

Mushlihuun Application as an Android-Based E-Monitoring and Support System for Qur'an Memorization at SMK Muhammadiyah Kaje

Journal manuscript adapted from the original undergraduate thesis

Tri Jaya Prasetya

Program Studi Teknik Informatika, STMIK Widya Pratama Pekalongan, Indonesia

Adapted from the 2020 bachelor's thesis

Abstract

This study aimed to develop the Mushlihuun Application as an Android-based e-monitoring and learning-support system for Qur'an memorization at SMK Muhammadiyah Kaje. The main problem in the existing process was that memorization submissions were recorded in student-held notebooks, which limited teachers' ability to monitor progress, while some students experienced difficulties related to motivation and memorization methods. Data were collected through observation, interviews, questionnaires, and document study. The system was developed using the Waterfall model, consisting of requirements analysis and definition, system and software design, implementation and unit testing, integration and system testing, and operation and maintenance. The application was implemented with Flutter and Dart and used Firestore for data storage. Evaluation employed white-box testing, black-box testing, and User Acceptance Testing (UAT). In the initial needs questionnaire involving 93 students, 97.8% owned an Android smartphone, 69.9% reported difficulties related to memorization motivation or methods, and 98.9% agreed with the concept of an application that combined memorization records with supporting media. The UAT results showed scores of 75.70% for operation across different devices, 78.90% for ease of use, 73.98% for security, 81.72% for memorization monitoring, 77.42% for supporting memorization features, and 73.76% for interface attractiveness. These findings indicate that Mushlihuun fulfilled its primary functions as an e-monitoring and memorization-support system in the research context. However, the study did not experimentally measure improvement in learning outcomes or memorization achievement; further development is directed toward more complete data management and reporting features.

Keywords: e-monitoring; Qur'an memorization; Android; learning support; educational technology; Waterfall

1. Introduction

Digital transformation in education is not limited to content delivery; it also concerns the management of learning processes and the documentation of learner progress. Monitoring systems generally collect, analyze, and present data so that a process can be observed in a more structured manner [13]. Mobile technologies have also been used in educational contexts, including mobile learning and Qur'an-related learning applications [6,15,17].

SMK Muhammadiyah Kaje requires students to memorize Juz 30 as part of religious education assessment. According to the source thesis, memorization is organized by grade level, while

submission records are written in guidance notebooks carried by students. Memorization submissions are generally conducted once a week during Qur'an and Hadith lessons or at other available times. This arrangement means that teachers do not always have direct access to up-to-date memorization data and remain dependent on paper records [12].

The learning process also presents student-related challenges. In the initial needs questionnaire involving 93 students, 69.9% reported difficulties related to memorization motivation or methods. In addition, 54.8% indicated a concern regarding the loss or damage of paper memorization records, while 97.8% already owned an Android smartphone. A total of 98.9% agreed with the proposed application concept that combined memorization records with supporting motivational media. These findings established a practical basis for a mobile solution integrating monitoring and learning support.

The literature reviewed in the original thesis includes Android-based learning, monitoring, and religious applications. Hendra et al. [6] developed an Android-based Iqra learning application; Listyorini and Widodo [15] developed mobile learning; Nurdiansah and Ibrahim [17] developed an application for daily Islamic prayers and remembrance; while Jumri [13] and Laili [14] used monitoring systems to facilitate data storage and observation. These studies support the use of mobile technology as a learning-support medium, whereas Mushlihuun specifically integrates memorization monitoring and memorization support within one school-based system.

Accordingly, this study addressed two questions: (1) how can an e-monitoring system be designed and developed to record and monitor student memorization submissions; and (2) how can an Android application provide supporting media for memorization while meeting user needs in the context of SMK Muhammadiyah Kajen?

2. Methods

2.1 Data Collection

The study used primary and secondary data. Primary data were obtained through observation of memorization submission activities, interviews with teachers involved in memorization assessment and implementation, and questionnaires administered to students. Secondary data were obtained from school documents and relevant literature [22,23]. Interviews and observation were used to understand the existing process, while the questionnaire captured device availability, memorization difficulties, data-backup needs, and interest in the proposed application.

2.2 System Development

The system was developed using the Waterfall model, consisting of requirements analysis and definition, system and software design, implementation and unit testing, integration and system testing, and operation and maintenance [18,20,24]. During requirements analysis, teacher and student needs were identified from observation, interviews, and documents. During design, UML was used to model the system and a Screen Design Worksheet (LKT) was used to prepare the interface design [3,5].

Teacher requirements included access validation, class and student management, memorization submission processing, score calculation, and retrieval of data from Firestore. Student

requirements included login, profile management, memorization lists, scores, teacher notes, and memorization-support features. Data storage followed a NoSQL/Firestore approach [1,9].

2.3 Implementation Technologies

The application was developed for Android using Flutter and the Dart programming language [10]. Android was selected because most respondents already used Android smartphones, while Android version documentation was considered in relation to device compatibility [8,11]. Firestore was used as the cloud-based data store [9].

