). Improvement was needed to address the weakness of the system as 12% of the complaints took more than three days to be responded and 9% of the complaints were not attended until the users did follow-up to speed up the process.

8.2.3 Team Issues and Challenges in Software Development

As discussed in Section 8.2.2, teams at UTMRC face several day-to-day hurdles that slow development and create extra rework. The issues often come from how information moves between system owners and developers, how tasks are tracked, and how roles are understood. The table below summarises the main challenges, explains what they look like in practise, and gives concrete examples from the UTMRC context.

Table 8.2 Key team challenges, descriptions, and UTMRC context

Challenge	Description	Example in UTMRC Context
Communication Gaps	Miscommunication between system owners, developers, and end-users delays problem resolution.	Researchers contacting different developers via multiple channels.
Task Tracking Difficulties	Without centralised tracking, issues get lost, duplicated, or unresolved.	Multiple systems (emails, WhatsApp, Trello) used before HTS.
Ambiguous Roles	Overlaps between system owners and developers cause unclear responsibilities.	Issues forwarded to the wrong unit or unresolved.

continue

Table 8.2 (Continued)

Challenge	Description	Example in UTMRC Context
Delayed Response	Lack of prioritisation and follow-up leads to longer resolution times.	Some tickets took more than three days to resolve.
Knowledge Silos	Solutions are not documented for future reference.	Same issues reoccur because previous fixes are not recorded.

8.3 TICKETING SYSTEM AS A COORDINATION FRAMEWORK

As UTMRC's portfolio of research systems expanded, the development team faced increasing coordination issues (unclear task ownership, overlapping responsibilities, and inconsistent communication across multiple units) as discussed in the previous sections. To address these challenges, the team implemented a HTS built on the osTicket framework (osTicket Documentation, 2021). Rather than merely functioning as a support tool, HTS serves as a structured coordination platform that integrates the efforts of researchers, system owners, analysts, and programmers as shown in Figure 8.3.

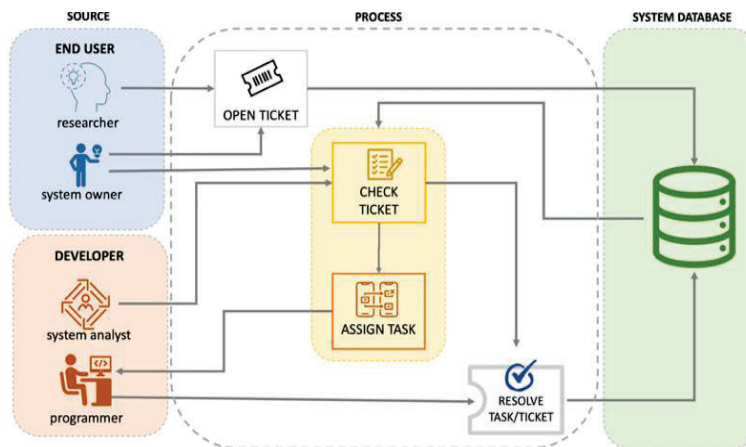


Figure 8.3 Overall framework of UTMRC Helpdesk Ticketing System communication

By standardising the reporting and tracking of issues, HTS ensures that every request is clearly documented, assigned, and monitored until completion. It eliminates ambiguity in roles, where researchers and system owners act as ticket initiators (as end users); and system analysts and programmers (as developers) manage diagnosis, task assignment, and resolution. This structured flow reduces miscommunication, shortens turnaround time, and enhances accountability within and across teams. A more detailed explanation of the framework is shown in Figure 8.4, where it shows the functional flow of the system.

