Prosiding Seminar Nasional Teknologi dan Sistem Informasi (SITASI) 2026
Surabaya, 3 Juni 2026

MOSIFY: PEMBERDAYAAN PENGURUS MASJID DALAM PENGELOLAAN KEBERSIHAN PERLENGKAPAN IBADAH BERBASIS TEKNOLOGI *MOBILE*

MOSIFY: EMPOWERING MOSQUE ADMINISTRATORS IN MANAGING WORSHIP EQUIPMENT CLEANLINESS THROUGH MOBILE TECHNOLOGY

Chiquita Angie Rahmadani'Ca[1], Brian Satria Hutama Susetyo[2], Satrio Alif Mahendra[3], Devano Varel Mareno[4], Agung Brastama Putra[5], Mohammad Al Hafidz[6]

E-mail: 24082010256@student.upnjatim.ac.id[1]
24082010246@student.upnjatim.ac.id[2]
24082010232@student.upnjatim.ac.id[3]
24082010264@student.upnjatim.ac.id[4]

[1]Sistem Informasi, Fakultas Ilmu Komputer, UPN "Veteran" Jawa Timur.

Abstrak - Pengelolaan kebersihan perlengkapan ibadah di masjid masih dilakukan secara manual sehingga menyulitkan proses penjadwalan, *monitoring*, dan pencatatan laundry perlengkapan ibadah. Penelitian ini bertujuan mengembangkan aplikasi Android bernama MOSIFY untuk membantu pengurus masjid dalam mengelola kebersihan perlengkapan ibadah secara digital. Metode penelitian menggunakan *Research and Development* (R&D) dengan model *Software Development Life Cycle* (SDLC) Waterfall yang meliputi tahap analisis kebutuhan, perancangan, implementasi, dan pengujian sistem. Aplikasi dikembangkan menggunakan Kotlin dan *Firebase Realtime Database*. Hasil pengujian menggunakan *black-box testing* menunjukkan seluruh fitur aplikasi berjalan dengan baik sesuai kebutuhan pengguna. MOSIFY mampu membantu pengelolaan jadwal kebersihan, *monitoring* perlengkapan, serta pencatatan laporan kebersihan secara lebih terstruktur dan efisien.

Kata kunci: *MOSIFY, manajemen masjid, kebersihan perlengkapan ibadah, Android, monitoring.*

Abstract - *The management of worship equipment cleanliness in mosques is still carried out manually, making scheduling, monitoring, and laundry recording processes less effective. This study aims to develop an Android-based application called MOSIFY to assist mosque administrators in managing worship equipment cleanliness digitally. The research method used was Research and Development (R&D) with the Software Development Life Cycle (SDLC) Waterfall model, which includes the stages of requirements analysis, system design, implementation, and testing. The application was developed using Kotlin and Firebase Realtime Database. The results of black-box testing showed that all application features functioned properly according to user needs. MOSIFY helps mosque administrators manage cleaning schedules, monitor equipment conditions, and record cleanliness reports in a more structured and efficient manner.*

Keywords: MOSIFY, mosque management, worship equipment cleanliness, Android, monitoring.

1. PENDAHULUAN

Masjid merupakan pusat peribadatan umat Islam yang membutuhkan standar kebersihan tinggi untuk menjaga kenyamanan jamaah. Salah satu fasilitas penting adalah perlengkapan ibadah seperti mukena dan sarung yang harus selalu bersih dan layak pakai. Namun, pengelolaan kebersihan perlengkapan ibadah di banyak masjid masih dilakukan secara manual, sehingga sering terjadi ketidakpastian jadwal pencucian, lemahnya koordinasi dengan jasa laundry, serta kurangnya pemantauan stok perlengkapan.

Perkembangan teknologi *mobile* membuka peluang untuk meningkatkan efisiensi pengelolaan fasilitas masjid. Penelitian sebelumnya telah membahas sistem manajemen kebersihan berbasis digital, tetapi belum banyak yang fokus pada pengelolaan kebersihan perlengkapan ibadah di masjid. Oleh karena itu, penelitian ini mengembangkan aplikasi *mobile* bernama MOSIFY sebagai solusi untuk membantu pengurus masjid dalam mengatur jadwal pencucian, memantau stok perlengkapan ibadah, dan mengawasi layanan kebersihan secara lebih terstruktur dan *real-time.* Aplikasi ini diharapkan dapat meningkatkan efisiensi pengelolaan serta menjaga standar higienitas perlengkapan ibadah secara berkelanjutan.

