



PENGEMBANGAN APLIKASI PENDAFTARAN ANGGOTA PERPUSTAKAAN BERBASIS ANDROID MENGGUNAKAN MODEL WATERFALL PADA DINAS PERPUSTAKAAN KOTA SAMARINDA

Muhammad Ramadhani Saputra¹, Abdul Rahim², Sayekti Harits Suryawan³

^{1,2,3} Teknik Informatika, Universitas Muhammadiyah Kalimantan Timur
Samarinda, Kalimantan Timur, Indonesia 75124
2011102441087@umkt.ac.id, ar622@umkt.ac.id, shs500@umkt.ac.id

Abstract

Technology, especially smartphones, has become essential to daily life, providing quick access to various services and information. However, the Library and Archive Service of Samarinda City still uses manual member registration, which could be more efficient and time-consuming. This research addresses this issue by developing an Android-based library membership registration application using Flutter, Laravel, and MySQL. The Waterfall method and UML diagrams are used to structure the development process and model the system. The resulting application allows prospective members to register online, streamlines the registration process, and reduces the administrative burden on staff. A book search feature is also included, enabling members to check availability remotely. This saves time and reduces unnecessary visits to the library, improving user experience and overall library efficiency. Survey results show a high level of satisfaction, with an average percentage of 83%, indicating that most respondents agree with implementing this system.

Keywords: Android Application, Flutter, Laravel, MySQL, Waterfall

Abstrak

Teknologi, terutama *smartphone*, telah menjadi bagian penting dalam kehidupan sehari-hari, memberikan akses cepat ke berbagai layanan dan informasi. Namun, Dinas Perpustakaan dan Arsip Kota Samarinda masih menggunakan pendaftaran anggota secara manual, yang tidak efisien dan memakan waktu. Penelitian ini bertujuan mengatasi masalah tersebut dengan mengembangkan aplikasi pendaftaran anggota perpustakaan berbasis *Android* menggunakan *Flutter*, *Laravel*, dan *MySQL*. Metode *Waterfall* dan diagram *UML* digunakan untuk menstrukturkan proses pengembangan dan memodelkan sistem. Aplikasi yang dihasilkan memungkinkan calon anggota mendaftar *online*, memperlancar proses pendaftaran, dan mengurangi beban administratif staf. Fitur pencarian buku juga disertakan, memungkinkan anggota memeriksa ketersediaan dari jarak jauh. Hal ini menghemat waktu dan mengurangi kunjungan yang tidak perlu ke perpustakaan, meningkatkan pengalaman pengguna dan efisiensi perpustakaan secara keseluruhan. Hasil kuesioner menunjukkan tingkat kepuasan yang tinggi, dengan nilai persentase rata-rata 83%, menandakan sebagian besar responden setuju dengan implementasi sistem ini.

Kata kunci: Aplikasi *Android*, *Flutter*, *Laravel*, *MySQL*, *Waterfall*

1. PENDAHULUAN

Di era modern ini, teknologi telah menjadi elemen krusial dalam keseharian masyarakat, memberikan pengaruh besar pada berbagai bidang, termasuk dunia pendidikan dan layanan untuk publik. Pesatnya perkembangan teknologi, khususnya telepon pintar, telah mengubah cara kita berinteraksi, menyelesaikan pekerjaan, dan memperoleh informasi. Meluasnya penggunaan telepon pintar di kalangan masyarakat memungkinkan akses cepat dan mudah terhadap beragam layanan dan informasi dari mana saja [1]. Di Indonesia sendiri, tren penggunaan telepon pintar terus menunjukkan peningkatan yang signifikan. Data

menunjukkan bahwa pada tahun 2015, kurang dari sepertiga penduduk Indonesia memiliki telepon pintar. Namun, empat tahun kemudian, angka ini melonjak menjadi lebih dari 60%, dan diprediksi akan mencapai hampir 90% dalam beberapa tahun ke depan [2].

Walaupun penggunaan teknologi semakin marak, beberapa lembaga pemerintah belum sepenuhnya memanfaatkan solusi digital dalam pelayanannya. Ambil contoh Dinas Perpustakaan dan Kearsipan Kota Samarinda. Meski sudah menerapkan sistem berbasis web dengan basis data *MySQL* untuk pengelolaan internal data anggota dan koleksi buku,

proses-proses yang melibatkan pengunjung masih dilakukan secara tradisional. Pendaftaran anggota baru masih mengharuskan calon anggota untuk datang langsung, mengisi formulir kertas, dan menyerahkan dokumen fisik seperti salinan KTP. Proses ini sering kali memaksa calon anggota untuk berkunjung lebih dari sekali. Selain itu, untuk menelusuri ketersediaan buku, anggota perpustakaan masih harus hadir secara fisik di perpustakaan. Hal ini berpotensi membuang waktu dan tenaga jika ternyata buku yang dicari sedang tidak tersedia.

Untuk mengatasi permasalahan tersebut, dikembangkan aplikasi pendaftaran anggota perpustakaan berbasis *Android* menggunakan *framework Flutter* untuk *frontend* dan *Laravel* untuk *backend*, dengan *database MySQL* sebagai basis data. *Flutter* adalah *framework* open-source buatan Google untuk pengembangan aplikasi mobile menggunakan bahasa *Dart*. *Flutter* memungkinkan pembuatan aplikasi yang kompatibel dengan *Android* dan *iOS* [3]. *Laravel* adalah kerangka kerja PHP yang populer untuk pengembangan situs web. Dengan fitur-fitur terintegrasi, *Laravel* mengurangi kompleksitas penulisan kode PHP, meningkatkan efisiensi pengelolaan kode, dan mempercepat pembuatan situs web [4]. *MySQL* adalah RDBMS gratis berlisensi GPL yang populer karena kinerja *query* data yang unggul. Penggunaannya yang bebas biaya membuatnya pilihan terjangkau, meskipun harus mematuhi ketentuan lisensi GPL yang melarang konversi ke produk tertutup atau komersial [5].

Pengembangan aplikasi ini akan menggunakan pendekatan model *Waterfall*, yang membagi proses pengembangan perangkat lunak menjadi tahapan-tahapan berurutan: analisis kebutuhan, desain, implementasi dan pengujian unit, integrasi dan pengujian sistem, serta operasi dan pemeliharaan [6]. Pendekatan ini memberikan kejelasan dan struktur yang baik dalam setiap tahap pengembangan. Tahap pertama, Analisis Kebutuhan (*Requirements Analysis*), melibatkan konsultasi dengan pengguna untuk menentukan kebutuhan sistem, batasan, dan tujuan. Informasi ini kemudian dijabarkan secara rinci dan berfungsi sebagai spesifikasi sistem [7]. Tahap kedua, Pengembangan sistem dan perangkat lunak melibatkan perencanaan aplikasi, termasuk pengaturan struktur data, penentuan arsitektur, perancangan antarmuka, dan prosedur pengkodean. Kebutuhan yang telah dianalisis disusun menjadi rencana yang dapat dijalankan. Selama proses ini, hasil desain didokumentasikan untuk referensi di masa depan [8]. Ketiga *Implementation and unit testing* adalah tahapan yang secara nyata dilalui dalam pengembangan suatu sistem. Tahapan ini mengoptimalkan penggunaan komputer. Setelah proses pengkodean selesai, sistem akan diuji untuk menemukan kesalahan dan kemudian diperbaiki [9]. Tahap keempat, *Integration and system testing*, bertujuan memverifikasi kecocokan sistem dengan tujuan yang ditetapkan melalui *blackbox testing*. Metode ini menguji aplikasi secara fungsional tanpa memperhatikan detail internal. Pengujian ini mengidentifikasi kesalahan,

fungsi yang tidak sesuai, atau fitur yang terlewat, serta mengukur tingkat kepuasan pengguna [10]. Tahap kelima, *Operation and maintenance*, Mengikutsertakan pengguna dalam uji langsung aplikasi, dilanjutkan dengan evaluasi potensi kelemahan. Jika ditemukan kelemahan, langkah pemeliharaan akan dijalankan untuk memperbaikinya [11].

