



RESEARCH ARTICLE

Development of a Django-Based Web Application for Quran Memorization Recapitulation and Automated WhatsApp Reporting to Parents

Muhammad Hayat Aqil Syiraj^{1*}, Kamil Malik¹, Wali Ja’far Shudiq¹

¹Department of Informatics Engineering, Faculty of Engineering, Universitas Nurul Jadid, Probolinggo, Indonesia

*Corresponding author: hayat.aqil202@gmail.com

ARTICLE INFO	ABSTRACT
<p>Article history: Received: 26 October 2026 Revised: 09 November 2026 Accepted: 16 November 2026 Published: 01 December 2026</p> <p>Keywords: Tahfidz monitoring System Usability Scale Waterfall model Parent communication Educational information system</p>	<p>Memorization (tahfidz) records in many Islamic boarding schools are still kept in paper notebooks and summarised by hand before they are sent to parents, which is slow and error-prone. In the studied tahfidz unit, mentors estimated that preparing and sending monthly reports to about 85 guardians took four to seven hours. This study aims to develop a web application that integrates memorization recording, automatic report generation, progress charts, and automated report delivery to parents through WhatsApp. The system was built with the Django framework and an SQLite database following the Waterfall model, with requirements derived from observation, interviews, and literature through triangulation. Reports are sent through the Fonnite gateway interface and every delivery is logged. Ten tahfidz mentors evaluated the application with a 12-feature functional test on a five-point scale, the System Usability Scale, and a ten-item user experience questionnaire on a seven-point scale. The functional test gave a mean of 4.47 out of 5 (89.33%), and no function failure was reported. The System Usability Scale score was 73.0 (standard deviation 7.1), within the good band, and the experience questionnaire averaged 5.57 out of 7, with visual design rated lowest (5.20). The results indicate that the application is functional and acceptable to its primary users in this single-institution setting. Time savings and parent-side outcomes were not measured and remain to be evaluated.</p>

I. INTRODUCTION

Tahfidz education, the systematic memorization of the Qur’an, is a core activity in Islamic boarding schools (*pesantren*). Students (*santri*) regularly present new memorization (*ziyadah*) and review earlier memorization (*muroja’ah*) to a mentor, and the progress is usually written by hand in a submission notebook and summarised periodically for parents. Paper-based record keeping in educational institutions is prone to recording errors and inefficient data handling [1]. Web-based information systems, in contrast, have been reported to improve the effectiveness of academic data management and to speed up access to information [2]. Instant messaging can further accelerate communication between an institution and its stakeholders when it is used as an automatic notification channel [3], and graphical presentation of data supports monitoring and decision making [4].

The problem is visible at the Tahfidzul Qur’an Region of Zaid bin Tsabit (*Daerah Tahfidzul Qur’an Wilayah Zaid Bin Tsabit*), part of Pondok Pesantren Nurul Jadid in Probolinggo, Indonesia. Observation and an interview with the tahfidz mentors in June 2026 showed that submissions are written in a notebook, recapitulated at the end of each month, and then sent to parents one by one through WhatsApp. About 85 guardians receive a monthly report. The mentors estimated that recapitulation takes two to three minutes per student and sending takes one to two minutes per guardian, so that one reporting period consumes roughly four to seven hours, depending on network conditions and the number of data revisions. When data change after a report has been produced, the report must be recapitulated and sent again, and a lost notebook means a lost memorization history.

Several studies have developed web-based tahfidz monitoring systems that record memorization, present progress as tables or charts, or serve as management dashboards [5],[6],[7],[8],[9]. Among the five studies reviewed in Section II, none combines automatic recapitulation, progress visualisation, and automated delivery of the report to parents through a WhatsApp gateway in one workflow, and none was implemented with the Django framework. Such an integration matters because it turns recorded data into a continuous monitoring mechanism for mentors and parents instead of a monthly manual task. Django was selected because its Model–View–Template (MVT) architecture supports structured web information systems with good security and convenient database management [10], and its built-in authentication and object-relational mapper further simplify development.

This study therefore develops and evaluates a Django-based web application that integrates memorization recording, automatic report generation, progress charts, and automated report delivery to parents through WhatsApp for the tahfidz unit described above. The application was evaluated by ten tahfidz mentors through a functional test, the System Usability Scale (SUS), and a user experience questionnaire. The main contributions of this paper are as follows:

- An integrated web application, named SMHTAHFIDZ-ZBT, that connects recording of memorization submissions, period reports, progress charts, and WhatsApp delivery with a delivery log.
- A documented design, from requirements triangulated from observation, interviews, and literature to use case, activity, and class diagrams and a six-entity database schema, that other tahfidz institutions can examine and adapt.
- An initial evaluation by ten mentors that reports functional acceptance, usability, and user experience scores together with their limitations.