2. Metodologi

Penelitian ini merupakan penelitian *Research and Development* (R&D) yang bertujuan untuk menghasilkan sebuah produk berupa aplikasi *mobile* bernama MOSIFY (Mosque Information System For You). Aplikasi ini dirancang berbasis Android untuk membantu pengurus masjid dalam mengelola kebersihan perlengkapan ibadah secara lebih terstruktur dan terdigitalisasi. Pendekatan yang digunakan dalam pengembangan sistem adalah *Software Development Life Cycle* (SDLC) dengan model *Waterfall*. Model *Waterfall* dipilih karena memiliki alur pengembangan yang sistematis, bertahap, dan terstruktur sehingga memudahkan peneliti dalam mengontrol setiap proses pembangunan aplikasi [1]. Tahapan dalam model *Waterfall* yang diterapkan pada penelitian ini meliputi empat tahap utama, yaitu analisis kebutuhan,



perancangan sistem, implementasi, dan pengujian sistem, sebagaimana diilustrasikan pada Gambar 1.

Gambar 1. Tahapan Pengembangan Sistem Model *Waterfall*

2.1 Tahap Analisis Kebutuhan

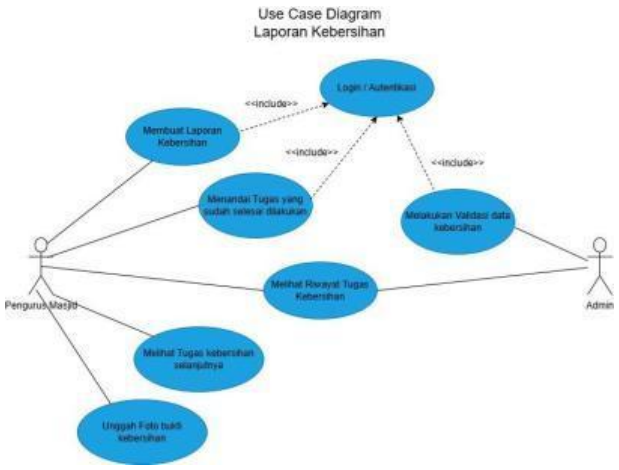
Analisis kebutuhan bertujuan untuk mengidentifikasi permasalahan dan kebutuhan pengguna dalam pengelolaan kebersihan perlengkapan ibadah di masjid. Pada tahap ini dilakukan observasi, wawancara, dan studi literatur. Observasi dilakukan di beberapa masjid di Surabaya dan sekitarnya untuk mengamati proses pengelolaan kebersihan perlengkapan ibadah. Wawancara dilakukan kepada marbot dan pengurus masjid untuk mengetahui prosedur operasional yang berjalan. Studi literatur dilakukan dengan menelaah penelitian terkait sistem informasi manajemen kebersihan dan aplikasi *mobile* berbasis Android [2][3].

Hasil observasi dan wawancara menunjukkan bahwa pengelolaan perlengkapan ibadah masih dilakukan secara manual tanpa sistem pencatatan digital. Pencucian mukena, sarung, dan sajadah dilakukan rutin seminggu sekali, sedangkan pengecekan kondisi perlengkapan dilakukan secara berkala oleh marbot. Pengelolaan pengiriman laundry dan komunikasi dengan pihak laundry juga masih dilakukan secara konvensional sehingga data sering tidak terdokumentasi dengan baik. Berdasarkan hasil analisis kebutuhan, sistem yang dikembangkan membutuhkan fitur jadwal kebersihan, *monitoring* kondisi perlengkapan, notifikasi otomatis, laporan kebersihan, pengelolaan data perlengkapan, dan pencatatan pengiriman laundry.