Implementasi model *Waterfall* dan *framework Flutter* dalam pengembangan sistem telah dilakukan dalam beberapa penelitian terdahulu. Nur Fadilah, Ali Ikhwan, dan Muhamad Alda melakukan penelitian berjudul 'Pengembangan Sistem Informasi Perpustakaan Pada Dinas Perpustakaan dan Kearsipan Kota Medan Berbasis *Android*'. Penelitian ini berhasil memanfaatkan *Flutter* sebagai alat utama untuk mengoptimalkan proses pendaftaran, peminjaman, dan perpanjangan buku. Dengan menerapkan metode *Research and Development* melalui pendekatan *Waterfall*, tim peneliti mengintegrasikan aplikasi dengan *Firebase*, sehingga menghasilkan pengelolaan data yang lebih efisien [12]. Penelitian ini memiliki keunggulan dibandingkan penelitian sebelumnya, yaitu dengan menggunakan *Laravel* sebagai *backend*. *Laravel* menawarkan tingkat keamanan yang lebih tinggi dan skalabilitas yang lebih baik dibandingkan *Firebase* yang digunakan pada penelitian sebelumnya.

Artikel ini akan membahas penerapan aplikasi pendaftaran anggota perpustakaan berbasis *Android* menggunakan *framework Flutter* dan *Laravel*. Artikel ini akan menjelaskan konsep dan implementasi dari aplikasi ini, serta manfaat dan dampaknya dalam meningkatkan layanan perpustakaan di Dinas Perpustakaan dan Kearsipan Kota Samarinda.

2. METODE PENELITIAN

2.1 Pengumpulan Data

Proses pengumpulan data dilakukan guna memperoleh informasi yang relevan yang diperlukan demi mencapai tujuan penelitian [13]. Pendekatan pengumpulan data yang diterapkan dalam studi ini mencakup:

a) Observasi

Penulis melakukan observasi langsung terhadap proses pendaftaran anggota perpustakaan yang berlangsung di Dinas Perpustakaan Kota Samarinda untuk mengidentifikasi masalah dan hambatan yang ada. Hasil data yang diperoleh adalah Proses pendaftaran anggota baru di Dinas Perpustakaan dan Kearsipan Kota Samarinda terdiri dari lima tahap: (i) Calon anggota datang ke perpustakaan, (ii) Mengisi formulir pendaftaran manual dengan data pribadi, (iii) Petugas memverifikasi data yang diisi, (iv) Pembuatan kartu anggota, meliputi pencetakan dan *input* data ke sistem, (v) Penyerahan kartu kepada calon anggota yang kini resmi menjadi anggota perpustakaan.

b) Wawancara

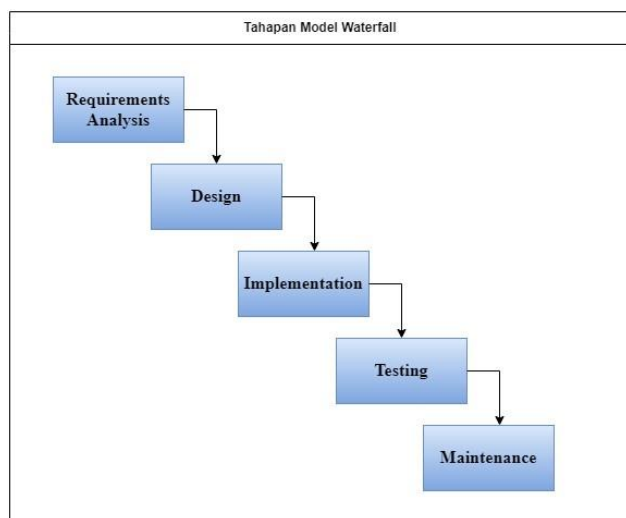
Penulis melakukan wawancara dengan petugas perpustakaan untuk memahami kebutuhan yang diperlukan dalam pengembangan aplikasi pendaftaran anggota perpustakaan berbasis *Android*. Hasil wawancara mengungkapkan beberapa fitur yang diharapkan dalam sistem baru, antara lain: *login* akun, registrasi anggota baru, *input* dan penyimpanan data, penghapusan data, pencarian berdasarkan kriteria, serta kemampuan melihat detail data anggota secara lengkap.

c) Studi Pustaka

Studi Pustaka dilakukan dengan meneliti beragam penelitian terdahulu, termasuk jurnal, skripsi, dan literatur yang relevan dengan pokok permasalahan penelitian ini [14].

2.2 Metode Pengembangan

Metode pengembangan sistem menggunakan metode *Waterfall*, yang termasuk dalam kerangka kerja *System Development Life Cycle* (SDLC). Dikenal karena pendekatannya yang terstruktur dan berkesinambungan dalam pengembangan perangkat lunak, model *Waterfall* menuntut penyelesaian setiap tahap sebelum memasuki tahap berikutnya, tanpa opsi untuk mundur ke tahap sebelumnya [15]. Tahapan Model *Waterfall* dapat dilihat pada Gambar 1 berikut.



Gambar 1. Tahapan Model *Waterfall*

a) Analisis Kebutuhan (*Requirements Analysis*)

Pada tahap awal, dilakukan pengkajian menyeluruh terhadap kebutuhan pengembangan aplikasi pendaftaran anggota perpustakaan untuk perangkat *Android*. Proses ini melibatkan pengamatan langsung di Dinas Perpustakaan Kota Samarinda, diskusi dengan staf perpustakaan untuk memahami ekspektasi fitur, serta penelaahan literatur terkait pengembangan aplikasi menggunakan *Flutter*, *Laravel*, dan *MySQL*. Hasil kajian ini menjadi dasar untuk menentukan fitur dan fungsi aplikasi.

b) Perancangan (*Design*)

Tahap ini meliputi desain struktur data menggunakan *MySQL* untuk penyimpanan informasi anggota dan data terkait. Diagram UML seperti *Use Case*, *Class*, dan *Activity* digunakan untuk visualisasi sistem. Arsitektur aplikasi menggabungkan *Flutter* untuk antarmuka pengguna dan *Laravel* untuk *backend*. Desain antarmuka dibuat menggunakan *BalsamiqWireframes* dengan fokus pada kemudahan penggunaan. Alur kerja dan algoritma aplikasi juga dirancang untuk memastikan efisiensi proses pendaftaran.

c) Implementasi (*Implementation*)

Pada fase ini, rancangan sistem diterjemahkan menjadi kode program fungsional. Bahasa pemrograman Dart digunakan bersama *frameworkFlutter* untuk mengembangkan fitur-fitur yang telah direncanakan. *Visual Studio Code* dipilih sebagai lingkungan pengembangan untuk memfasilitasi pengkodean yang efisien dan terstruktur.

d) Pengujian (*Testing*)

Aplikasi yang telah dikembangkan kemudian melalui proses pengujian *blackbox*. Metode ini berfokus pada fungsionalitas aplikasi tanpa melihat kode internal. Tujuannya adalah memastikan aplikasi dapat menerima *input*, memproses data, dan menghasilkan output sesuai kebutuhan pengguna. Tahap ini penting untuk mengidentifikasi dan memperbaiki kesalahan sebelum implementasi penuh. Selain pengujian *blackbox*, juga dilakukan pengujian menggunakan *Technology Acceptance Model* (TAM) untuk mengevaluasi penerimaan pengguna terhadap sistem. Pengujian ini mencakup aspek *perceived usefulness* (PU), *perceived ease of use* (PEOU), *attitude toward using* (ATU), *behavioral intention to use* (BI) dan *ExternalVariables* (EV).

e) Pemeliharaan (*Maintenance*)

Setelah implementasi, dilakukan pemeliharaan berkelanjutan terhadap aplikasi. Ini mencakup perbaikan *bug* yang mungkin muncul selama penggunaan, seperti masalah perhitungan atau tampilan. Selain itu, dilakukan peningkatan kinerja aplikasi, termasuk optimalisasi *database*, penyempurnaan algoritma, dan peningkatan performa secara keseluruhan untuk memastikan aplikasi tetap efektif dan efisien.