The remainder of this paper is organised as follows. Section II reviews related work, Section III describes the materials and methods, Section IV presents the results, Section V discusses them, and Section VI concludes the paper.

II. RELATED WORK

Previous work on digital tahfidz monitoring can be grouped into three themes. The first theme is web and mobile monitoring systems that replace paper records. Ridwan and Mutia [5] developed a web and mobile monitoring and evaluation application for a Qur'an school in Pidie District to replace paper sheets and spreadsheets; it presents memorization as tables, text, charts, and achievement reports, and the authors reported an average effectiveness of 96% from a user questionnaire. Alfitra and Tanti [7] designed a web-based monitoring and evaluation application for tahsin and tahfidz programmes that accelerates recapitulation and supports the ranking of the best memorisers, and Sholehatusun [8] built a Waterfall-based web system that helps mentors and foundation leaders monitor memorization and reduces recording errors. Marier and Dewi [9] added progress graphs and a target-achievement dashboard for Islamic boarding schools so that teachers and parents can follow memorization visually.

The second theme concerns governance and administration. Dewi Rahmawati et al. [6] strengthened the governance of a Qur'an house through a website, a memorization dashboard, and digital content, and mentioned automatic reports to guardians only as a plan. The third theme is the use of a WhatsApp gateway as a notification channel in school information systems. Imran et al. [3] combined a website with a WhatsApp gateway in a monitoring information system for an elementary school, which suggests that the channel is practical for institution-guardian communication, although that study does not address tahfidz.

Table 1 compares the five tahfidz studies with the present work. All five provide a web-based system and three present charts or dashboards. WhatsApp reporting appears only as a plan in [6], and automatic recapitulation and the Django framework are not reported in any of them. Apart from the 96% effectiveness score in [5], the studies report effectiveness mainly in qualitative terms, which makes a quantitative comparison with the present results difficult. This study differs from them by combining all five features in one application and by reporting standardised usability and experience measures from the target users.

Table 1. Comparison of this study with previous tahfidz monitoring studies, based on the features reported in each paper

Study	Web system	Dashboard / chart	WhatsApp reporting	Automatic recap	Django
Ridwan and Mutia [5]	Yes	Yes	No	No	No
Dewi Rahmawati et al. [6]	Yes	Yes	Planned	No	No
Alfitra and Tanti [7]	Yes	No	No	No	No
Sholehatusun [8]	Yes	No	No	No	No
Marier and Dewi [9]	Yes	Yes	No	No	No
This study	Yes	Yes	Yes	Yes	Yes

Note: "No" means the feature is not reported in the respective paper.

III. MATERIALS AND METHODS

A. Research Design and Study Setting

This study follows a development research approach whose product is a Django-based web application, and the System Development Life Cycle with the Waterfall model was used. Agile models suit projects whose requirements change often, and prototyping suits projects whose requirements cannot be defined clearly at the start [11], [12]. Here the requirements were obtained from observation and interviews before implementation and did not change during development, so the sequential Waterfall stages of requirements analysis, system design, implementation, and testing were judged more suitable [11]; the maintenance stage lies beyond the evaluation reported here. The study was conducted at the Tahfidzul Qur'an Region of Zaid bin Tsabit, Pondok Pesantren Nurul Jadid, Paiton, Probolinggo.

B. Data Collection and Requirements Validation

Four techniques were used. First, the recording, recapitulation, and reporting process was observed directly. Second, the tahfidz mentors were interviewed with nine questions based on the 5W+1H approach (what, who, when, where, why, and how). Third, a literature study covered tahfidz monitoring, web applications, Django, WhatsApp gateways, and usability testing. Fourth, questionnaires were used for the evaluation described in Section III-D. The requirements were validated by source and method triangulation: the findings from observation and interviews were compared with the literature, and a requirement was accepted when the three sources were consistent.

C. System Design and Implementation

The proposed system is a web application with two roles. The administrator has full access to student data, submissions, reports, users, WhatsApp settings, and the WhatsApp log, whereas the mentor (*pembina*) records and manages submissions, views progress, creates reports, and sends them to parents. The architecture and data flow are shown in Fig. 1. The application follows the MVT pattern of Django, stores data in SQLite, and uses Bootstrap for a responsive interface that can be opened on computers and smartphones.