2.2 Tahap Perancangan Sistem

Perancangan sistem bertujuan mengubah hasil analisis kebutuhan menjadi desain teknis sebelum implementasi. Unified Modeling Language (UML) digunakan sebagai pemodelan visual sistem [4]. Diagram yang digunakan meliputi *use case diagram* untuk menggambarkan interaksi aktor dengan sistem dan *activity diagram* untuk menunjukkan alur proses aplikasi MOSIFY.

Use case diagram melibatkan dua aktor utama, yaitu pengurus masjid dan admin. Pengurus masjid dapat *login*, melihat tugas kebersihan, menandai tugas selesai, mengunggah foto bukti, membuat laporan kebersihan, dan melihat riwayat tugas. Admin bertugas memvalidasi data kebersihan dan memantau laporan dari pengurus masjid. Selain itu, dilakukan juga perancangan antarmuka pengguna (*user interface*) dan basis data untuk mengelola data pengguna, jadwal kebersihan, laporan, serta dokumentasi foto dalam aplikasi.



Gambar 2. *Use Case Diagram* Laporan Kebersihan

2.3 Tahap Implementasi

Tahap implementasi merupakan tahap penerjemahan desain sistem ke dalam kode program yang dapat dieksekusi. Aplikasi MOSIFY dikembangkan berbasis Android dengan menggunakan bahasa pemrograman Kotlin yang merupakan bahasa resmi yang direkomendasikan oleh Google untuk pengembangan aplikasi Android [5]. Untuk pengelolaan basis data lokal, aplikasi menggunakan MySQL yang terintegrasi secara *native* pada platform Android, sementara untuk kebutuhan sinkronisasi data secara *real-time* dan penyimpanan data berbasis *cloud*, sistem menggunakan *Firebase Realtime Database* yang disediakan oleh Google. Pengembangan antarmuka pengguna mengikuti panduan desain *Material Design* untuk memastikan pengalaman pengguna yang konsisten dan intuitif. Seluruh kode program dikembangkan menggunakan Android Studio sebagai *Integrated Development Environment* (IDE) utama.

2.4 Tahap Pengujian Sistem

Tahap pengujian sistem merupakan tahap verifikasi dan validasi untuk memastikan bahwa aplikasi yang telah dibangun berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan. Metode pengujian yang digunakan pada penelitian ini adalah *black-box testing.* Metode ini dipilih karena pengujian dilakukan dari perspektif pengguna dengan cara memasukkan *input* dan mengamati *output* yang dihasilkan tanpa memperhatikan struktur internal kode program [6]. Pengujian dilakukan terhadap seluruh fitur utama aplikasi MOSIFY, meliputi fitur autentikasi pengguna, pengelolaan data perlengkapan, jadwal kebersihan, monitoring kondisi perlengkapan, notifikasi otomatis, laporan kebersihan, serta pencatatan pengiriman ke laundry. Hasil pengujian kemudian didokumentasikan dalam bentuk tabel yang mencantumkan nama fitur, skenario pengujian, hasil yang diharapkan, dan keterangan hasil pengujian.

3. HASIL DAN PEMBAHASAN

3.1 Analisis Kebutuhan

Berdasarkan observasi dan wawancara yang dilakukan kepada pengurus masjid dan marbot di beberapa masjid di wilayah Surabaya, diperoleh gambaran yang komprehensif mengenai kondisi pengelolaan kebersihan perlengkapan ibadah yang ada saat ini. Hasil temuan menunjukkan bahwa pengelolaan perlengkapan ibadah seperti mukena, sajadah, sarung, karpet, dan Al-Qur'an masih sepenuhnya bergantung pada ingatan dan catatan manual marbot, sehingga rentan terhadap kelalaian dan tidak adanya standarisasi prosedur perawatan. Pencucian perlengkapan ibadah dilakukan seminggu sekali sebagai rutinitas tetap, namun tidak jarang frekuensi ini berubah tergantung pada intensitas penggunaan masjid, misalnya saat bulan Ramadan atau menjelang hari raya. Pengecekan rutin dilakukan terhadap mukena, sarung, rompi, dan Al-Qur'an untuk memastikan perlengkapan tersebut dalam kondisi layak pakai, akan tetapi pemeriksaan ini belum memiliki standar yang jelas dan tidak terdokumentasi.