3. HASIL DAN PEMBAHASAN

3.1 Analisis Kebutuhan (*Requirements Analysis*)

Setelah mengumpulkan data yang diperlukan untuk pengembangan sistem informasi di Dinas Perpustakaan Kota Samarinda, hasil analisis kebutuhan seluruh pengguna sistem dibagi menjadi beberapa halaman yaitu halaman admin berbasis *website* yang dapat mengelola data calon anggota, data anggota perpustakaan dan data buku.

Halaman calon anggota berbasis *android* yang dapat mengisi formulir pendaftaran, lihat buku dan lihat profil.

3.2 Perancangan (*Design*)

Pada tahap ini dilakukan perancangan sistem berdasarkan hasil analisis yang diperoleh dari tahap sebelumnya. Pada tahap ini, rancangan sistem dikembangkan dalam bentuk aplikasi *android*. Rancangan sistem mencakup struktur menu, desain basis data, dan berbagai fitur. Untuk pemodelan data, digunakan UML (*Unified Modelling Language*) berupa *Use Case Diagram*, *Activity Diagram* dan *Class Diagram*.

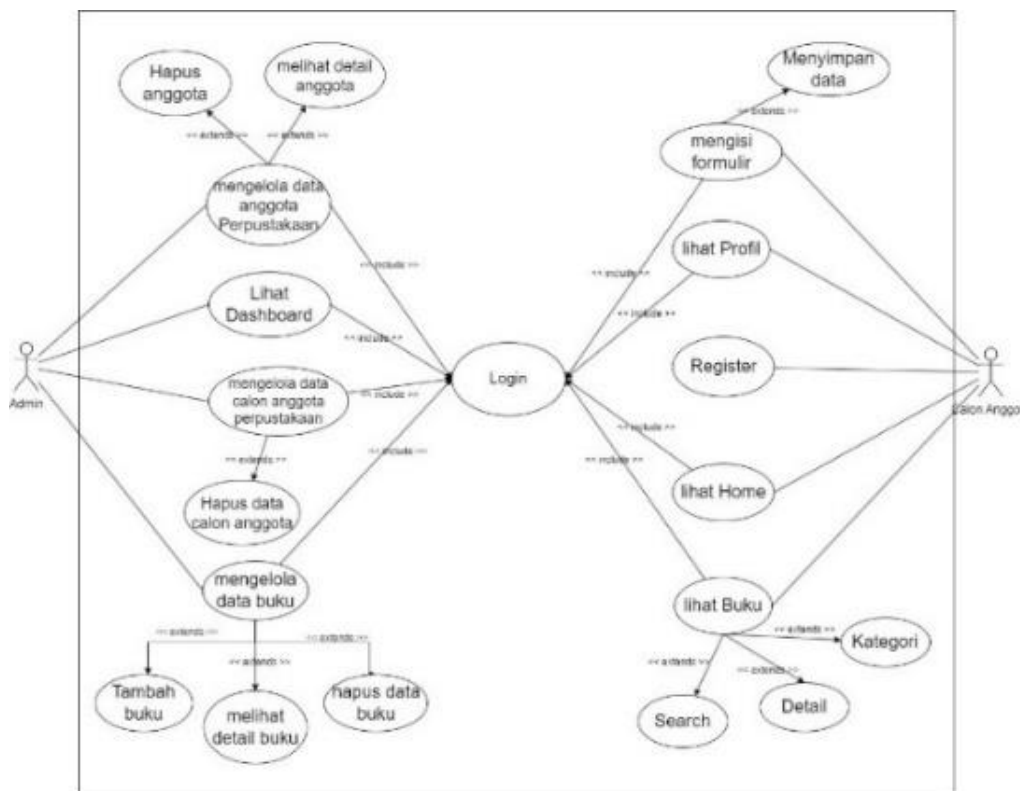
a) *Use Case Diagram*

Diagram *usecase* menjelaskan skenario *usecase* yang akan digunakan untuk menggambarkan alur proses dari perspektif aktor dalam *usecase* serta modul dari sistem yang ingin dibuat dan dikembangkan [16].

Pada gambar 2 *usecase* yang ditampilkan menunjukkan interaksi antara dua aktor utama, yaitu Admin dan Calon Anggota

Anggota, dengan sistem perpustakaan. Admin memiliki akses yang lebih luas dan bertanggung jawab atas pengelolaan data, termasuk mengelola data anggota perpustakaan, mengelola data calon anggota perpustakaan, serta mengelola data buku. Aktivitas yang dapat dilakukan oleh Admin mencakup melihat *dashboard*, menambah buku baru, melihat detail buku, serta menghapus data buku atau anggota. Semua tindakan ini dilakukan setelah Admin berhasil *login* ke dalam sistem.

Di sisi lain, Calon Anggota memiliki akses yang lebih terbatas dan fokus pada interaksi dengan informasi yang tersedia dalam sistem. Anggota dapat melakukan pendaftaran (*register*), mengisi formulir, melihat profil mereka, serta mencari dan melihat detail buku yang tersedia. Setelah *login*, Anggota dapat mengakses halaman utama, melihat kategori buku, dan menyimpan data yang telah mereka isi dalam formulir. Diagram ini menggambarkan dengan jelas peran dan aktivitas yang dapat dilakukan oleh masing-masing aktor dalam sistem perpustakaan, serta hubungan antar aktivitas tersebut. *Use Case Diagram* bisa dilihat pada gambar 2.