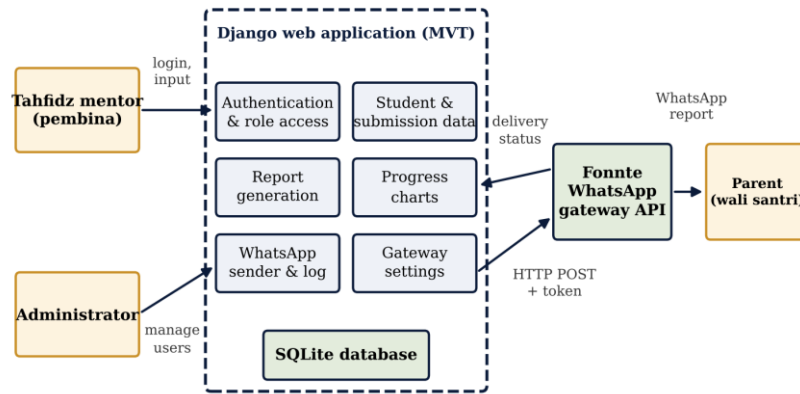


Fig. 1. Architecture and data flow of the proposed system.

The system was modelled with use case, activity, and class diagrams and an entity–relationship diagram (ERD). The database has six entities (Fig. 2). A student has many submissions and many reports, and a report can have many delivery log entries because a report that is sent again creates another entry. A submission stores the date, type (*ziyadah* or *muroja'ah*), juz category, surah, pages, quality, and mentor notes. A report stores the period, the number of submissions, the average daily submissions, the latest progress, the latest juz category, and notes. The log stores the target number, message, delivery status, sending time, and gateway response, and the settings entity stores the API token, gateway number, country code, and a simulation mode.

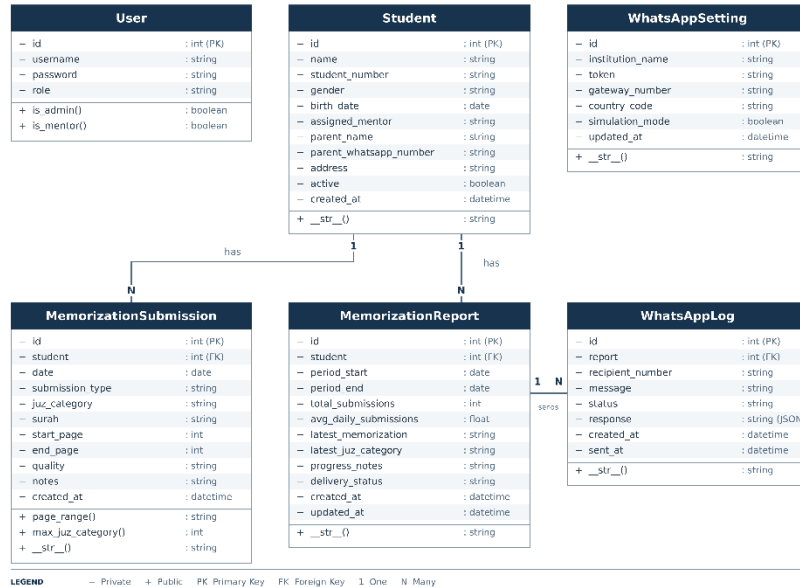


Fig. 2. Entity–relationship diagram of the database (attributes are summarised). 1 and N mark one-to-many relationships; User and WhatsAppSetting are standalone entities.

Report delivery uses the Fonnte WhatsApp gateway, a third-party service. When a mentor sends a report, the application posts an HTTP request to the Fonnte API with the target number, the message text, and an authentication

token. Fonnte forwards the message to the guardian's WhatsApp account and returns a delivery status that the application stores in the log. The message is previewed in the application before it is sent. Fonnte was chosen for its straightforward integration with Django, its documented API, and because it could be used at no cost in this study. Development used Python, Django, Visual Studio Code, and SQLite on a laptop (HP Notebook 240 G6, Intel Core i3, 8 GB RAM, 500 GB SSD) with a stable Internet connection.

