

Pengembangan Aplikasi Mobile Pengelola Data Sarana Berbasis Android Studio Dengan Fitur Crud Menggunakan SQLite

Delfi Kurnia

¹Fakultas Sains dan Teknologi, Program Studi Sistem Informasi, Universitas Labuhanbatu, Labuhanbatu, Indonesia

Email: delfikurnia08@icloud.com

Abstrak—Penelitian ini bertujuan untuk mengembangkan aplikasi mobile berbasis Android Studio yang berfungsi sebagai sistem pengelolaan data sarana. Aplikasi ini dilengkapi dengan fitur CRUD (Create, Read, Update, Delete) menggunakan SQLite sebagai basis data lokal. Permasalahan utama yang diangkat adalah kurangnya sistem informasi portabel dan efisien dalam pencatatan serta pengelolaan data sarana pada instansi tertentu. Pengembangan aplikasi dilakukan melalui metode prototipe dengan tahapan analisis kebutuhan, desain antarmuka, implementasi, dan pengujian. Hasil pengujian menunjukkan bahwa aplikasi ini mampu membantu pengguna dalam mengelola data secara efisien, dengan performa yang responsif dan antarmuka yang user-friendly. Sistem ini juga dapat digunakan secara offline sehingga meningkatkan kepraktisan pengguna dalam mengakses data.

Kata Kunci: Android Studio, CRUD, SQLite, Mobile App, Manajemen Sarana

Abstract—This study aims to develop a mobile application based on Android Studio that functions as a facility data management system. The application is equipped with CRUD (Create, Read, Update, Delete) features using SQLite as a local database. The main problem addressed is the lack of portable and efficient information systems in recording and managing facility data in certain institutions. The application development is carried out through a prototyping method consisting of needs analysis, UI design, implementation, and testing. The test results show that the application helps users manage data efficiently, with responsive performance and a user-friendly interface. The system can also be used offline, thus increasing user convenience in accessing data.

Keywords: Android Studio, CRUD, SQLite, Mobile App, Facility Management

1. PENDAHULUAN

Pengelolaan data sarana memiliki peran penting dalam mendukung operasional berbagai institusi seperti sekolah, kantor desa, lembaga pemerintahan, maupun organisasi swasta. Data sarana meliputi informasi terkait peralatan, infrastruktur, dan aset fisik lainnya yang digunakan untuk menunjang layanan atau aktivitas utama lembaga. Pengelolaan yang tidak efisien akan berdampak pada penurunan kualitas layanan, pemborosan anggaran pemeliharaan, dan hilangnya jejak histori penggunaan sarana. Di banyak kasus, proses pencatatan dan pelaporan sarana masih dilakukan secara manual melalui formulir kertas atau spreadsheet yang disimpan secara lokal dalam komputer. Pendekatan ini memiliki banyak keterbatasan, terutama dari sisi aksesibilitas, keamanan data, dan efektivitas operasional.

Salah satu permasalahan utama dari sistem manual adalah keterlambatan dalam pencatatan dan pelaporan, khususnya saat pengguna harus berpindah lokasi untuk mengakses komputer atau menginput data dari catatan lapangan. Selain itu, penyimpanan data secara terpisah atau tersebar di beberapa komputer menyebabkan kesulitan dalam konsolidasi data serta berisiko kehilangan data akibat kerusakan perangkat atau kesalahan pengguna. Hal ini juga menyulitkan lembaga dalam melakukan audit aset atau pengambilan keputusan berbasis data. Oleh karena itu, dibutuhkan suatu solusi yang bersifat portabel, efisien, dan memiliki antarmuka yang mudah digunakan oleh petugas lapangan.

Teknologi mobile menawarkan alternatif yang sangat potensial untuk menjawab kebutuhan tersebut. Perangkat Android, yang kini telah merata digunakan di berbagai lapisan masyarakat, dapat menjadi platform yang ideal untuk mengembangkan sistem informasi portabel. Dengan memanfaatkan perangkat mobile, proses pencatatan data dapat dilakukan langsung di lapangan tanpa harus bergantung pada koneksi jaringan atau perangkat komputer tetap. Selain itu, tampilan aplikasi mobile yang intuitif memungkinkan pengguna dari latar belakang non-teknis sekalipun dapat menggunakan sistem secara efektif.