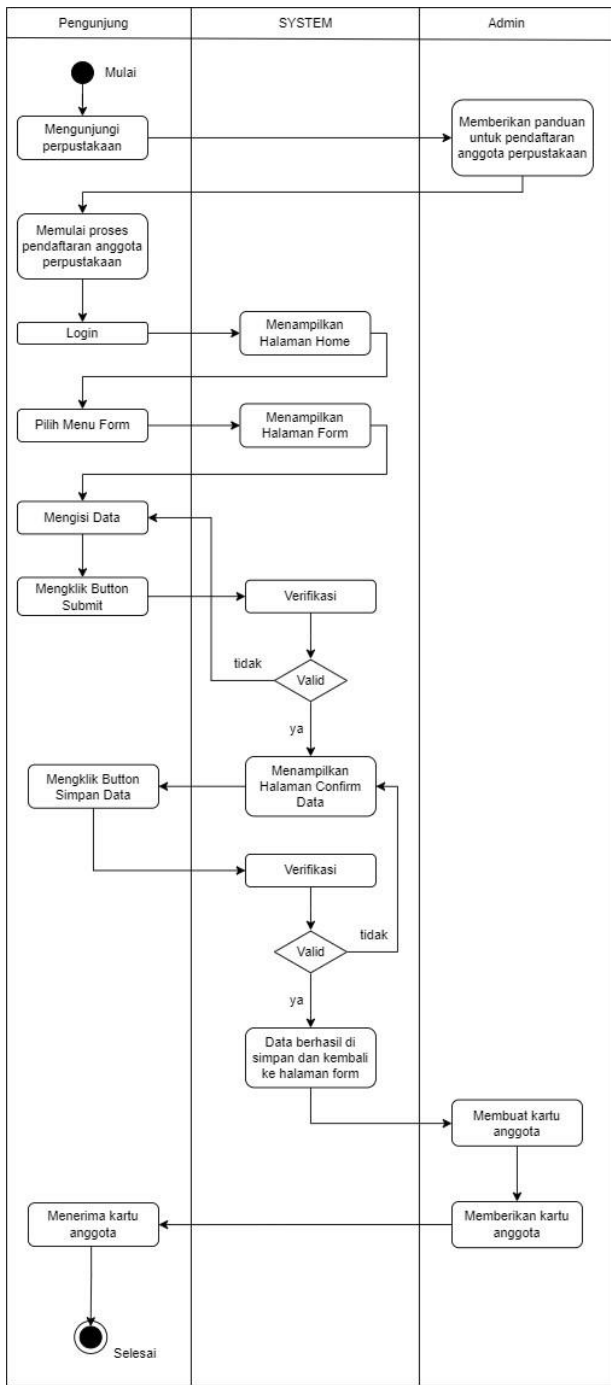


Gambar 2. *Use Case Diagram*

b) *Activity Diagram*

Diagram aktivitas pada Gambar 3 menunjukkan alur proses yang dimulai dengan pengguna masuk ke sistem. Setelah sistem menampilkan halaman utama, pengguna memilih menu formulir dan mengisi informasi yang diminta pada halaman yang muncul. Setelah menekan tombol kirim, sistem memeriksa kebenaran data. Jika ada kesalahan, pengguna diminta mengisi ulang. Jika benar, sistem

menampilkan halaman konfirmasi. Pengguna kemudian menekan tombol simpan, dan sistem melakukan pemeriksaan akhir. Apabila masih ditemukan kesalahan, proses kembali ke tahap pengisian. Setelah semua data diverifikasi dan disimpan oleh sistem, pengguna akan diarahkan kembali ke halaman formulir. Selanjutnya, admin akan membuat kartu anggota dan menyerahkannya kepada pengguna. *Activity Diagram* bisa dilihat pada gambar 3.



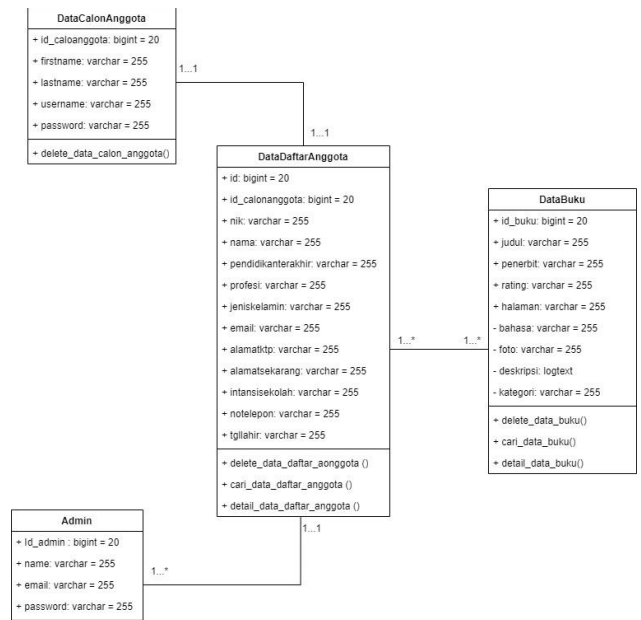
Gambar 3. Activity Diagram

c) Class Diagram

Diagram kelas memberikan gambaran sistem dari sudut pandang strukturnya, menunjukkan komponen-komponen utama yang akan dibuat dalam pengembangan sistem. Diagram ini memperlihatkan berbagai elemen sistem beserta ciri-ciri pentingnya, serta bagaimana elemen-elemen tersebut saling berhubungan. Pendekatan visual ini sangat membantu dalam merancang kerangka sistem secara keseluruhan, menyajikan peta konsep yang jelas dan teratur. Dengan demikian, tim pengembang dapat lebih mudah memahami dan merencanakan arsitektur sistem, termasuk

bagaimana data akan disimpan dan dikelola dalam penyimpanan yang terorganisir [17].

Pada gambar 4 *class diagram* ini menggambarkan hubungan antara empat kelas utama: 'DataCalonAnggota', 'DataDaftarAnggota', 'DataBuku', dan 'Admin'. 'DataCalonAnggota' memiliki relasi one-to-one dengan 'DataDaftarAnggota', dimana setiap calon anggota yang terdaftar di 'DataCalonAnggota' memiliki satu entri yang sesuai di 'DataDaftarAnggota'. 'DataDaftarAnggota' juga memiliki relasi one-to-one dengan 'Admin', yang menunjukkan bahwa setiap anggota yang terdaftar diawasi atau dikelola oleh satu admin. Selain itu, 'DataDaftarAnggota' memiliki relasi one-to-many dengan 'DataBuku', yang menunjukkan bahwa satu anggota dapat terhubung dengan banyak buku, namun satu buku hanya dapat terhubung dengan satu anggota dalam konteks relasi ini.



Gambar 4. Class Diagram

3.3 Implementasi (Implementation)

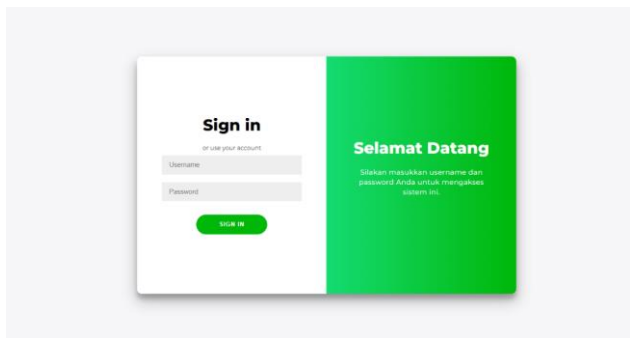
Tahap implementasi meliputi tampilan, menu, serta fitur-fitur dari hasil perancangan desain sistem yang telah dirancang dan disetujui oleh pengguna sistem atau pihak Perpustakaan Kota Samarinda.

Berikut beberapa hasil tampilan pada sistem pendaftaran anggota perpustakaan di Kota Samarinda.

3.3.1 Halaman Admin

a) Halaman Login Sistem Website

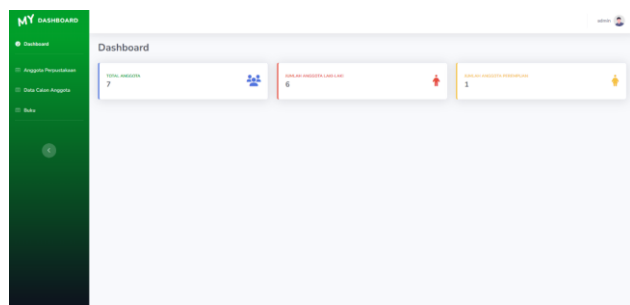
Pada gambar 5 pengguna harus memasukkan *username* dan *password* untuk mengakses situs web. Jika tidak sesuai, mereka tidak dapat mengakses sistem.



Gambar 5. Halaman Login Sistem Website

b) Halaman Dashboard Website

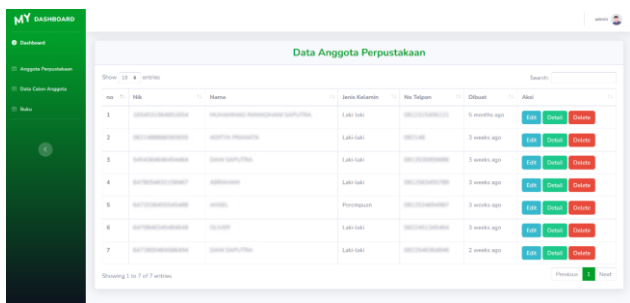
Dashboard website menampilkan data secara langsung kepada pengguna, termasuk total anggota, jumlah anggota laki-laki, dan jumlah anggota perempuan. Panel navigasi di sebelah kiri memudahkan akses ke bagian lain dari dashboard, seperti daftar anggota perpustakaan, data calon anggota, dan data buku. Profil administrator dan informasi nama terletak di bagian kanan atas. Dapat dilihat pada gambar 6.