Pengelolaan laundry yang dilakukan oleh marbot masih bersifat manual, di mana pengiriman dan penerimaan perlengkapan dicatat pada buku tulis atau bahkan tidak dicatat sama sekali. Komunikasi dengan pihak laundry dilakukan secara langsung atau melalui pesan singkat yang tidak terstruktur, sehingga data mengenai jumlah perlengkapan yang dikirim, status pencucian, dan jadwal pengambilan tidak dapat dipantau secara akurat. Kondisi ini memperbesar risiko kehilangan atau kerusakan perlengkapan ibadah karena tidak adanya rekam jejak digital yang dapat dijadikan acuan. Berdasarkan seluruh temuan tersebut, disimpulkan bahwa kebutuhan utama yang harus dipenuhi oleh aplikasi MOSIFY mencakup fitur pencatatan dan pengelolaan data perlengkapan ibadah, penjadwalan kebersihan, monitoring kondisi perlengkapan,

pencatatan transaksi laundry, notifikasi pengingat jadwal pencucian, serta pelaporan kebersihan yang dapat diakses secara digital oleh pengurus masjid.

3.3 Implementasi Sistem

Tahap implementasi merupakan proses menerjemahkan desain menjadi kode program yang siap dieksekusi. Aplikasi MOSIFY dikembangkan untuk perangkat Android menggunakan bahasa Kotlin melalui perangkat lunak Android Studio, serta menerapkan standar *Material Design* untuk tampilan antarmukanya. Sistem ini memadukan MySQL sebagai basis data lokal dan Firebase Realtime *Database* untuk kebutuhan penyimpanan *cloud* serta sinkronisasi data secara langsung. Hasil akhir dari pengkodean ini mewujudkan halaman utama aplikasi, seperti *login*, *dashboard*, jadwal kebersihan, monitoring, dan laporan.

3.3.1 Halaman *login* dan *Dashboard*

Halaman *login* merupakan tampilan awal aplikasi MOSIFY untuk proses autentikasi pengguna menggunakan *username* dan password yang diverifikasi melalui Firebase Authentication. Setelah berhasil masuk, pengguna akan diarahkan ke halaman *dashboard*. Halaman *dashboard* ini berfungsi sebagai pusat informasi dengan tata letak berbasis *card* yang terstruktur, menampilkan ringkasan kondisi perlengkapan ibadah, jumlah barang yang harus segera dibersihkan, jadwal kebersihan terdekat, serta notifikasi sistem terbaru.



Gambar 3. Tampilan *login*

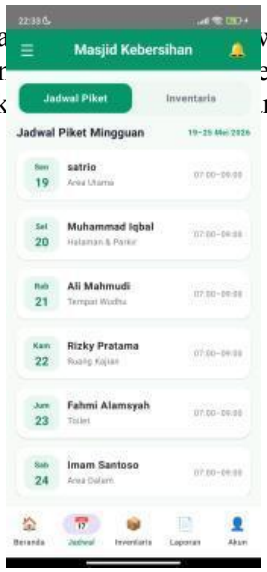


Gambar 4. Halaman Dashboard Mosify

3.3.2 Halaman Jadwal Kebersihan

Halaman jadwal kebersihan merupakan salah satu fitur inti dari aplikasi MOSIFY yang dirancang untuk membantu pengurus masjid dalam merencanakan dan mengelola agenda pembersihan perlengkapan ibadah secara sistematis. Pada halaman ini, pengurus dapat melihat daftar jadwal kebersihan yang telah dibuat sebelumnya dalam format tabel yang mencakup informasi nama perlengkapan atau area yang akan dibersihkan, periode pembersihan (harian, mingguan, atau bulanan), dan nama petugas yang bertanggung jawab. Pengurus dapat menambahkan jadwal baru dengan menekan tombol tambah yang tersedia, kemudian mengisi formulir yang muncul dengan informasi yang dibutuhkan. Selain itu, tersedia pula opsi untuk

mengedit jadwal yang sudah ada ma wal yang tidak lagi diperlukan. Sistem akan secara otomatis mengirimkan r epada pengurus sesuai dengan periode yang telah ditetapkan, sehingga tidak n yang terlewatkan.