D. Evaluation Design

Ten tahfidz mentors, who are the primary users of the application, were recruited by purposive sampling. Usability evaluations are commonly run with small samples of about five to twenty participants [13], and SUS scores have been reported to stabilise with about eight to twelve respondents [15]. Three instruments were used. (1) A functional test covered 12 features (login; add, edit, and delete student; record submission; generate report; report detail; progress chart; send WhatsApp report; WhatsApp log; WhatsApp settings; and logout), each rated from 1 (very poor) to 5 (very good). (2) The SUS [14] has ten statements rated on a five-point Likert scale. (3) A ten-item user experience questionnaire was adapted from the User Experience Questionnaire (UEQ) dimensions of attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty [16], [17] and extended with four items on ease of use, visual design, interaction, and satisfaction, each rated from 1 (very negative) to 7 (very positive). Because this adapted instrument does not contain the full UEQ item set, the UEQ benchmark [17] is not applicable and the scores are interpreted only against the scale midpoint of 4.

E. Evaluation Metrics

The feasibility percentage of the functional test is calculated with (1).

$$P = \frac{S}{S_{\max}} \times 100\% \quad (1)$$

where S is the sum of all ratings and $S_{\max} = n \times m \times 5$ is the maximum attainable score for n respondents and m features.

The SUS score of a respondent is calculated with (2).

$$\text{SUS} = 2.5 \times \sum_{i=1}^{10} s_i \quad (2)$$

where s_i is the contribution of item i , equal to the rating minus 1 for odd-numbered (positively worded) items and 5 minus the rating for even-numbered (negatively worded) items. Descriptive results are reported as the mean and the sample standard deviation in (3).

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}} \quad (3)$$

where x_i is the i -th observation, $\bar{x}$ is the mean, and n is the number of observations. The SUS bands used in this study are above 80.3 (excellent), 68 to 80.3 (good), and below 68 (needs improvement). No significance tests were applied because there were no comparison groups.

IV. RESULTS

A. Requirements

Observation showed that submissions were recorded only in a notebook that can be lost or damaged, that searching for a record took a long time, that recapitulation was manual and error-prone, and that parents could not see progress directly because the information stayed in the mentor's notebook; reports were sent manually through WhatsApp. In the interviews, the mentors named recording, charts for evaluation, and reporting to parents as the most important needs. They said that recording and reporting are done once a month at the regional office and expected reports to be sent through WhatsApp because most guardians use it. The literature indicated that web systems, dashboards, and charts support monitoring [5]–[9]. The three sources were consistent, so the requirements were accepted as valid. They comprise eight functional requirements (login, student management, submission management, report generation, WhatsApp report delivery, progress chart monitoring, access-right management, and logout) and five non-functional requirements (web-based, Django, SQLite, Fonnte integration, and browser access).

B. Implemented System

All designed modules were implemented and integrated in one application. It provides authentication with role-based access for administrators and mentors, student data management, submission recording, creation of a report for a selected student and period from the stored submissions, and a report detail page that shows a summary and a preview of the WhatsApp message before sending. A progress chart shows the daily number of submissions, the WhatsApp log shows the time, target number, status, and gateway response of each delivery, and the gateway can be configured from a settings page without changing the code. Django's administration site is used for data management. A report message contains a greeting, the reporting period, the number of submissions, the latest pages, the juz category, the average daily submissions, and the mentor's note.

Fig. 3 shows the dashboard, which is the landing page after login. It combines the number of active students, the total number of submissions, the number of reports created, and the WhatsApp delivery log with a chart of submission activity over the last 30 days, all computed from the database. The mentor therefore sees the overall condition of the memorization records without searching the notebook or recapitulating by hand.

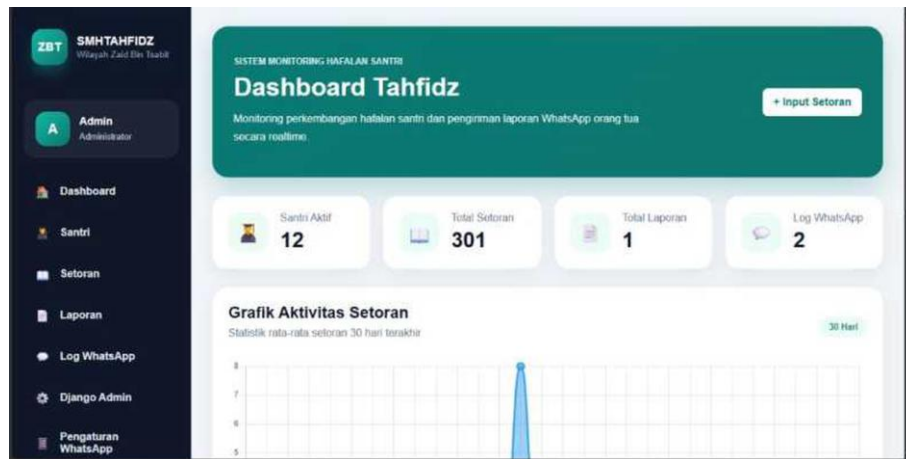


Fig. 3. Dashboard of the system, showing summary cards and the submission activity chart.