Gambar 6. Halaman Dashboard Sistem Website

c) Halaman Data Anggota Perpustakaan Sistem Website

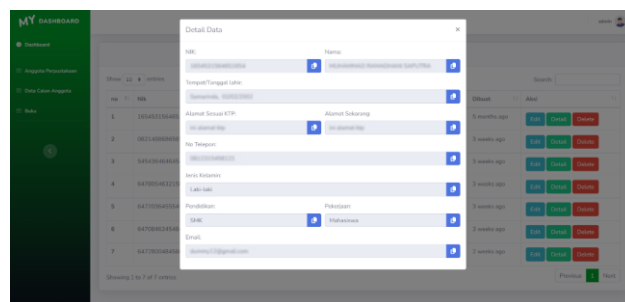
Pada gambar 7 merupakan halaman "Data Anggota Perpustakaan" pada website menampilkan tabel yang berisi informasi anggota perpustakaan, seperti NIK, nama, jenis kelamin, nomor telepon, tanggal pembuatan akun, serta opsi untuk melihat detail atau menghapus data anggota. Terdapat juga fitur pencarian di sudut kanan atas tabel untuk memudahkan pengguna mencari data anggota.



Gambar 7. Tampilan Halaman Data Anggota Perpustakaan website

d) Tampilan Detail Data Anggota Perpustakaan Website

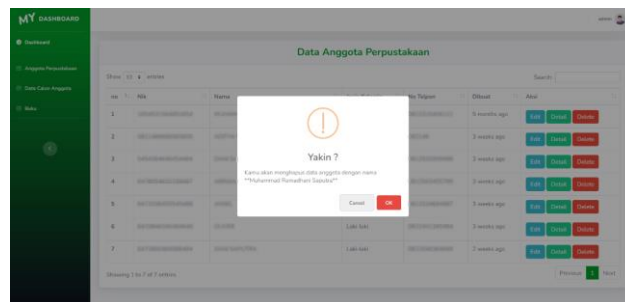
Ketika pengguna mengklik tombol "Detail" pada halaman "Data Anggota Perpustakaan", halaman "Detail" muncul dalam bentuk *pop-up* yang menampilkan informasi lengkap setiap anggota, termasuk NIK, nama lengkap, tempat dan tanggal lahir, alamat sesuai KTP, alamat sekarang, nomor telepon, jenis kelamin, pendidikan, pekerjaan, dan email. Dengan tampilan yang disusun dalam bentuk formulir, pengguna dapat dengan mudah menelusuri dan memeriksa detail dari setiap anggota. Dapat dilihat pada gambar 8.



Gambar 8. Tampilan Detail Data Anggota Perpustakaan website

e) Tampilan Hapus Data

Saat Anda mengklik tombol "Delete" pada halaman "Data Anggota", gambar 9 menunjukkan kotak dialog konfirmasi penghapusan data anggota yang muncul. Kotak dialog ini memberikan peringatan dengan pesan "Yakin?" dan mengkonfirmasi bahwa Anda telah menghapus data anggota dengan nama yang ditentukan, dalam hal ini "Muhammad Ramadhani Saputra". Pengguna memiliki dua opsi: tombol "Batal" untuk membatalkan penghapusan dan tombol "OK" untuk melanjutkannya.

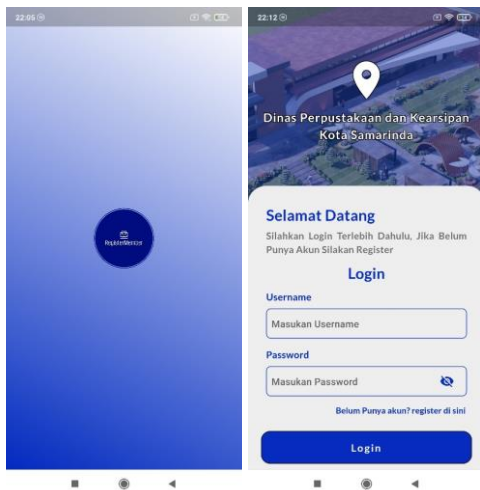


Gambar 9. Tampilan Hapus Data website

3.3.2 Halaman Calon Anggota

a) Tampilan Splashscreen dan Halaman Login Sistem Android

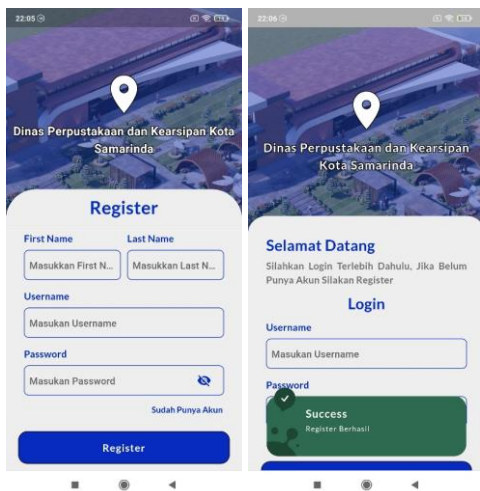
Pada gambar 10 menampilkan halaman ini *splashscreen* aplikasi "RegisterMember" menampilkan logo dan teks "RegisterMember" di tengah layar dengan latar belakang gradien biru. Ini adalah tampilan pembuka saat aplikasi pertama kali dijalankan. Halaman *login* memiliki *text* instansi, pesan selamat datang, *input* username, password, tombol "Login", dan tautan untuk pendaftaran.



Gambar 10. Tampilan Halaman *Splash Screen* dan Halaman *Login*

b) Halaman *Register*

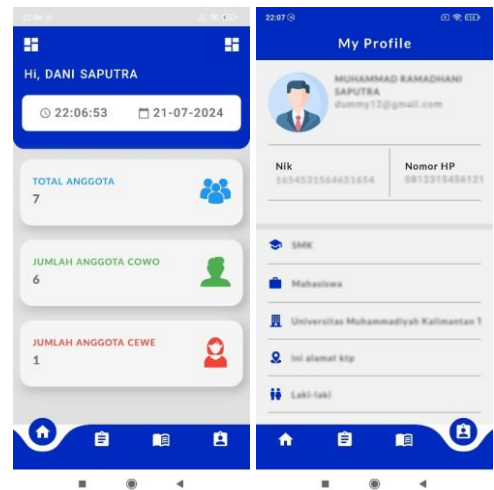
Pada gambar 11 menampilkan halaman pendaftaran di mana pengguna diminta memasukkan data seperti *first name*, *last name*, *username*, dan *password*, serta tombol '*Register*' untuk menyimpan data ke dalam sistem. Jika pendaftaran berhasil, akan muncul pemberitahuan bahwa pendaftaran berhasil.