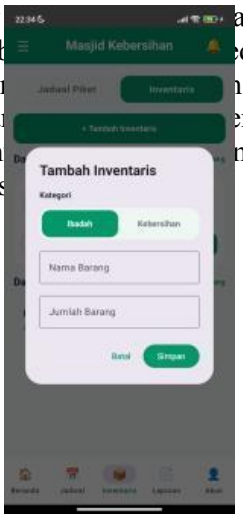


Gambar 5. Halaman Jadwal Kebersihan Piket

3.3.3 Halaman Monitoring Perlengkapan

Halaman Monitoring Perlengkapan berfungsi sebagai pusat pengelolaan dan pendataan inventaris perl Halaman ini menyajikan daftar a terstruktur, mencakup nam status kondisi terkini seperti b dang dicuci. Data yang dip *-time* membantu pengurus me n stok secara akurat tanpa p sehingga memudahkan penga rkait jadwal pencucian ata ng baru. Sistem juga men mulir untuk menambahkan baru yang langsung terintegras



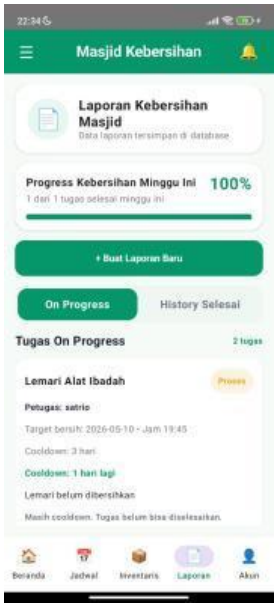


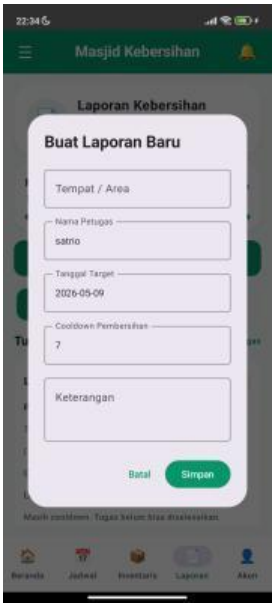
Gambar 6. Data Perlengkapan

Gambar 7. Menambahkan Data Perlengkapan

3.3.4 Halaman Laporan Kebersihan

Pada Halaman Laporan Kebersihan berfungsi sebagai dokumentasi digital dan bentuk pertanggungjawaban aktivitas kebersihan oleh petugas. Melalui fitur ini, pengguna dapat melihat riwayat laporan beserta statusnya, serta membuat laporan baru dengan mengisi detail kegiatan dan mengunggah foto sebagai bukti otentik. Seluruh laporan yang dikirim akan tersimpan di basis data secara *real-time* agar dapat diarsipkan, divalidasi, dan dipantau langsung oleh admin.





Gambar 8. Progress Laporan Kebersihan

Gambar 9. Membuat Laporan Baru

3.4 Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode *black-box testing* yang berfokus pada verifikasi fungsionalitas setiap fitur aplikasi dari sudut pandang pengguna tanpa memperhatikan struktur internal kode program [6]. Pengujian dilakukan secara menyeluruh terhadap 16 skenario pengujian yang mencakup seluruh fitur utama aplikasi MOSIFY. Setiap skenario pengujian dirancang untuk mencakup kondisi normal (*valid input*) maupun kondisi tidak normal (*invalid input*) guna memastikan aplikasi mampu menangani berbagai situasi penggunaan dengan benar. Hasil pengujian secara lengkap disajikan pada Tabel 1 berikut.