Android Studio merupakan Integrated Development Environment (IDE) resmi dari Google yang menyediakan berbagai alat bantu dan pustaka untuk membangun aplikasi Android. Android Studio mendukung bahasa pemrograman Java dan Kotlin, serta menyediakan sistem build berbasis Gradle yang fleksibel. Salah satu keunggulan Android Studio adalah kemampuannya untuk berintegrasi dengan berbagai sistem penyimpanan data lokal maupun cloud. Untuk aplikasi yang tidak memerlukan koneksi internet, penggunaan SQLite menjadi pilihan yang efisien. SQLite merupakan sistem manajemen basis data relasional (RDBMS) ringan dan portabel yang disematkan langsung dalam perangkat Android, sehingga tidak memerlukan server eksternal untuk menyimpan data.

Penggunaan SQLite dalam aplikasi Android telah banyak diterapkan dalam berbagai bidang. Beberapa penelitian sebelumnya menunjukkan bahwa integrasi Android–SQLite sangat cocok untuk aplikasi berskala kecil hingga menengah, yang membutuhkan penyimpanan lokal dan akses data yang cepat. Yuce Yuliani | Sistem [1]

rancangan aplikasi belajar bahasa inggris berbasis mobile menggunakan database sqlite, Aries Saifudin et al. [2] Pengembangan Aplikasi Mobile pada Studi Kasus Penjualan Sayur dan Buah Berbasis Android, Rifky Azmil et al. [3] perancangan aplikasi todolist berbasis android menggunakan flutter sdk dan database sqlite, Ian Setyo Aji et al. [4] pengembangan aplikasi perangkat bergerak sistem informasi dan manajemen bank sampah kota batu berbasis android (studi kasus: bank sampah teratai putih), Firman Ali Eka Atmojo et al. [5] perbandingan kinerja local database pada aplikasi mobile dengan flutter.

Meskipun integrasi Android–SQLite telah banyak digunakan, namun penelitian yang secara khusus fokus pada pengembangan aplikasi pengelolaan data sarana masih sangat terbatas. Sarana dan prasarana adalah komponen penting dalam mendukung fungsi utama lembaga, namun sering kali pengelolaannya kurang diperhatikan dalam pengembangan sistem informasi mobile. Hal ini menciptakan gap penelitian, di mana belum banyak solusi yang dirancang khusus untuk mempermudah pencatatan dan pelaporan sarana melalui aplikasi Android yang bekerja secara offline.

Melalui penelitian ini, penulis mencoba untuk mengisi kekosongan tersebut dengan merancang dan mengembangkan aplikasi mobile pengelolaan data sarana berbasis Android Studio yang dilengkapi dengan fitur CRUD (Create, Read, Update, Delete) menggunakan SQLite sebagai sistem manajemen basis data lokal. Sistem ini dirancang untuk memudahkan pengguna dalam menambahkan, menampilkan, memperbarui, dan menghapus data sarana secara praktis melalui perangkat Android. Dengan kemampuan offline yang dimiliki SQLite, aplikasi ini juga dapat digunakan dalam kondisi tanpa koneksi internet, yang sangat berguna untuk wilayah-wilayah dengan infrastruktur jaringan yang terbatas.