2.4 Testing

Testing consisted of white-box testing to examine program logic, black-box testing to verify functional inputs and outputs, and User Acceptance Testing to assess system suitability for intended users [7,21]. In the white-box example for the memorization-submission form, the cyclomatic complexity value was $V(G)=6$ and was categorized as a low-risk level within the framework used in the thesis [16]. Black-box tests covered authentication, data management, memorization submission, memorization lists, teacher notes, score calculation, recommended applications, and video playback.

UAT used a five-point scale: Strongly Agree (5), Agree (4), Neutral (3), Disagree (2), and Strongly Disagree (1). Scores were calculated as the mean response relative to the maximum possible score.

3. Results and Discussion

3.1 Needs Analysis

Table 1. Summary of initial needs

Initial need	Result
Questionnaire respondents	93 students
Owned an Android smartphone	91 students (97.8%)
Reported difficulty with memorization motivation/methods	65 students (69.9%)
Agreed with the supporting-app + data-backup concept	92 students (98.9%)
Concern related to loss/damage of memorization records	51 students (54.8%)

The needs analysis shows that smartphones were already available to almost all respondents. More than two-thirds reported difficulties related to memorization motivation or methods. In addition, 98.9% agreed with the idea of an application combining digital memorization records or backup with supporting media. Therefore, the Mushlihuun design was not intended simply to replace the notebook; it was designed to make memorization status and supporting resources more accessible.

3.2 System Design and Features

The Mushlihuun use case model contains two main actors: the teacher as data manager and the student as application user. The two central functions are e-monitoring and memorization support. Teachers can manage classes, students, submissions, and scores, while students can view their memorization lists, scores, and teacher notes [5].

On the student side, the supporting-media tab contains videos intended to provide motivation and Islamic knowledge, while the monitoring tab displays memorization lists, teacher notes, and submission scores. Recommendation features also provide listening, consultation, counter, and random-number methods. The application therefore combines process documentation and learning support within one mobile environment.

3.3 Implementation and Functional Testing

The implementation produced an Android application with teacher, student, and About menus. The system included teacher access validation, Google account sign-in, class and student management, memorization-submission storage, and student access to memorization data. The data structure was stored in Firestore [1,9].

The black-box tests showed that the outputs matched the expected results for the tested functions. The tests included no-internet conditions, login validation, class and student management, submission input, memorization lists, teacher notes, total scores, application recommendations, and video playback. These results support the thesis conclusion that the main functions operated as intended within the designed test scenarios.

3.4 User Acceptance Test

Table 2. Summary of UAT results

UAT dimension	Percentage
Operation across different devices	75.70%
Ease of use	78.90%
Security	73.98%
Memorization monitoring	81.72%
Supporting memorization features	77.42%
Interface attractiveness	73.76%

The highest score was obtained for memorization monitoring (81.72%), followed by supporting memorization features (77.42%) and ease of use (78.90%). Security received 73.98%, operation across different devices 75.70%, and interface attractiveness 73.76%. In the teacher UAT interview, the teacher indicated that the application could simplify submission recording and monitoring because the information was no longer dependent only on the student's notebook. The teacher also recommended adding reporting functionality as a future improvement.

These findings indicate positive acceptance of the system's primary functions, but the results should be interpreted according to the study design. UAT scores represent user perceptions of

suitability and usability; they do not constitute causal evidence that the application improves memorization achievement. The study also did not compare memorization outcomes before and after implementation through an experimental design.

3.5 Discussion

From an educational-technology perspective, the contribution of Mushlihuun lies in integrating three needs into one workflow: submission recording, progress monitoring, and learning support. The original problem of fragmented information between students' notebooks and teachers was translated into digital records accessible through a mobile application. This aligns with the broader role of monitoring systems in collecting and presenting information for process observation [13].

From a software-development perspective, the Waterfall model provided explicit stages from requirements analysis to maintenance [18,20,24]. UML supported the translation of user requirements into interactions between teachers, students, and the system [5]. White-box and black-box testing complemented UAT so that evaluation considered program logic, functional behavior, and user perception [7,21].

Compared with several studies reviewed in the thesis, Mushlihuun has a more specific focus on Juz 30 memorization monitoring and supporting media within a school environment.

Nevertheless, the study remains an early-stage development and evaluation. Future research should examine effects on memorization achievement, extend reporting functions, and evaluate sustained use over a longer period.

4. Conclusion

Mushlihuun was developed as an Android application integrating memorization e-monitoring with Qur'an memorization support at SMK Muhammadiyah Kajen. The initial needs analysis showed high Android availability and strong interest in digital recording and learning support. The implementation used Flutter, Dart, and Firestore, while evaluation employed white-box testing, black-box testing, and UAT. UAT scores ranged from 73.76% to 81.72% across the assessed dimensions, indicating positive acceptance of the system's main functions. The main contribution of the study is the development and early validation of an educational-technology solution for memorization monitoring and support. Future research should test learning outcomes more rigorously and add more complete reporting and data-management capabilities.