Tabel 1. Hasil *Black-box Testing* Aplikasi MOSIFY

No.	Fitur	Skenario Pengujian	Hasil yang Diharapkan	Keterangan
1.	*login* Pengurus	Memasukkan *username* dan password yang valid	Masuk ke halaman dashboard utama	Berhasil

2.	*login* Pengurus	Memasukkan *username* dan password yang tidak valid	Muncul pesan kesalahan dan tetap di halaman *login*	Berhasil
3.	Dashboard Utama	Mengakses halaman dashboard setelah *login*	Menampilkan ringkasan status perlengkapan dan notifikasi terbaru	Berhasil
4.	Tambah Data Perlengkapan	Mengisi formulir data perlengkapan baru (nama, jumlah, kondisi)	Data perlengkapan tersimpan dan muncul pada daftar perlengkapan	Berhasil
5.	Edit Data Perlengkapan	Memperbarui data perlengkapan yang sudah ada	Data berhasil diperbarui pada daftar perlengkapan	Berhasil
6.	Hapus Data Perlengkapan	Menghapus data perlengkapan dari daftar	Data berhasil dihapus dari sistem	Berhasil
7.	Tambah Jadwal Kebersihan	Mengisi nama area tugas, periode, dan petugas yang bertanggung jawab	Jadwal berhasil ditambahkan dan muncul pada halaman jadwal	Berhasil
8.	Hapus Jadwal Kebersihan	Menghapus jadwal kebersihan yang ada	Jadwal berhasil dihapus dari sistem	Berhasil
9.	Monitoring Perlengkapan	Mengakses halaman monitoring dan melihat status perlengkapan	Menampilkan daftar perlengkapan beserta status kondisi terkini	Berhasil
10.	Notifikasi Otomatis	Sistem mendeteksi perlengkapan yang sudah melewati jadwal pencucian	Notifikasi muncul sebagai pengingat kepada pengurus	Berhasil
11.	Tambah Laporan Kebersihan	Mengisi formulir laporan kebersihan dengan tanggal dan periode	Laporan berhasil disimpan pada halaman laporan kebersihan	Berhasil
12.	Lihat Laporan Kebersihan	Mengakses halaman laporan dan melihat riwayat laporan	Menampilkan daftar laporan kebersihan yang tersedia beserta detailnya	Berhasil
13.	Kirim ke Laundry	Membuat catatan pengiriman perlengkapan ke laundry	Catatan pengiriman berhasil tersimpan dan status perlengkapan berubah	Berhasil
14.	*Logout*	Menekan tombol *logout* dari aplikasi	Pengguna berhasil keluar dan kembali ke halaman *login*	Berhasil

Berdasarkan hasil *black-box testing* pada Tabel 1, seluruh fitur aplikasi MOSIFY berjalan dengan baik sesuai kebutuhan pengguna. Dari 14 skenario pengujian, semua menghasilkan *output* yang sesuai sehingga aplikasi dinyatakan layak digunakan oleh pengurus masjid. Fitur *login* berhasil memvalidasi autentikasi pengguna, fitur pengelolaan data perlengkapan mampu melakukan tambah, ubah, dan hapus data dengan baik, sedangkan fitur jadwal kebersihan dan monitoring perlengkapan dapat berjalan sesuai fungsinya. Selain itu, fitur notifikasi otomatis berhasil mengirim pengingat sesuai jadwal, dan fitur laporan kebersihan mampu menyimpan serta menampilkan riwayat laporan dengan baik.

Hasil penelitian ini sejalan dengan penelitian sebelumnya mengenai sistem pengelolaan kebersihan berbasis website yang mampu meningkatkan pengelolaan laporan kebersihan [8]. Perbedaannya, MOSIFY dikembangkan berbasis Android sehingga lebih fleksibel digunakan melalui *smartphone*. MOSIFY juga memiliki fitur monitoring perlengkapan ibadah dan notifikasi otomatis yang belum banyak ditemukan pada sistem serupa. Penerapan MOSIFY turut

mendukung peningkatan akuntabilitas dan transparansi pengurus masjid dalam pengelolaan fasilitas ibadah [7][9].

4. KESIMPULAN DAN SARAN

Penelitian ini berhasil merancang aplikasi Android bernama MOSIFY (Mosque Information System For You) sebagai solusi digital untuk pengelolaan kebersihan perlengkapan ibadah di masjid. Aplikasi dikembangkan menggunakan metode *Research and Development* (R&D) dengan model *Waterfall* melalui tahap analisis kebutuhan, perancangan, implementasi, dan pengujian. Hasil *black-box testing* terhadap 16 skenario menunjukkan seluruh fitur aplikasi berjalan dengan baik sehingga MOSIFY layak digunakan oleh pengurus masjid.