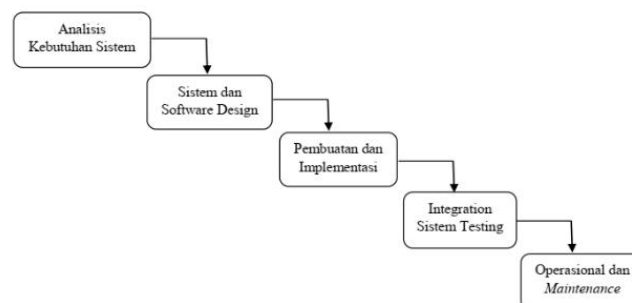
Penelitian ini tidak hanya berfokus pada aspek teknis pengembangan aplikasi, tetapi juga pada pengujian performa dan fungsionalitas sistem untuk memastikan bahwa semua fitur CRUD berjalan dengan baik. Evaluasi dilakukan dengan pendekatan black-box testing guna memverifikasi output sistem berdasarkan input yang diberikan tanpa memperhatikan kode internal. Diharapkan, hasil penelitian ini dapat memberikan kontribusi nyata dalam menyediakan solusi digital sederhana namun efektif untuk pengelolaan data sarana di berbagai institusi. Tujuan dari penelitian ini mengembangkan aplikasi mobile pengelola data sarana berbasis Android Studio dengan implementasikan fitur CRUD dengan penyimpanan lokal menggunakan SQLite. Melakukan pengujian terhadap fungsionalitas sistem secara menyeluruh. Memberikan alternatif solusi praktis yang dapat diadopsi oleh institusi dengan keterbatasan akses jaringan atau sumber daya TI.

Dengan pendekatan ini, aplikasi yang dikembangkan diharapkan tidak hanya menjadi produk teknis, tetapi juga solusi praktis yang dapat meningkatkan efisiensi kerja di lapangan, mempermudah pelaporan data sarana, dan mendukung pengambilan keputusan berbasis informasi yang akurat.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini menggunakan metode pengembangan perangkat lunak Waterfall, yang merupakan model klasik dalam rekayasa perangkat lunak. Metode ini dipilih karena memiliki tahapan yang terstruktur dan sistematis, sehingga sesuai untuk pengembangan aplikasi dengan ruang lingkup yang sudah jelas. Model Waterfall terdiri dari beberapa tahapan berurutan, yaitu: analisis kebutuhan, desain sistem, implementasi, pengujian, dan pemeliharaan. [6]



Gambar 2.1. Use Case Diagram

a. Analisis Kebutuhan

Tahap awal ini dilakukan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional dari aplikasi yang akan dikembangkan. Kebutuhan fungsional meliputi fitur utama seperti menambah, menampilkan, mengubah, dan menghapus data sarana (CRUD), sementara kebutuhan non-fungsional mencakup antarmuka yang sederhana, penggunaan tanpa koneksi internet, dan kecepatan akses data. Analisis dilakukan dengan pendekatan observasional terhadap proses manual sebelumnya serta studi literatur untuk mendukung perumusan spesifikasi sistem.

b. Desain Sistem

Setelah kebutuhan dianalisis, tahap berikutnya adalah perancangan sistem. Perancangan mencakup desain antarmuka pengguna menggunakan komponen Android Studio (seperti TextView, EditText, Button, RecyclerView), perancangan struktur database menggunakan SQLite, serta perancangan alur navigasi aplikasi. Desain sistem bertujuan untuk memberikan gambaran teknis mengenai cara kerja aplikasi, baik dari sisi tampilan maupun struktur penyimpanan data.

c. Implementasi

Pada tahap implementasi, rancangan sistem diterjemahkan ke dalam kode program menggunakan bahasa Java di Android Studio. Komponen utama yang dikembangkan antara lain DatabaseHelper untuk pengelolaan data di SQLite, RecyclerView untuk menampilkan daftar sarana, serta aktivitas seperti AddSaranaActivity dan EditSaranaActivity untuk menangani input dan pengeditan data. Seluruh fungsionalitas CRUD diimplementasikan secara lokal, sehingga aplikasi dapat digunakan tanpa bergantung pada server eksternal.

d. Pengujian

Pengujian dilakukan dengan metode black-box testing untuk memastikan bahwa setiap fitur berfungsi sesuai spesifikasi tanpa melihat kode sumber. Pengujian mencakup seluruh fungsi CRUD dan dilakukan langsung di perangkat Android emulator dan fisik. Fokus pengujian adalah memastikan validitas input, penyimpanan ke database lokal, serta performa antarmuka pengguna. Setiap kesalahan yang ditemukan diperbaiki sebelum aplikasi dinyatakan stabil.