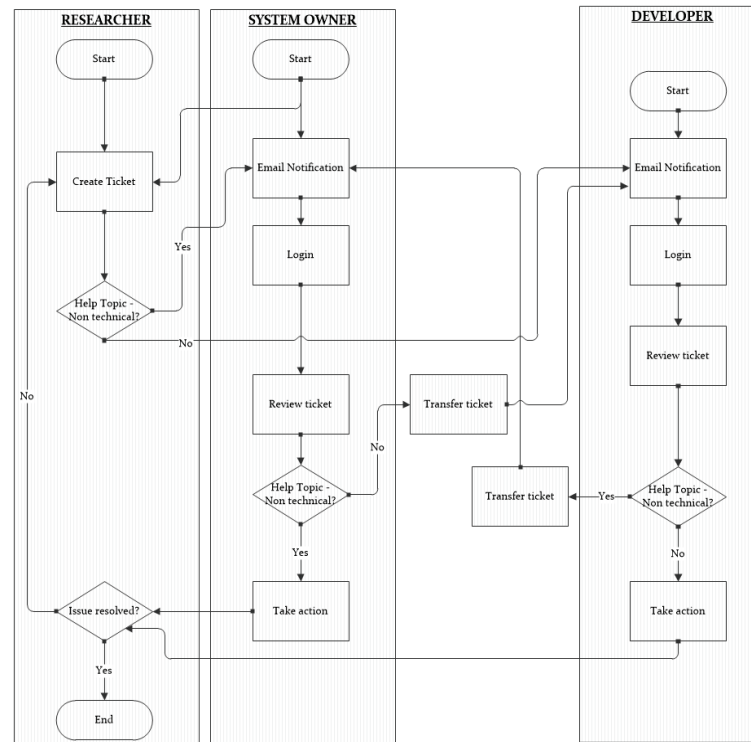


Figure 8.4 Cross functional flowchart for Helpdesk Ticketing System

It is showed that the HTS functions as a socio-technical tool that enables distributed collaboration among diverse teams involved in system development at UTMRC. As illustrated in Figures 8.3 and 8.4, the HTS framework enhances workflow transparency by mapping how

information flows across units and ensuring that every action is traceable. Communication within development teams has become more efficient, as members can track progress, exchange updates, and document resolutions in a shared environment. Beyond improving operational coordination, HTS also facilitates team learning; archived tickets and solution records serve as reference points that help developers recognise recurring issues, refine documentation, and continuously improve their technical and collaborative practises.

8.4 IMPACT ON TEAM PRODUCTIVITY AND COLLABORATION

The introduction of the HTS at UTMRC brought measurable improvement in how development teams collaborated and delivered results. One of the most immediate outcomes was a significant reduction in response times.

Before the system was introduced, issues were often communicated through scattered channels such as emails, phone calls, or personal messaging applications, making it difficult to trace progress or hold specific individuals accountable. With HTS, all issues are logged, categorised, and assigned to the appropriate personnel, ensuring that every request is visible to both the end user (as the requester) and the developer team (Harun et al., 2021).

This is shown in Figure 8.5, where it showed that the number of tickets assigned to be completed within 1–2 days was high, especially for non-transferred tickets. Even for a transferred ticket, the number of days for a ticket to be completed more than 2 days are quite low.

This transparency accelerated communication loops, enabled quicker task prioritisation, and reduced the likelihood of overlooked or duplicated requests.

Equally important was the establishment of clear role boundaries. HTS formalised the interaction chain among researchers, system owners, analysts, and programmers, so that each team understood its responsibilities in problem identification, analysis, and resolution.

This structured workflow not only improved accountability but also minimised confusion over task ownership, a common source of friction in multi-unit development environments. Team members could now focus on their specific responsibilities while maintaining awareness of how their work contributed to the broader system-support ecosystem.

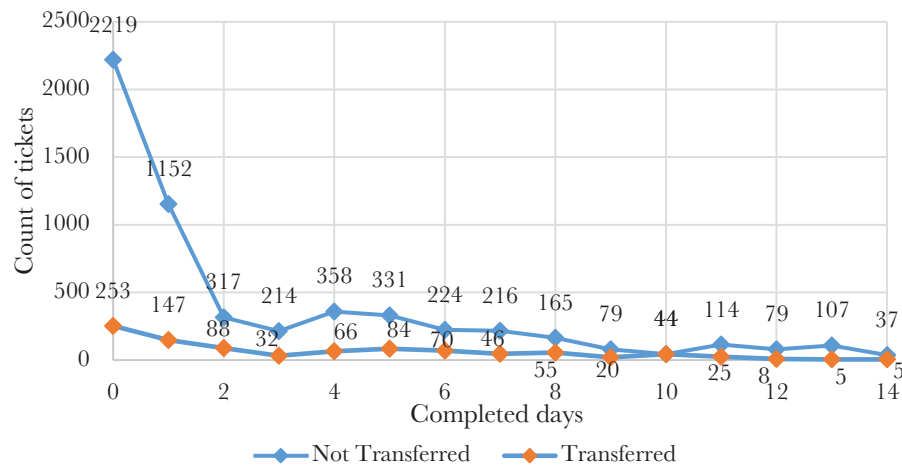


Figure 8.5 Number of tickets and the duration of completed tasks.