The student data page (Fig. 4) lists each student with the student number, assigned mentor, parent, parent WhatsApp number, and status, and offers search by name and add, edit, and delete actions. The parent number stored here is the target of the automated report.

The table lists the following students:

Nama	NIS	Ustadz	Orang Tua	Status	Aksi
Abdul Rohman	2026003	Ustadz Hanan	Orang Tua Abdul Rohman	Aktif	Edit Hapus
Ahmad Fauzi	2026001	Ustadzah Aisyah	Orang Tua Ahmad Fauzi	Aktif	Edit Hapus
Aisyah Putri	2026004	Ustadz Hanan	Orang Tua Aisyah Putri	Aktif	Edit Hapus
Fajar Ramadhan	2026007	Ustadz Hanan	Orang Tua Fajar Ramadhan	Aktif	Edit Hapus
Hasan Basri	2026010	Ustadz Fikri	Orang Tua Hasan Basri	Aktif	Edit Hapus

Fig. 4. Student data page (the WhatsApp number column is blurred).

Fig. 5 shows the memorization submission page. Each record contains the date, student, submission type (new memorization or review), juz category, memorized pages, and a quality rating, and can be edited or deleted. These records replace the notebook entries and are the source for the report, the chart, and the WhatsApp message, so they do not need to be entered again.

The table lists the following submission records:

Tanggal	Santri	Jenis	Kategori	Hafalan	Kualitas	Aksi
13/06/2026	Abdul Rohman	Ziyadah	1-5 Juz	Halaman 120-125	Baik	Edit Hapus
05/06/2026	Hasan Basri	Ziyadah	1-25 Juz	Halaman 28-33	Cukup	Edit Hapus
05/06/2026	Nabila Putri	Ziyadah	1-25 Juz	Halaman 68-73	Sangat Baik	Edit Hapus
04/06/2026	Nur Hidayah	Ziyadah	1-15 Juz	Halaman 61-64	Baik	Edit Hapus
04/06/2026	Nur Hidayah	Ziyadah	1-15 Juz	Halaman 64-67	Baik	Edit Hapus
03/06/2026	Iham Maulana	Murajaah	1-20 Juz	Halaman 4-9	Baik	Edit Hapus
02/06/2026	Abdul Rohman	Ziyadah	1-10 Juz	Halaman 27-31	Baik	Edit Hapus

Fig. 5. Memorization submission data page.

The report page (Fig. 6) lists the reports that have been created, with the student, period, total submissions, daily average, juz category, and delivery status. From here the mentor can open the report detail, edit the report, or delete it, and a report is sent to the parent through the WhatsApp gateway from its detail page.

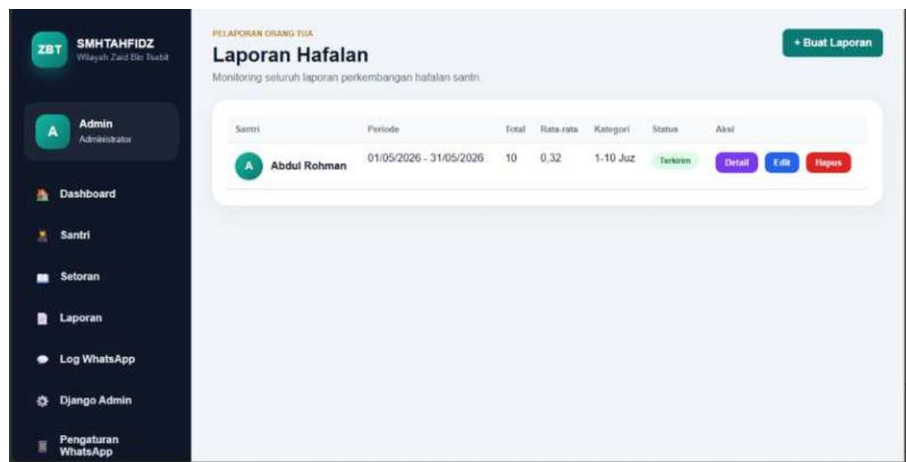


Fig. 6. Memorization report data page.