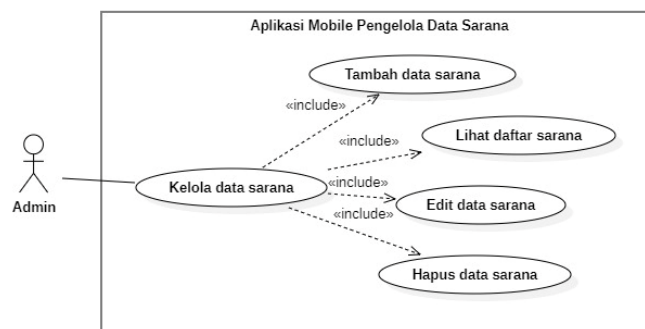
e. Pemeliharaan

Tahap pemeliharaan dilakukan untuk memastikan aplikasi tetap berjalan dengan baik pasca implementasi. Meskipun versi awal telah mencakup fitur dasar, evaluasi dan penyempurnaan terus dilakukan. Rencana pengembangan lebih lanjut meliputi penambahan fitur pencarian data, penyaringan data berdasarkan kategori, unggah gambar sarana, sistem autentikasi pengguna, dan fitur pencadangan data. Dengan pemeliharaan yang berkelanjutan, aplikasi diharapkan dapat menyesuaikan diri dengan kebutuhan pengguna yang terus berkembang.

Secara keseluruhan, tahapan Waterfall dalam penelitian ini mendukung proses pengembangan aplikasi secara berurutan dan terstruktur, mulai dari analisis kebutuhan hingga evaluasi akhir, sehingga hasil akhir dapat memenuhi tujuan penelitian secara efektif.

3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil implementasi dan pembahasan mendalam mengenai aplikasi Pengelola Data Sarana, yang dikembangkan menggunakan Android Studio berbasis Java dan menggunakan SQLite sebagai basis data lokal. Pembahasan mencakup detail fungsionalitas utama aplikasi, tampilan antarmuka pengguna, fitur-fitur CRUD yang telah diimplementasikan, serta rencana pengembangan lebih lanjut. Fokus utama adalah bagaimana aplikasi ini dapat mempermudah pengelolaan data sarana (seperti inventaris peralatan, fasilitas, dll.) secara efisien di perangkat Android. Use case diagram menunjukkan berbagai tindakan atau fungsionalitas yang dapat dilakukan oleh aktor terhadap sistem yang dirancang. Berikut Gambar 3.1 menampilkan Use Case Diagram:



Gambar 3.1. Use Case Diagram

3.1 Implementasi Aplikasi

Aplikasi Data Sarana berhasil dikembangkan menggunakan Android Studio dengan bahasa pemrograman Java dan memanfaatkan database SQLite sebagai media penyimpanan data secara lokal di perangkat. Aplikasi ini dirancang untuk memudahkan pengguna dalam mengelola data sarana yang mencakup nama barang, jenis barang, lokasi, dan kondisi. Fitur utama yang diterapkan meliputi CRUD (Create, Read, Update, Delete), yang memungkinkan pengguna untuk menambahkan data baru, menampilkan daftar data yang tersimpan, mengedit data yang sudah ada, serta menghapus data yang tidak diperlukan. Seluruh fitur tersebut telah diuji dan berjalan dengan baik, menunjukkan bahwa aplikasi mampu menjalankan fungsi dasarnya secara efisien dan stabil.

Aplikasi Pengelola Data Sarana terdiri dari satu komponen utama yaitu aplikasi mobile Android yang dibangun menggunakan Java di Android Studio. Data disimpan secara lokal menggunakan SQLite, tanpa ketergantungan pada koneksi internet atau backend eksternal. Pendekatan ini dipilih untuk mendukung penggunaan di area dengan keterbatasan jaringan serta memberikan akses cepat terhadap data. Struktur proyek di Android Studio disusun secara modular. Beberapa komponen utama meliputi:

- a. Activity: seperti MainActivity, AddSaranaActivity, dan EditSaranaActivity.
- b. Model: Kelas Sarana yang merepresentasikan entitas data sarana.
- c. Database Helper: Kelas DatabaseHelper yang menangani koneksi dan operasi database SQLite seperti insert, update, delete, dan select.
- d. Adapter: SaranaAdapter untuk menampilkan daftar sarana dalam RecyclerView.
- e. Fungsi CRUD diimplementasikan secara penuh:
- f. Create: Menambahkan data sarana melalui form input di AddSaranaActivity.
- g. Read: Menampilkan data dalam bentuk daftar di MainActivity menggunakan RecyclerView.
- h. Update: Mengedit data sarana yang dipilih dengan membuka EditSaranaActivity.
- i. Delete: Menghapus data dengan tombol khusus di tiap item daftar.