5. Limitations and Future Research

The study was conducted in one school and focused on application development and evaluation. The UAT design was not an experiment for measuring improvement in memorization achievement. The source thesis also recommended extending the system from monitoring into more complete memorization-submission management with downloadable reports, together with stronger NoSQL/Firestore management capabilities [9]. Future work could use a pretest-posttest or comparison-group design, add learning analytics, and provide reporting dashboards for teachers.

Manuscript note: all results, figures, features, numerical values, and references in this version are adapted from the 2020 source thesis. No claim of improved memorization achievement has been added beyond the evidence reported in the thesis.

References

- [1] Ardiansah, F., Nasir, M., & Hutrianto. (2016). Perangkat Lunak Pengajaran Basis Data NoSQL Berbasis Mobile. Palembang: Jurnal Ilmiah Universitas Binadarma.
- [2] Kadir, A., & Triwahyuni, T. C. (2013). Pengantar Teknologi Informasi Edisi Revisi. Yogyakarta: ANDI.
- [3] Gata, W., & Gata, G. (2013). Sukses Membangun Aplikasi Penjualan dengan Java. Jakarta: Elex Media Komputindo.
- [4] Hasyim, M. F. (2018). Aplikasi Pengaduan Masyarakat (E-Complain) Kabupaten Batang Dengan Menggunakan Web Service Berbasis Android. Pekalongan: Skripsi STMIK Widya Pratama.
- [5] Hendini, A. (2016). Pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak). Pontianak: Jurnal Khatulistiwa Informatika, Vol. IV, No. 2.
- [6] Hendra, K. M., Ramadhany, A., Artina, N., & Willy. (2013). Rancang Bangun Aplikasi Pembelajaran Iqra Berbasis Android. Palembang: STMIK MDP.
- [7] Hendradjaya, B. (2017). Konsep Dasar Pengujian Perangkat Lunak. Bandung: ITB Press.
- [8] Hermawan, S. (2011). Mudah Membuat Aplikasi Android. Jakarta: Elex Media Komputindo.
- [9] Firebase. Firebase Database documentation. <https://firebase.google.com/docs/database>. Diakses pada 16 September 2019.
- [10] Flutter. Flutter documentation. <https://flutter.dev>. Diakses pada 16 September 2019.
- [11] Android Open Source Project. Android build numbers documentation. <https://source.android.com/setup/start/build-numbers>. Diakses pada 12 November 2019.
- [12] SMK Muhammadiyah Kajen. Website sekolah. <https://www.smkmuhamka.sch.id>. Diakses pada 29 Maret 2019.
- [13] Jumri, J. P. (2013). Perancangan Sistem Monitorng Konsultasi Bimbingan Akademik Mahasiswa dengan Notifikasi Realtime Berbasis SMS Gateway. Pontianak: Universitas Tanjungpura.
- [14] Laili, F. Y. (2018). Sistem Monitoring dan Tracking Kerusakan Laboratorium Komputer STMIK Widya Pratama Pekalongan. Pekalongan: Skripsi STMIK Widya Pratama.
- [15] Listyorini, T., & Widodo, A. (2013). Perancangan Mobile Learning Mata Kuliah Sistem Operasi Berbasis Android. Kudus: Jurnal SIMETRIS, Vol. 3, No. 1.
- [16] Murphy, J. R., & Robinson, J. E. (2007). Design of a Research Platform for En Route Conflict Detection and Resolution. Northern Ireland: 7th AIAA Aviation Technology, Integration and Operations Conference (ATIO).
- [17] Nurdiansah, & Ibrahim, A. (2019). Perancangan Aplikasi Doa dan Wirid Harian Muslim Berbasis Android. Makasar: Prosiding Seminar Ilmiah Sistem Informasi dan Teknologi Informasi Vol. VIII, No. 2.
- [18] Pressman, R. S. (2012). Rekayasa Perangkat Lunak – Buku Satu, Pendekatan Praktisi (Edisi 7). Yogyakarta: ANDI.
- [19] Salbino, S. (2014). Buku Pintar Gadget Android Untuk Pemula. Jakarta: Kunci Komunikasi.

- [20]** Sasmito, G. W. (2017). Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal. Tegal: Jurnal Pengembangan IT (JPIT), Vol. 2, No. 1.
- [21]** Sidi Mustaqbal, M., Firdaus, R. F., & Rahmadi, H. (2015). Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus: Aplikasi Prediksi Kelulusan SNMPTN). Bandung: Jurnal Ilmiah Teknologi Informasi Terapan Vol. I, No. 3.
- [22]** Siswanto, V. A. (2012). Strategi dan Langkah-langkah Penelitian. Yogyakarta: Graha Ilmu.
- [23]** Sudaryono. (2015). Metodologi Riset di Bidang TI (Panduan Praktis, Teori dan Contoh Kasus). Yogyakarta: Andi Offset.
- [24]** Sommerville, I. (2011). Software Engineering 9th Edition. Addison-Wesley.