Every delivery is recorded in the WhatsApp log (Fig. 7), which stores the time, the delivery status, and the response returned by the gateway; in the figure, successful deliveries show the response “Message successfully added to queue”. This record lets the administrator check whether a report was sent and, when a delivery fails, read the gateway response to find the cause and send the report again.



Fig. 7. WhatsApp delivery log page.

C. Functional Test

Table 2 shows the functional test results. The total score was 536 out of 600, which equals a feasibility of 89.33%, and the mean over all 120 ratings was 4.47 (SD 0.50). Every rating was 4 or 5. Sending a report through WhatsApp was rated highest (mean 4.80), and the progress chart, the WhatsApp log, the WhatsApp settings, and logout were rated 4.30 each. No function failure was reported during testing.

Table 2. Functional test results from ten mentors (rating scale 1–5)

Feature	Total score (max 50)	Mean	SD
Login	44	4.40	0.52
Add student	46	4.60	0.52
Edit student	46	4.60	0.52
Delete student	46	4.60	0.52
Record submission	44	4.40	0.52
Generate report	45	4.50	0.53
Report detail	45	4.50	0.53
Progress chart	43	4.30	0.48
Send report via WhatsApp	48	4.80	0.42
WhatsApp log	43	4.30	0.48

Feature	Total score (max 50)	Mean	SD
WhatsApp settings	43	4.30	0.48
Logout	43	4.30	0.48
Overall (12 features)	536 / 600	4.47	0.50

Note: SD is the sample standard deviation across the ten respondents (overall: across all 120 ratings).

D. Usability and User Experience

Fig. 8(a) shows the SUS score of each respondent. The scores ranged from 62.5 to 87.5, and the mean was 73.0 (SD 7.1; 95% confidence interval 67.9 to 78.1), which falls in the good band. Seven respondents scored 70 or higher and three scored below 68 (67.5, 62.5, and 67.5). Among the items, the statements on expected frequent use, ease of use, and confidence received the highest mean rating (4.2 each), whereas the negatively worded statements about needing to learn many things first (item 10, mean 2.7) and about inconsistency (item 6, mean 2.5) were the least favourable.

The user experience questionnaire gave an overall mean of 5.57 on the seven-point scale, and all ten items were above the neutral midpoint of 4 (Fig. 8(b)). Ease of use and satisfaction were rated highest (5.80 each), perspicuity followed with 5.70, and visual design was rated lowest (5.20). The item standard deviations ranged from 0.42 to 0.79, and the respondent-level means ranged from 5.0 to 6.0.

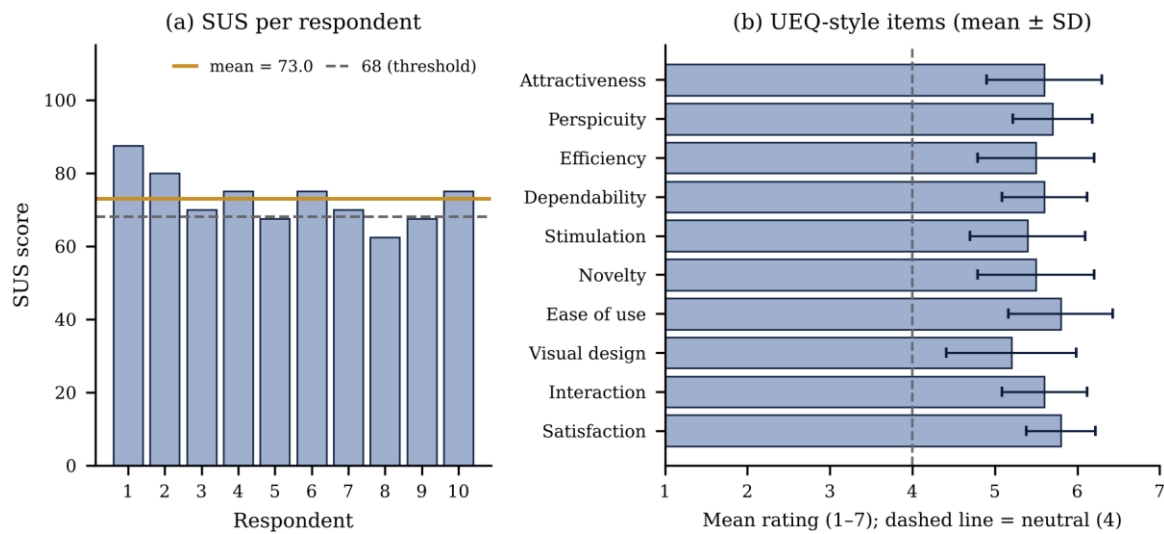


Fig. 8. Usability and user experience results from ten mentors: (a) SUS score per respondent, where the dashed line marks 68 and the solid line marks the mean; (b) mean $\pm$ standard deviation of the ten user experience items on a 1–7 scale.