3.2 Implementasi Fitur CRUD

Salah satu tujuan utama dalam pengembangan aplikasi ini adalah menerapkan fungsi CRUD (Create, Read, Update, Delete) yang menjadi pilar utama dalam sistem informasi berbasis data. Fitur-fitur ini memungkinkan pengguna melakukan manipulasi data secara penuh terhadap entitas sarana yang tersimpan dalam basis data lokal.

a. Create (Menambah Data Sarana)

Fitur ini diakses melalui tombol “Tambah Data” di halaman utama. Ketika pengguna mengisi semua kolom (nama barang, jenis, lokasi, kondisi) lalu menekan tombol “Simpan”, aplikasi akan melakukan validasi sederhana terhadap input (tidak boleh kosong). Setelah itu, data akan dimasukkan ke dalam basis data SQLite menggunakan metode insert(). Jika berhasil, pengguna akan diarahkan kembali ke halaman utama, dan data terbaru langsung muncul dalam daftar.

Kode insert contoh:

```
java
CopyEdit
ContentValues values = new ContentValues();
values.put("nama_barang", nama);
values.put("jenis_barang", jenis);
values.put("lokasi", lokasi);
values.put("kondisi", kondisi);
db.insert("sarana", null, values);
```

b. Read (Menampilkan Data Sarana)

Semua data sarana yang tersimpan dalam database ditampilkan dalam RecyclerView menggunakan SaranaAdapter. Tampilan dirancang agar efisien dan rapi, serta menampilkan data secara real-time. Fitur ini juga mendukung pembaruan tampilan otomatis setiap kali pengguna menambah atau mengedit data.

```
java
CopyEdit
Cursor cursor = db.rawQuery("SELECT * FROM sarana", null);
```

c. Update (Mengedit Data Sarana)

Saat pengguna menekan tombol “Edit” pada salah satu item data, maka FormSaranaActivity akan terbuka dengan data yang sudah diisi secara otomatis. Setelah pengguna memperbarui isian dan menekan tombol “Simpan”, maka data lama akan di-*replace* dengan data baru melalui fungsi update() SQLite.

```
java
CopyEdit
db.update("sarana", values, "id=?", new String[]{String.valueOf(id)});
```

d. Delete (Menghapus Data Sarana)

Tiap entri memiliki tombol “Hapus”. Saat ditekan, sistem akan menampilkan dialog konfirmasi sebelum data benar-benar dihapus. Jika pengguna menyetujui, maka data akan dihapus dengan metode delete().

```
java
CopyEdit
db.delete("sarana", "id=?", new String[]{String.valueOf(id)});
```

Struktur tabel database ditampilkan dalam tabel berikut:

Tabel 3.1 Struktur tabel sarana

Nama Field	Type Data	Keterangan
id_sarana	Integer PRIMARY KEY	ID unik tiap data
nama_barang	String	Nama barang
jenis_barang	String	Jenis atau kategori barang
lokasi	String	Lokasi atau tempat penyimpanan
kondisi	String	Keterangan kondisi fisik barang

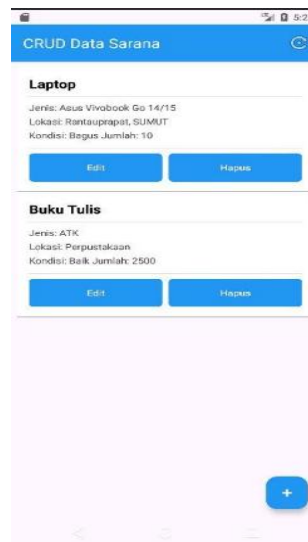
Secara keseluruhan, implementasi fitur CRUD berjalan dengan lancar dan berhasil memenuhi seluruh kebutuhan yang ditetapkan dalam tahap analisis. Struktur kode modular memungkinkan setiap fitur dapat dikembangkan lebih lanjut secara mandiri tanpa mengganggu komponen lain.