Gambar 11. Tampilan Halaman *Register*

c) Halaman *Home* dan *Profil*

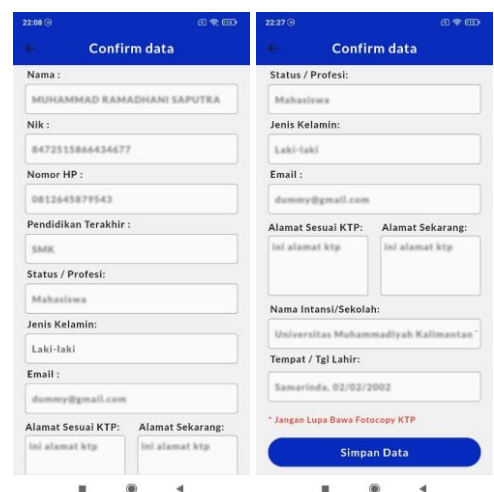
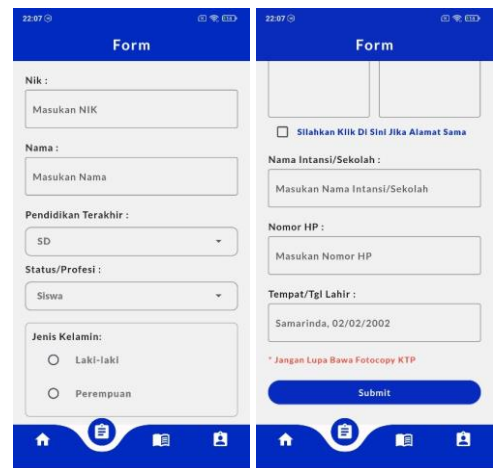
Pada gambar 12 menampilkan halaman *home*. Pada halaman ini terdapat informasi nama pengguna, disertai dengan jam dan tanggal saat ini. Terdapat juga tiga data statistik, mencakup total anggota, jumlah anggota pria, jumlah anggota wanita. Di bagian bawah terdapat menu navigasi *home*, formulir, buku dan profil.



Gambar 12. Tampilan Halaman *Home* dan Halaman *Profil*

d) Halaman *Formulir* dan *Konfirmasi Data*

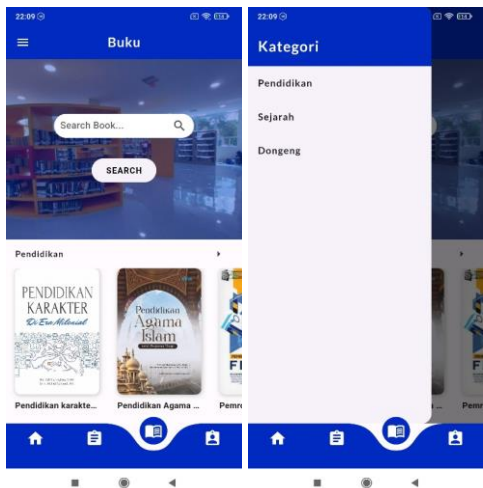
Pada gambar 13 halaman ini, pengguna meng-*input* data sesuai dengan kriterianya dan terdapat tombol *submit* untuk menuju ke halaman konfirmasi data. Halaman konfirmasi data memungkinkan pengguna untuk meninjau dan mengkonfirmasi data sebelum dimasukkan ke dalam sistem.



Gambar 13. Tampilan Halaman *Formulir* dan Halaman *Konfirmasi Data*

e) Halaman Buku dan Halaman Kategori

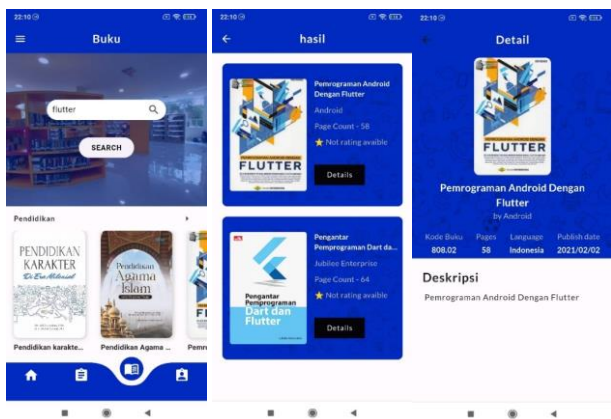
Pada gambar 14 menampilkan halaman buku dan kategori yang dilengkapi dengan sidebar untuk melihat kategori, fitur pencarian buku, dan tampilan buku dengan kategori pendidikan. Di samping itu, halaman tersebut menampilkan kategori buku seperti pendidikan, sejarah, dan dongeng. Selain itu, halaman hasil kategori menampilkan semua kategori yang dipilih oleh pengguna.



Gambar 14. Tampilan Halaman Buku dan Halaman Kategori

f) Halaman Hasil Search Buku dan Halaman Detail Buku

Pada gambar 15 menampilkan hasil pencarian buku dengan kata kunci 'Flutter' dan halaman detail buku. Di halaman hasil, terdapat buku-buku yang sesuai dengan kata kunci tersebut, masing-masing dilengkapi dengan tombol 'Details' untuk melihat detail lengkapnya. Halaman detail buku menampilkan informasi seperti gambar buku, judul, rating, jumlah halaman, bahasa, tanggal terbit, dan deskripsi buku.



Gambar 15. Tampilan Halaman Hasil Search Buku dan Halaman Detail Buku

3.4 Pengujian (Testing)

Pada tahap ini, penulis melakukan pengujian sistem menggunakan *blackbox testing* yang bertujuan untuk mengevaluasi fungsionalitas dari fitur-fitur yang terdapat

pada halaman sistem. Pengujian ini mencakup dua sistem, yaitu sistem *website* dan sistem aplikasi *Android*. Selain itu, penulis juga menggunakan *Technology Acceptance Model* (TAM) untuk mendapatkan respon yang berfokus pada lima variabel, yaitu *Perceived usefulness* (PU), *Perceived ease of use* (PEOU), *Attitude toward using* (ATU), *Behavioral intention to use* (BI), dan *External Variables* (EV).

3.4.1 Pengujian Sistem Website

Pengujian ini mencakup dua sistem, yaitu sistem *website* dan sistem aplikasi *Android* yang dapat dilihat pada Tabel 1 dan Tabel 2 berikut.

Tabel 1. Pengujian Sistem Website

Testcase	Keterangan	Hasil
Loginuser	Sistem berhasil menampilkan <i>dashboard</i>	Berhasil
Loginuser	Sistem berhasil menampilkan pemberitahuan kesalahan data	Berhasil
Lihat detail data	Sistem berhasil menampilkan <i>popup</i> detail data	Berhasil
Hapus data	Sistem berhasil menampilkan <i>popup</i> hapus data dan data berhasil dihapus	Berhasil
fitur search	Sistem berhasil menampilkan data sesuai dengan <i>keyword</i>	Berhasil
	Sistem berhasil untuk tidak menampilkan data <i>keyword</i> yang salah	Berhasil
Showentries	Sistem berhasil menampilkan entry sesuai dengan jumlah yang dipilih	Berhasil
Logout	Sistem berhasil menampilkan <i>popup</i> konfirmasi <i>logout</i> dan berhasil menuju halaman <i>login</i>	Berhasil

3.4.2 Pengujian Sistem Android

Tabel 2. Pengujian Sistem Android

Testcase	Keterangan	Hasil
Loginuser	Sistem berhasil menampilkan halaman <i>home</i>	Berhasil
Loginuser	Sistem berhasil menampilkan pemberitahuan kesalahan data	Berhasil
Register	Sistem berhasil menampilkan notifikasi bahwa data berhasil <i>register</i> dan diarahkan ke halaman <i>login</i>	Berhasil
Register	Sistem berhasil menampilkan notifikasi kesalahan	Berhasil
Mengisi formulir	Sistem berhasil menampilkan halaman <i>Confirm</i> data	Berhasil