V. DISCUSSION

The results indicate that the application meets the needs identified at the study site, as seen by its primary users. All functional ratings were 4 or 5, and sending a report through WhatsApp, which replaces one of the two manual steps described in Section I, received the highest rating. The functional test, however, is a user rating of each feature and not a scripted test with defined cases and defect counts, so the absence of reported failures should be read as perceived functional acceptance and not as proof of correctness.

The SUS mean of 73.0 lies slightly above the 71.4 reported for the adjective “good” and below the 85.5 reported for “excellent” in the adjective-rating study of Bangor et al. [18]. With ten respondents the confidence interval is wide, and its lower bound (67.9) touches the threshold of 68 used here, so the evidence supports “acceptable to good” usability and not a precise grade. The weakest SUS items (the effort needed to learn the system and perceived inconsistency) and the lowest-rated experience item (visual design, 5.20) suggest that interface consistency and onboarding for new mentors are plausible targets for refinement, but the present data cannot show which change would help most.

Compared with the studies in Section II, the present work adds an integrated delivery workflow and standardised instruments. The comparison with earlier effectiveness results is only indirect because the instruments differ; for example, Ridwan and Mutia [5] reported a 96% effectiveness score from their own questionnaire, which is not comparable with a SUS score. Practically, the application removes the need to retype recapitulated data for each guardian, builds each message from stored data, previews it before sending, and records the outcome of every delivery, which supports accountability when a message fails. The efficiency gain itself was not measured: the four to seven hours per reporting period is a mentor estimate of the manual process, and no time measurement of the new process was made.

Several limitations apply. First, the study covers one tahfidz unit and ten respondents, all of them mentors who use the system directly; guardians, who receive the reports, were not surveyed, so parent-side acceptance and understanding

of the messages remain untested. Second, the ratings are self-reported, there was no comparison group, and respondents who evaluate a system built for them may rate it favourably. Third, the user experience questionnaire was adapted and has not been validated, and no before-and-after time study was done. Fourth, delivery depends on Internet connectivity and on a third-party gateway whose availability, policies, and API limits are outside the control of the developers, and the application has no automatic retry. Fifth, SQLite suits a small deployment and would have to be replaced for larger use. Finally, the system transmits guardians' phone numbers and children's learning records through a third-party service, so consent and data-protection practices should be settled before wider deployment.

VI. CONCLUSION

This study developed a Django-based web application that integrates memorization recording, automatic report generation, progress charts, and WhatsApp report delivery with a delivery log for a tahfidz unit of an Islamic boarding school. Ten mentors gave a mean functional rating of 4.47 out of 5 (89.33%), a SUS score of 73.0, and a mean user experience rating of 5.57 out of 7, with visual design rated lowest at 5.20. These results indicate that the application works as intended and is acceptable to its primary users in this setting.

The evidence is limited to one site, ten mentors, and self-reported ratings, and efficiency and parent-side outcomes were not measured. Future work should add automatic retry of failed deliveries and the official WhatsApp Business API, move to a server-grade database with backup, provide parent accounts or a mobile version, evaluate the application with guardians and in other institutions, measure the time needed for reporting before and after adoption, and explore data analytics to support mentors in identifying students who need attention.

ACKNOWLEDGMENT

The authors thank the tahfidz mentors of the Tahfidzul Qur'an Region of Zaid bin Tsabit, Pondok Pesantren Nurul Jadid, for their participation in the interviews and evaluation.

AUTHOR CONTRIBUTIONS

Muhammad Hayat Aqil Syiraj: Conceptualization, Methodology, Software, Investigation, Formal analysis, Writing – original draft. Kamil Malik and Wali Ja'far Shudiq: Supervision, Validation, Writing – review & editing.