8.5 EVALUATING TEAM EXPERIENCE AND SYSTEM EFFECTIVENESS

To assess the impact of the HTS on teamwork and service efficiency, a user satisfaction survey was conducted involving system owners, researchers, and developers within UTMRC. Approximately 64% of respondents were end users, providing a comprehensive perspective on how the system influenced day-to-day collaboration and communication across roles.

As shown in Figure 8.6, the majority of respondents (44%) reported using HTS primarily to record and monitor system-related complaints. The consistent use of HTS across different user groups highlights its growing importance as a centralised repository for documenting and resolving issues in systems managed by UTMRC.

End users noted that the system simplified the process of reporting issues and tracking their progress, while developers observed that tasks were easier to assign, trace, and prioritise. Additionally, about 31% of developers agreed that HTS made it easier to record, monitor, and complete tasks systematically.

As shown in Figure 8.7, both end users and developers expressed overall satisfaction with HTS, indicating that it successfully enhanced communication, task visibility, and accountability. The findings suggest that the system not only improved the operational efficiency of UTMRC's technical teams but also strengthened mutual trust and collaboration between system owners, researchers, and developers. In essence, HTS has become a vital component of the team's socio-technical environment, supporting both productivity and professional communication in system development and maintenance.

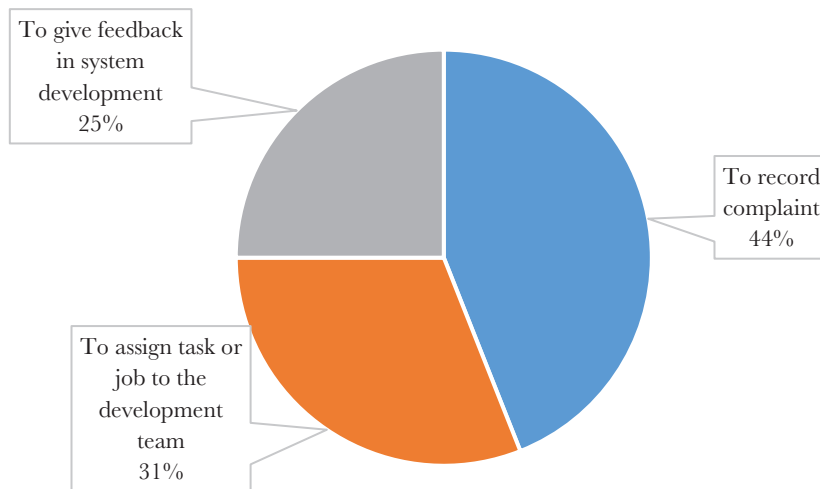
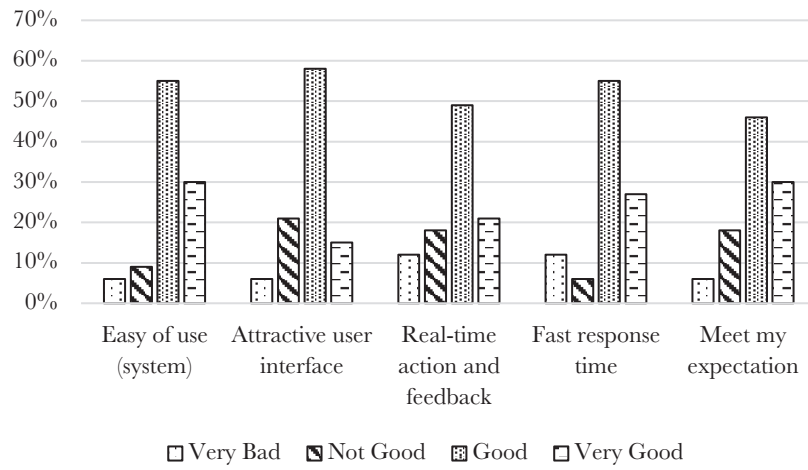
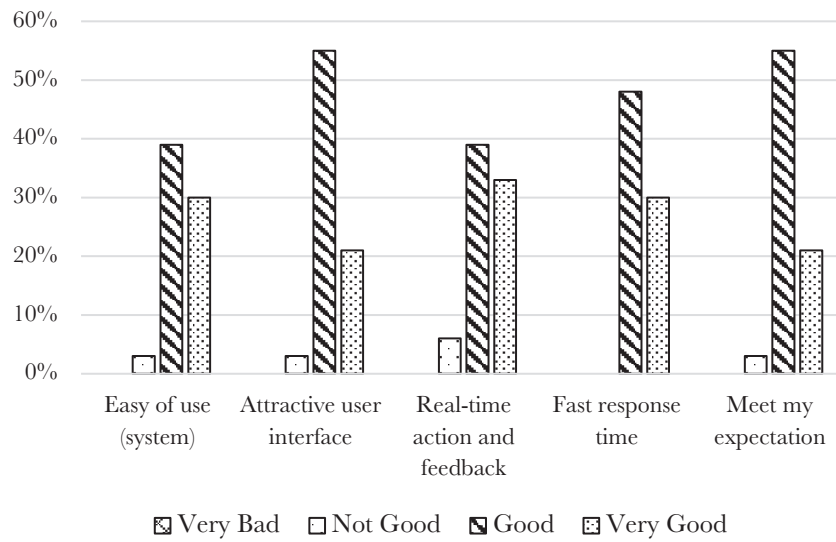