Testcase	Keterangan	Hasil
Mengisi formulir	Sistem berhasil menampilkan kesalahan	Berhasil
Confirm data	Sistem berhasil menampilkan notifikasi bahwa data berhasil di daftar dan menuju ke halaman formulir	Berhasil
Confirm data	Sistem berhasil menampilkan notifikasi bahwa nik sudah terdaftar dan data tidak bisa <i>input</i> dua kali	Berhasil
Search buku	Sistem berhasil menampilkan data sesuai dengan <i>keyword</i>	Berhasil
Search buku	Sistem berhasil untuk tidak menampilkan data <i>keyword</i> yang salah	Berhasil
Detail buku	Sistem berhasil menampilkan detail buku yang dipilih	Berhasil
klik kategori buku	Sistem berhasil menampilkan semua data yang sesuai dengan kategori yang dipilih	Berhasil
logout	Sistem berhasil menampilkan <i>popup</i> konfirmasi <i>logout</i> dan berhasil menuju halaman <i>login</i>	Berhasil

3.4.3 Pengujian TAM

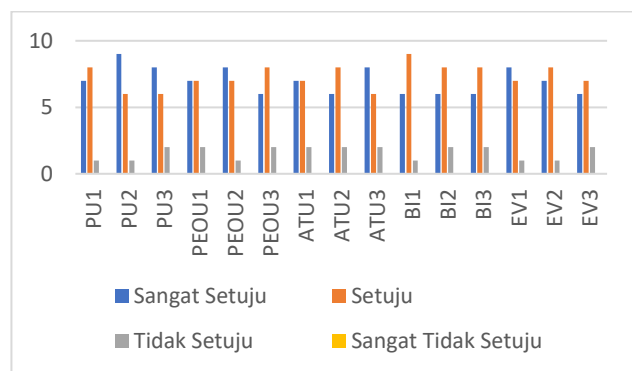
Pada pengujian ini bertujuan untuk mendapatkan respons yang berfokus pada 5 variabel yaitu, *PerceivedUsefulness* (PU), *PerceivedEaseof Use* (PEOU), *AttitudeTowardUsing* (ATU), *BehavioralIntentionto Use* (BI), dan *ExternalVariables* (EV). Tabel 3 adalah rincian pertanyaan dari tiap variabel.

Tabel 3. Kuesioner

Variable	Pertanyaan	Symbol
<i>PerceivedUsefulness</i> (PU)	Apakah Anda merasa bahwa menggunakan aplikasi pendaftaran anggota perpustakaan berbasis <i>Android</i> akan meningkatkan efisiensi proses pendaftaran?	PU1
	Apakah Anda percaya bahwa aplikasi ini akan membuat proses pendaftaran anggota perpustakaan menjadi lebih cepat?	PU2
	Apakah Anda merasa bahwa aplikasi ini akan mengurangi kesalahan dalam pencatatan data anggota perpustakaan?	PU3
<i>PerceivedEaseof Use</i> (PEOU)	Apakah Anda merasa bahwa aplikasi pendaftaran anggota perpustakaan berbasis <i>Android</i> ini mudah untuk dipelajari?	PEOU1

Variable	Pertanyaan	Symbol
<i>AttitudeTowardUsing</i> (ATU)	Apakah Anda merasa bahwa antarmuka aplikasi pendaftaran anggota perpustakaan ini intuitif dan mudah dimengerti?	PEOU2
	Apakah Anda merasa bahwa mengoperasikan aplikasi pendaftaran anggota perpustakaan ini tidak memerlukan banyak usaha?	PEOU3
	Apakah Anda merasa senang menggunakan aplikasi ini untuk mendaftar sebagai anggota perpustakaan?	ATU1
<i>BehavioralIntentionto Use</i> (BI)	Anda merasa puas menggunakan aplikasi pendaftaran anggota perpustakaan ini?	ATU2
	Anda memberikan nilai yang baik secara keseluruhan terhadap aplikasi pendaftaran anggota perpustakaan ini?	ATU3
	besar kemungkinan Anda akan menggunakan aplikasi pendaftaran anggota perpustakaan berbasis <i>Android</i> ini di masa mendatang?	BI1
<i>ExternalVariables</i> (EV)	besar keinginan Anda untuk merekomendasikan aplikasi pendaftaran anggota perpustakaan ini kepada orang lain?	BI2
	Anda akan memilih menggunakan aplikasi ini jika tersedia sebagai pilihan untuk pendaftaran atau layanan perpustakaan lainnya?	BI3
	infrastruktur teknologi yang tersedia (misalnya koneksi internet, <i>smartphone</i>) untuk mendukung penggunaan aplikasi pendaftaran anggota perpustakaan ini?	EV1
<i>ExternalVariables</i> (EV)	Anda merasa bahwa lingkungan sosial (misalnya teman, keluarga, sesama pengguna perpustakaan) mendukung penggunaan aplikasi pendaftaran anggota perpustakaan berbasis <i>Android</i> ?	EV2
	Anda percaya bahwa data pribadi Anda aman saat menggunakan aplikasi pendaftaran anggota perpustakaan ini?	EV3

Berikut adalah ringkasan hasil jawaban responden terkait kinerja sistem, yang diperoleh dari pertanyaan-pertanyaan pada Tabel 3. Hasil lengkapnya dapat dilihat pada Gambar 16.



Gambar 16. Grafik Hasil Kuesioner

Berdasarkan hasil kuesioner dari 16 responden, dapat disimpulkan dengan menggunakan skala Likert bahwa responden memilih dari empat opsi jawaban dengan bobot sebagai berikut: Sangat Setuju (ST) = 4, Setuju (S) = 3, Tidak Setuju (TS) = 2, dan Sangat Tidak Setuju (STS) = 1.

Tabel 4. Hasil Nilai Persentase Tiap Pertanyaan Kuesioner

no	Pertanyaan	Frekuensi Jawaban				Jumlah Skor				Total Skor
		SS	S	TS	STS	SS	S	TS	STS	
1	PU1	7	8	1	0	44	57	0	0	101
2	PU2	9	6	1	0	48	51	2	0	101
3	PU3	8	6	2	0	36	57	4	0	97
4	PEOU1	7	7	2	0	32	60	4	0	96
5	PEOU2	8	7	1	0	40	54	4	0	98
6	PEOU3	6	8	2	0	36	57	4	0	97
7	ATU1	7	7	2	0	44	51	4	0	99
8	ATU2	6	8	2	0	44	54	2	0	100
9	ATU3	8	6	2	0	36	57	4	0	97
10	BI1	6	9	1	0	52	48	2	0	102
11	BI2	6	8	2	0	36	57	4	0	97
12	BI3	6	8	2	0	40	54	4	0	98
13	EV1	8	7	1	0	40	57	2	0	99
14	EV2	7	8	1	0	40	57	2	0	99
15	EV3	6	7	2	0	40	51	6	0	97
Total Akhir Skor										798
Total Skor Tertinggi										(Skor Skala Tertinggi x Jumlah Responden x Jumlah Soal) 960
Persentase Rata-Rata										(Total Akhir Skor / Total Skor Tertinggi x 100) 83%

Pada tabel 4 Hasil survei menunjukkan tingkat kepuasan yang tinggi dengan nilai persentase rata-rata sebesar 83%. Beberapa faktor yang mendukung hasil positif ini antara lain antarmuka pengguna yang intuitif dan mudah digunakan, efisiensi proses pendaftaran yang lebih signifikan dibandingkan dengan metode manual, serta fitur pencarian buku yang memudahkan pengguna dalam mengakses informasi. Selain itu, keamanan data juga terjamin melalui penggunaan *Laravel* sebagai *backend*, dan aplikasi menunjukkan responsivitas yang baik pada berbagai perangkat *Android*.