DATA AVAILABILITY

The questionnaire results supporting the findings of this study are reported in the tables and figures of this article. The application source code is not publicly available at the time of submission.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

- [1] E. I. Wahyuni, F. Muthmainnah, B. Permana, and T. Novalima, "Pengembangan sistem informasi manajemen sekolah berbasis web untuk meningkatkan efisiensi administrasi pendidikan," *Uranus: J. Ilm. Tek. Elektro, Sains dan Inform.*, vol. 3, no. 2, pp. 65–76, 2025.
- [2] R. A. Sunarjo, M. H. R. Chakim, S. Maulana, and G. Fitriani, "Management of educational institutions through information systems for enhanced efficiency and decision-making," *Int. Trans. Educ. Technol.*, vol. 3, no. 1, pp. 47–61, 2024.
- [3] A. Imran, M. S. N. Wahid, F. Baso, and A. Fadil, "Development monitoring information system based on website and WhatsApp gateway at SD Telkom Makassar," *J. Embedded Syst., Secur. Intell. Syst.*, vol. 5, no. 3, pp. 240–248, 2024, doi: 10.59562/jessi.v5i3.5193.
- [4] D. D. Hermawan and M. Andriansyah, "Visualisasi data pendidikan SD & SMP daerah Kabupaten Tangerang menggunakan aplikasi Microsoft Power Business Intelligence," *J. Comprehensive Sci.*, vol. 4, no. 2, 2025.
- [5] R. Ridwan and C. Mutia, "Pengembangan aplikasi web dan mobile pada monitoring dan evaluasi program tahfidz Quran," *Circuit: J. Ilm. Pendidik. Tek. Elektro*, vol. 6, no. 1, pp. 47–53, 2022, doi: 10.22373/crc.v6i1.11700.
- [6] A. Dewi Rahmawati et al., "Penguatan tata kelola Rumah Qur'an melalui pengembangan website, dashboard hafalan santri, dan konten digital," *J. Pengabdian Masyarakat Charitas*, vol. 5, no. 2, 2025.
- [7] A. Alfira and L. Tanti, "Sistem informasi monitoring dan evaluasi program tahsin dan tahfidz Al-Qur'an berbasis web," 2024. [Journal name, volume, and pages to be completed].
- [8] E. Sholehatur, "Sistem monitoring hafalan santri rumah tahfidz berbasis web," *Riau J. Tek. Inform.*, vol. 2, no. 3, pp. 67–71, 2023, doi: 10.61876/rjti.v2i3.2904.
- [9] M. Marier and F. Dewi, "Tahfidz Quran monitoring system in Islamic boarding schools," *J. Inform. dan Teknol. Inf.*, vol. 18, no. 1, pp. 1–11, 2021, doi: 10.31515/telematika.v18i1.3931.
- [10] F. Risya et al., "Perancangan sistem informasi manajemen invoice menggunakan generator framework Django-Python berbasis website pada PT. Lampuind Tekno Elektrik," *J. Sist. Inf. Univ. Suryadarma*, vol. 10, no. 1, pp. 207–220, 2023.
- [11] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York, NY, USA: McGraw-Hill, 2020.
- [12] I. Sommerville, *Software Engineering*, 10th ed. Harlow, U.K.: Pearson, 2016.

- [13] J. Nielsen and T. K. Landauer, "A mathematical model of the finding of usability problems," in *Proc. INTERCHI '93*, Amsterdam, The Netherlands, 1993, pp. 206–213.
- [14] J. Brooke, "SUS: A quick and dirty usability scale," in *Usability Evaluation in Industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, Eds. London, U.K.: Taylor & Francis, 1996, pp. 189–194.
- [15] T. S. Tullis and J. N. Stetson, "A comparison of questionnaires for assessing website usability," in *Proc. Usability Professionals' Assoc. Conf.*, Minneapolis, MN, USA, 2004.
- [16] B. Laugwitz, T. Held, and M. Schrepp, "Construction and evaluation of a user experience questionnaire," in *HCI and Usability for Education and Work (USAB 2008)*, A. Holzinger, Ed., Lecture Notes in Computer Science, vol. 5298. Berlin, Germany: Springer, 2008, pp. 63–76, doi: 10.1007/978-3-540-89350-9_6.
- [17] M. Schrepp, A. Hinderks, and J. Thomaschewski, "Construction of a benchmark for the User Experience Questionnaire (UEQ)," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 4, no. 4, pp. 40–44, 2017, doi: 10.9781/ijimai.2017.445.
- [18] A. Bangor, P. Kortum, and J. Miller, "Determining what individual SUS scores mean: Adding an adjective rating scale," *J. Usability Stud.*, vol. 4, no. 3, pp. 114–123, May 2009.