3.3 Implementasi UI (User Interface)

Implementasi antarmuka pengguna (UI) dalam aplikasi Pengelola Data Sarana dirancang menggunakan pendekatan material design yang sederhana, intuitif, dan mudah digunakan oleh pengguna dari berbagai latar belakang. Seluruh elemen UI dibangun menggunakan komponen bawaan Android Studio seperti TextView, EditText, Button, RecyclerView, dan FloatingActionButton. Desain antarmuka mengikuti prinsip konsistensi dan keterbacaan, agar pengguna dapat memahami fungsi aplikasi dengan cepat tanpa perlu panduan khusus.

a. Halaman Utama (MainActivity)

Halaman utama menampilkan daftar data sarana dalam bentuk daftar vertikal yang diimplementasikan menggunakan RecyclerView. Setiap item dalam daftar menampilkan informasi seperti Nama Barang, Jenis Barang, Lokasi, Kondisi. Selain itu, setiap item memiliki tombol aksi untuk Edit dan Hapus. Di pojok kanan bawah, terdapat tombol FloatingActionButton (FAB) dengan ikon tambah (+) yang berfungsi untuk membuka halaman tambah data sarana baru (AddSaranaActivity).



Gambar 3.2 Halaman Dashboard

b. Form Tambah Data Sarana

Tampilan form tambah data (activity_tambah.xml) memungkinkan pengguna menginput data baru dengan beberapa EditText untuk mengisi seperti Nama Barang, Jenis Barang, Lokasi, Kondisi, Tersedia tombol Simpan di bagian bawah yang akan menyimpan data ke database SQLite melalui fungsi insertSarana().

c. Form Edit Data Sarana

Halaman edit (activity_edit.xml) memiliki tampilan yang serupa dengan halaman tambah, namun seluruh EditText sudah terisi otomatis berdasarkan data sarana yang dipilih. Pengguna cukup mengubah data yang diperlukan, kemudian menekan tombol Simpan Perubahan untuk memicu fungsi updateSarana().

d. Layout Item Sarana

Setiap item yang ditampilkan dalam daftar menggunakan item_sarana.xml, yang berisi elemen-elemen seperti

TextView untuk menampilkan nama barang, jenis, lokasi, dan kondisi Dua tombol Button kecil (atau ImageButton) untuk Edit dan Hapus. Desainnya dibuat kompak dan responsif agar nyaman dilihat di berbagai ukuran layar.

e. Navigasi Antar Halaman

Navigasi antar halaman dilakukan menggunakan Intent dari Android. Misalnya, ketika pengguna menekan tombol tambah, maka Intent akan digunakan untuk berpindah dari MainActivity ke AddSaranaActivity. Demikian juga ketika memilih opsi edit, aplikasi akan mengirim ID sarana melalui Intent Extra ke EditSaranaActivity.

3.4 Pengujian menggunakan BlackBox

Pengujian pada aplikasi Pengelola Data Sarana dilakukan menggunakan metode Black Box Testing, yaitu metode pengujian yang berfokus pada fungsionalitas sistem tanpa melihat struktur internal kode program. Tujuan utama dari pengujian ini adalah untuk memastikan bahwa seluruh fitur yang tersedia dapat berjalan sesuai dengan kebutuhan pengguna. Pengujian dilakukan pada setiap fungsi utama, seperti menambahkan data sarana, menampilkan data, mengedit data, dan menghapus data. Selain itu, skenario pengujian juga mencakup validasi input, seperti mengosongkan kolom wajib, serta mengevaluasi tampilan data setelah aplikasi ditutup dan dibuka kembali (persistensi data). Semua pengujian dilakukan langsung di perangkat Android maupun emulator.