3.5 Pemeliharaan (*Maintenance*)

Pada tahap pemeliharaan, dilakukan 2 langkah utama: perbaikan sistem, peningkatan sistem. Perbaikan sistem diperlukan jika ditemukan *bugs* atau kesalahan yang tidak terdeteksi selama pengujian, untuk memastikan sistem tetap berfungsi dengan baik. Peningkatan sistem dilakukan untuk memperbaiki kinerja atau menambah fitur baru setelah sistem berjalan, agar tetap relevan dan memenuhi kebutuhan pengguna. Pemeliharaan ini memastikan sistem berjalan dengan lancar dan siap menghadapi perubahan di masa depan.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pengembangan aplikasi pendaftaran anggota perpustakaan berbasis *Android* menggunakan model *Waterfall* pada Dinas Perpustakaan dan Kearsipan Kota Samarinda, dapat ditarik beberapa kesimpulan. (i) Aplikasi ini berhasil mengatasi permasalahan dalam proses pendaftaran anggota perpustakaan yang sebelumnya dilakukan secara manual. Dengan adanya aplikasi ini, calon anggota dapat melakukan pendaftaran secara *online* tanpa harus mengunjungi perpustakaan berulang kali, menghemat waktu proses pendaftaran. (ii) Fitur pencarian buku yang diimplementasikan dalam aplikasi memudahkan anggota perpustakaan untuk mencari ketersediaan buku tanpa harus datang langsung ke perpustakaan, mengurangi kemungkinan kunjungan yang sia-sia karena buku yang dicari tidak tersedia. Hasil kuesioner menunjukkan tingkat kepuasan yang tinggi, dengan nilai persentase rata-rata sebesar 83%, yang berarti bahwa sebagian besar responden setuju atau sangat setuju atas implementasi sistem pendaftaran anggota perpustakaan berbasis *android*.

DAFTAR PUSTAKA

- [1] A. Bimantoro, A. Pramesti, Wanda, W. Bakti, Satria, M. A. Samudra, and Y. Amrozi, "Paradoks Etika Pemanfaatan Teknologi Informasi di Era 5.0," *J. Teknol. Inf.*, vol. 7, no. 1, pp. 58–68, 2021, doi: 10.52643/jti.v7i1.1425.
- [2] Yosepha Pusparisa, "Pengguna *Smartphone* diperkirakan Mencapai 89% Populasi pada 2025," databoks.katadata.co.id. Accessed: Mar. 06, 2024. [Online]. Available: <https://databoks.katadata.co.id/datapublish/2020/09/15/pengguna-smartphone-diperkirakan-mencapai-89-populasi-pada-2025>
- [3] C. Chandra, D. Sanjaya, J. Narabel, N. Vilano, and S. Budi, "Analisis Estimasi Waktu Antrian dengan menggunakan Markov Chain dan Algoritma Pagerank," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 3, pp. 406–414, 2019.
- [4] S. Hendrawan, A. Manuputty, and B. Haryanto, "Design of Information Systems for Research Permit Application with Agile Method and *Website* Based *Laravel Framework*," *J. Inf. Syst. Informatics*, vol. 2, no. 1, pp. 60–78, Mar. 2020, doi: 10.33557/journalisi.v2i1.45.
- [5] I. Zulfa and R. Wanda, "Klik: kajian ilmiah informatika dan komputer rancangan sistem informasi akademik berbasis *website* menggunakan php dan *MySQL*," *Klik Kaji. Ilm. Inform. Dan Komput.*, vol. 3, no. 4, pp. 393–399, 2023, [Online]. Available: <https://djournals.com/klik/article/view/617>
- [6] Y. Anis, A. B. Mukti, and A. N. Rosyid, "KLIK: Kajian Ilmiah Informatika dan Komputer Penerapan Model *Waterfall* Dalam Pengembangan Sistem Informasi Aset Destinasi Wisata Berbasis *Website*," *Media Online*, vol. 4, no. 2, pp. 1134–1142, 2023, doi: 10.30865/klik.v4i2.1287.
- [7] S. R. Siregar and P. Pristiwanto, "Penerapan Metode *Waterfall* Dalam Pengembangan Sistem Informasi Masjid," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 3, no. 1, pp. 26–32, 2022, [Online]. Available: <https://djournals.com/klik>
- [8] A. H. Kahfi, M. Hasan, and A. Fazriansyah, "Perancangan Program Pembayaran Administrasi Sekolah Berbasis *Website* Menggunakan Metode *Waterfall*," *Anal. Tingkat Kepuasan Pengguna Sist. Inf. Perpust. Menggunakan Pieces Framew.*, vol. 3, no. 6, pp. 1063–1069, 2023, doi: 10.30865/klik.v3i6.801.
- [9] M. F. A. Azrial and N. Fadillah, "Sistem Informasi Pengangkutan Pupuk Menggunakan Metode *Waterfall* (Studi Kasus PT. Pupuk Iskandar Muda, Aceh Utara)," *J-ICOM - J. Inform. dan Teknol. Komput.*, vol. 1, no. 2, pp. 75–81, 2020, doi: 10.33059/j-icom.v1i2.2897.
- [10] F. Nugraha, D. Restendi, and A. Triyanto, "Pengembangan Sistem Pelatihan Jarak Jauh Berbasis Moodle di Balai Diklat Keagamaan Bandung," *Andragogi J. Diklat Tek. Pendidik. dan Keagamaan*, vol. 8, no. 2, pp. 540–552, 2020, doi: 10.36052/andragogi.v8i2.166.
- [11] D. Handayani and M. Salam, "Aplikasi Sistem Informasi Simpan Pinjam Koperasi Berbasis *Website* Menggunakan Metode *Waterfall*," *Media Online*, vol. 3, no. 5, pp. 425–434, 2023, [Online]. Available: <https://djournals.com/klik>
- [12] N. Fadilah, A. Ikhwan, and M. Alda, "Pengembangan Sistem Informasi Perpustakaan Pada Dinas Perpustakaan dan Kearsipan Kota Medan Berbasis *Android*," *J-SISKO TECH (Jurnal Teknol. Sist. Inf. dan Sist. Komput. TGD)*, vol. 6, no. 2, p. 298, 2023, doi: 10.53513/jsk.v6i2.8228.
- [13] M. Yasin, S. Garancang, and A. A. Hamzah, "Metode Dan Instrumen Pengumpulan Data Penelitian Kuantitatif dan Kualitatif," *J. Int. Multidiscip. Res.*, vol. 2, no. 3, pp. 162–173, 2024.
- [14] D. Assyakurrohim, D. Ikhrum, R. A. Sirodj, and M. W. Afgani, "Metode Studi Kasus dalam Penelitian Kualitatif," *J. Pendidik. Sains dan Komput.*, vol. 3, no. 01, pp. 1–9, 2022, doi: 10.47709/jpsk.v3i01.1951.
- [15] B. Fachri and R. W. Surbakti, "Perancangan Sistem Dan Desain Undangan Digital Menggunakan Metode *Waterfall* Berbasis *Website* (Studi Kasus: Asco Jaya)," *J. Sci. Soc. Res.*, vol. 4, no. 3, p. 263, 2021, doi: 10.54314/jssr.v4i3.692.
- [16] I. Irianto, S. Sudarmin, and A. Afrisawati, "Penerapan Metode Customer Relationship Management Pada Penjualan Toko Baju Azzahra," *J. Sci. Soc. Res.*, vol. 4, no. 2, p. 191, 2021, doi: 10.54314/jssr.v4i2.584.
- [17] E. Dwi Setiawan and M. Raharjo, "PERANCANGAN *WEBSITE* PEMBELAJARAN BAHASA JEPANG DENGAN FOKUS MATERI KLAUSAMELALUI METODE *WATERFALL*," *J. Inform. Terpadu*, vol. 9, no. 1, pp. 34–39, 2023, [Online]. Available: <https://journal.nurulfikri.ac.id/index.php/JIT>