MOSIFY membantu pengurus masjid dalam menjadwalkan dan memantau kebersihan perlengkapan ibadah secara digital, meningkatkan pencatatan kondisi perlengkapan, menyediakan notifikasi otomatis, serta mendukung dokumentasi laporan kebersihan yang lebih terstruktur. Pengembangan selanjutnya dapat difokuskan pada peningkatan notifikasi *real-time* berbasis *Firebase Cloud Messaging* (FCM), penggunaan basis data *cloud* yang lebih skalabel, penambahan fitur monitoring berbasis IoT, serta pengembangan versi web atau panel administrasi agar pemantauan dapat dilakukan lebih fleksibel melalui berbagai perangkat.

5. DAFTAR RUJUKAN

[1] Android Developers, "Kotlin for Android," Google LLC, 2023. [Online]. Available: https://developer.android.com/kotlin. [Accessed: 20 Feb. 2025].

[2] A. Andriani, E. Elfaradayanti, R. Fadhillah, and M. A. Budiman, "Implementation of ISAK 35 to Increase the Accountability of Mosque Financial Report," in *Proc. 3rd Annual Management, Business and Economics Conference (AMBEC 2021)*, 2023, pp. 156–164.

[3] D. Febiharsa, I. M. Sudana, and N. Hudallah, "Uji Fungsionalitas (*Black-box Testing*) Sistem Informasi Lembaga Sertifikasi Profesi (SILSP) Batik dengan AppPerfect Web Test dan Uji Pengguna," *Joined Journal (Journal of Informatics Education)*, vol. 1, no. 2, p. 117, 2018.

[4] R. Kurniawan, "Rekayasa Perangkat Lunak Monitoring dan Pelaporan Kebersihan dan Pertamanan Kota Palembang dengan Teknologi *Mobile* Cross Platform," *Jurnal Nasional Ilmu Komputer*, vol. 1, no. 2, pp. 85–92, 2020.

[5] A. Latifah, A. Mulyani, and T. A. Wahidah, "Rancang Bangun Sistem Pengelolaan Kebersihan di Lingkungan Institut Teknologi Garut Berbasis Website," *Jurnal Algoritma*, vol. 19, no. 2, pp. 747–758, 2022.

[6] E. Nuryani and M. Y. Romdoni, "Aplikasi Monitoring Kebersihan Sarana dan Prasarana Kampus Berbasis Web Menggunakan Teknik Responsif pada Android," *JSiI (Jurnal Sistem Informasi)*, vol. 7, no. 2, pp. 89–96, 2020.

[7] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 8th ed. New York: McGraw-Hill Education, 2014.

[8] M. T. Prihandoyo, "Unified Modeling Language (UML) Model untuk Pengembangan Sistem Informasi Akademik Berbasis Web," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 3, no. 1, pp. 126–129, 2018.

[9] S. Priyanto, H. Suharno, and Haryono, "Value Engineering dan Strategi Bisnis pada PT Angkasa Pura II," *Jurnal Manajemen Transportasi & Logistik (JMTranslog)*, vol. 3, no. 2, pp. 245–265, Jul. 2016.

[10] I. K. A. H. Putra, D. Pramana, and N. L. P. Srinadi, "Sistem Manajemen Arsip Menggunakan *Framework* Laravel dan Vue.Js," *Jurnal Sistem dan Informatika*, vol. 13, no. 2, pp. 97–104, 2019.

[11] S. Rahayu and Andriani, "Analisis Penerapan Akuntabilitas dan Transparansi bagi Pengurus Masjid di Indonesia," *Jurnal Proaksi*, vol. 11, no. 1, pp. 126–142, 2024.

[12] A. F. Sallaby and I. Kanedi, "Perancangan Sistem Informasi Jadwal Dokter Menggunakan *Framework* Codeigniter," *Jurnal Media Infotama*, vol. 16, no. 1, pp. 48–53, 2020.