Tabel 3.2. Tabel hasil pengujian

No	Fitur	Hasil yang Diharapkan	Hasil Uji
1	Tambah Data	Data tersimpan dan tampil di daftar	Sesuai
2	Tambah (Kosong)	Muncul pesan error input wajib	Sesuai
3	Tampil Data	Data sarana tampil di halaman utama	Sesuai
4	Edit Data	Data berubah sesuai input baru	Sesuai
5	Edit (Kosong)	Muncul validasi error input kosong	Sesuai
6	Hapus Data	Data hilang dari daftar	Sesuai
7	Navigasi Form	Halaman tambah/edit terbuka	Sesuai
8	Buka Ulang Aplikasi	Data tetap tersimpan (persisten)	Sesuai

4. KESIMPULAN

Aplikasi Pengelola Data Sarana berbasis Android yang dikembangkan menggunakan metode Waterfall berhasil memenuhi tujuan penelitian, yaitu menyediakan solusi digital untuk pencatatan dan pengelolaan data sarana secara efektif. Dengan memanfaatkan Android Studio, bahasa pemrograman Java, dan SQLite sebagai penyimpanan data lokal, aplikasi ini mampu menjalankan fungsi CRUD (Create, Read, Update, Delete) secara optimal tanpa memerlukan koneksi internet. Seluruh tahapan dalam model Waterfall mulai dari analisis kebutuhan, perancangan antarmuka, implementasi, hingga pengujian black-box telah dilakukan secara sistematis dan menghasilkan aplikasi yang stabil serta mudah digunakan. Hasil pengujian menunjukkan bahwa semua fitur berjalan sesuai harapan, sementara desain antarmuka yang sederhana mendukung kemudahan penggunaan bagi admin. Aplikasi ini juga memiliki potensi pengembangan lebih lanjut, seperti penambahan fitur pencarian, login, dan backup data. Dengan demikian, aplikasi ini dapat dijadikan sebagai alat bantu praktis dan efisien dalam mengelola data sarana di berbagai institusi pendidikan atau organisasi.

UCAPAN TERIMAKASIH

Terima kasih disampaikan kepada pihak-pihak yang telah mendukung terlaksananya penelitian ini.

REFERENCES

- [1] Y. Yuliani, "RANCANGAN APLIKASI BELAJAR BAHASA INGGRIS BERBASIS MOBILE MENGGUNAKAN DATABASE SQLite," *JORAPI J. Res. Publ. Innov.*, vol. 2, no. 1, pp. 146–150, 2024, [Online]. Available: <https://jurnal.portalpublikasi.id/index.php/JORAPI/index>
- [2] A. Saifudin, A. Wahid, I. U. Aranda, S. Amruallah, and J. Simanungkalit, "OKTAL : Jurnal Ilmu Komputer dan Science Pengembangan Aplikasi Mobile pada Studi Kasus Penjualan Sayur dan Buah Berbasis Android," vol. 3, no. 6, pp. 1576–1582, 2024, [Online]. Available: <https://journal.mediapublikasi.id/index.php/oktal>
- [3] A. Rifky and Rahmawati, "Perancangan Aplikasi Todolist Berbasis Android menggunakan Flutter SDK dan Database SQLite," *Semin. Nas. Ris. dan Teknol. (SEMNAS RISTEK)*, pp. 567–572, 2021.

- [4] I. S. Aji, A. P. Kharisma, and M. A. Akbar, "Pengembangan Aplikasi Perangkat Bergerak Sistem Informasi dan Manajemen Bank Sampah Kota Batu berbasis Android (Studi Kasus: Bank Sampah Teratai Putih)," *J. Pengemb. Teknol. ...*, vol. 7, no. 2, pp. 515–524, 2023, [Online]. Available: <https://jptiik.multi.web.id/index.php/j-ptiik/article/view/12234>
- [5] F. Ali, E. Atmojo, M. Widiartha, and S. Kuta, "Perbandingan Kinerja Local Database Pada Aplikasi Mobile Dengan Flutter," *J. Nas. Teknol. Inf. dan Apl.*, vol. 1, no. 1, pp. 501–508, 2022.
- [6] B. Fachri, C. Rizal, and Supiyandi, "Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Merdeka Belajar Kampus Merdeka Berbasis Web," *J. Komput. Teknol. Inf. dan Sist. Inf.*, vol. 2, no. 3, pp. 591–597, 2024, doi: 10.62712/juktisi.v2i3.147.