Figure 8.6 Purpose of using UTMRC research support



(a)



(b)

Figure 8.7 Satisfaction level by (a) End Users and (b) Developers

8.6 CONCLUSION

Beyond operational efficiency, the most profound effect of the HTS lies in its role as a learning and knowledge-building platform. Each ticket functions as a record of inquiry, diagnosis, and resolution. Over time, these

accumulated records form an institutional memory that captures not only technical solutions but also decision rationales and communication patterns. Developers can revisit previous cases to identify recurring issues, analyse root causes, and propose preventive measures.

This feedback loop transforms daily troubleshooting into a structured learning process. Teams begin to anticipate potential problems, refine documentation standards, and develop reusable guidelines that enhance system stability. The insights derived from ticket analytics support targeted training, process redesign, and improved coordination between system owners and developers (Harun et al., 2023). Consequently, HTS has evolved from a simple issue-tracking tool into a mechanism for continuous improvement, reinforcing a culture of reflection, shared knowledge, and adaptive learning across UTMRC's development community.

ACKNOWLEDGEMENT

The authors would like to express their appreciation to the UTM Research Computing group, especially the users and developers of UTMRC Helpdesk Ticketing System for supporting this research.

REFERENCES

- Harun, N. A., Huspi, S. H., & A. Iahad, N. (2023). Question classification for helpdesk support forum using Support Vector Machine and Naïve Bayes Algorithm. *International Journal of Innovative Computing*, 13(1), 37–45. <https://doi.org/10.11113/ijic.v13n1.388>
- Harun, N. A., Huspi, S. H. and Iahad, N. A. (2021). *Question classification framework for helpdesk ticketing support system using Machine Learning* [Conference session]. Proceedings of the 2021 7th International Conference on Research and Innovation in Information Systems (ICRIIS), 1–7, doi: 10.1109/ICRIIS53035.2021.9617077.
- osTicket. (2021). *osTicket documentation*. <https://docs.osticket.com/en/latest/>

- Paramesh, S. P., & Shreedhara, K. S. (2019). Automated IT service desk systems using machine learning techniques. In H. S. Saini, R. K. Singh, & S. C. Satapathy (Eds.), *Innovations in Computer Science and Engineering* (pp. 247–254). Springer. https://doi.org/10.1007/978-981-13-2514-4_28.
- Research Management Centre (RMC) (2016). *RMC virtual helpdesk document*, Universiti Teknologi Malaysia (UTM).
- Rist, O. & Haslam, O. (2020). *The Best Help Desk Software for 2020*, PCMag. <https://www.pcmag.com/picks/the-best-help-desksoftware>
- Spicework. (2021). *Spiceworks documentations*. <https://www.spiceworks.com/>
- Trello. (2021). *Trello documentations*. <https://trello.com>
- Wang, D., Li, T., Zhu, S., & Gong, Y. (2011). Ihelp: An intelligent online helpdesk system. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*. 41(1), 173–182. <https://doi.org/10.1109/TSMCB.2010.2048749>