

PERANCANGAN APLIKASI ADMINISTRASI KANTOR BERBASIS ANDROID PADA PT RONADAMAR SEJAHTERA

Nicolas¹⁾ Tony²⁾

^{1) 2)} Program Studi Sistem Informasi, Fakultas Teknologi Informasi Universitas Tarumanagara

Jl. Letjen S. Parman No.1, Tomang, Grogol petamburan, Jakarta Barat 11440

email : nicolasn28@gmail.com ¹⁾ tony@fti.untar.ac.id ²⁾

ABSTRAK

Kegiatan administrasi di sebuah kantor dituntut semakin efisien seiring perkembangan teknologi. PT Ronadamar Sejahtera yang merupakan distributor untuk produk Lem Fox telah memiliki sistem administrasi berbasis website yang menggunakan Framework Laravel, namun penggunaannya pada perangkat ponsel masih kurang optimal karena keterbatasan tampilan dan kenyamanan. Untuk mengatasi hal tersebut, penelitian ini bertujuan untuk merancang aplikasi administrasi kantor berbasis Android yang terintegrasi dengan sistem website yang sudah ada. Aplikasi dikembangkan menggunakan Framework Flutter sebagai front-end mobile dan Framework Laravel sebagai back-end, dengan metode System Development Life Cycle (SDLC) dengan pendekatan Waterfall. Tahap awal adalah tahap perancangan yang meliputi analisis kebutuhan, pemodelan sistem menggunakan Unified Modeling Language (UML), perancangan Database, dan perancangan Wireframe dari aplikasi. Dengan adanya aplikasi ini, diharapkan proses administrasi di PT Ronadamar Sejahtera dapat dilakukan dengan lebih mudah, fleksibel, dan efisien melalui perangkat Android.

Key words

administrasi, android, flutter, laravel, mobile, front-end, back-end, SDLC, waterfall, wireframe, database

ABSTRACT

Administrative activities in an office are required to become more efficient along with technological advancements. PT Ronadamar Sejahtera, a distributor of Lem Fox products, already has a web-based administrative system built using the Laravel framework. However, its usability on mobile devices remains limited due to display and convenience constraints. To address this issue, this study aims to design an Android-based office administration application integrated with the existing web system. The application is developed using the Flutter framework for the mobile front-end and the Laravel framework for the back-end, following the System Development Life Cycle (SDLC) method with a Waterfall approach. The initial phase involves the design stage, including requirements analysis, system modeling using Unified Modeling Language (UML), database design, and wireframe design. With this application, it is expected that administrative processes at PT Ronadamar Sejahtera can be carried out more easily, flexibly, and efficiently through Android devices.

Key words

administration, Android, Flutter, Laravel, mobile, front-end, back-end, SDLC, Waterfall, wireframe, database

1. Pendahuluan

Administrasi pada sebuah kantor di perusahaan sangat penting bagi aktivitas karyawan sehari-hari. Berdasarkan buku yang ditulis oleh Restanti [1], administrasi perkantoran merupakan penerapan fungsi-fungsi yang ditujukan untuk meningkatkan efektifitas kerja kantor dan meningkatkan kepuasaan karyawan. Fungsi-fungsi yang dimaksud adalah fungsi seperti perencanaan, pengorganisasian, penggerakan, dan pengawasan kantor. Pada era digitalisasi, administrasi perkantoran sudah mulai menerapkan pendekatan yang modern terhadap proses administrasinya. Choirunnissa et al. [2] menjelaskan bahwa terjadinya digitalisasi administrasi perkantoran dikarenakan proses manual yang biasa dilakukan oleh sebuah kantor memiliki berbagai masalah seperti rentan terjadi masalah akibat human-error, prosedur administrasi yang rumit dan lama, proses birokrasi yang berbelit-belit dan sebagainya. Digitalisasi administrasi perkantoran dapat menyelesaikan masalah-masalah ini dengan mengimplementasikan teknologi-teknologi yang tersedia terhadap proses administrasi dan membuatnya menjadi sebuah sistem [3].

PT Ronadamar Sejahtera yang selanjutnya akan disingkat menjadi RDS merupakan perusahaan yang bergerak di bidang distributor untuk produk-produk dari perusahaan utamanya Produk yang didistribusikan berupa berbagai jenis lem yang bernama lem Fox. PT Ronadamar Sejahtera sendiri memiliki 4 cabang yang tersebar di seluruh pulau Jawa, sehingga memiliki banyak sekali karyawan yang tersebar luas di keempat cabang. Dikarenakan karyawan yang banyak, PT Ronadamar Sejahtera memiliki sebuah teknologi untuk segala proses administrasi kantor yang dinamakan aplikasi Integrasi-RDS. Integrasi-RDS merupakan aplikasi berbasis website yang difokuskan untuk mempermudah pengelolaan karyawan, mengurangi penggunaan kertas pada kantor, dan otomatisasi proses administrasi.

Namun, permasalahan muncul ketika karyawan menggunakan ponsel untuk menggunakan aplikasi Integrasi-RDS. Pengembangan aplikasi Integrasi-RDS berbasis website dilakukan menggunakan 1 komputer dengan perancangan User Interface yang dikhawasukan

untuk penggunaan di komputer. Bagi karyawan yang hanya memiliki ponsel, *website* yang digunakan akan terasa kurang optimal dan sulit untuk di navigasi sehingga kurang nyaman ketika menggunakan website Integrasi-RDS. Tidak semua karyawan juga dapat mengakses komputer setiap saat sehingga penggunaan aplikasi Integrasi-RDS menjadi kurang praktis. Absensi karyawan di kantor juga kurang dirasa kurang praktis karena sistem absen yang diterapkan harus dilakukan di kantor dengan mesin absen sehingga ketika ada karyawan yang sedang tugas diluar kantor tidak bisa melakukan absensi.

Maka dari itu, aplikasi administrasi kantor berbasis *Android* dirancang untuk menyelesaikan permasalahan-permasalahan tersebut dengan mengintegrasikan aplikasi *website* yang sudah dibuat menjadi aplikasi berbasis *Android* yang dapat digunakan pada ponsel. Aplikasi tersebut akan dirancang menggunakan metode *System Development Life Cycle (SDLC)* dengan pendekatan *Waterfall*. Pada perancangan aplikasi administrasi kantor berbasis *Android*, perancangan *Unified Modeling Language (UML)* dan perancangan *Entity Relationship Diagram (ERD)* juga dilakukan agar pengembangan aplikasi menjadi efektif dan efisien. Selain itu, perancangan *Wireframe* dari aplikasi juga akan dilakukan agar pengembangan *User Interface* menjadi lebih mudah. Perancangan aplikasi administrasi kantor berbasis *Android* diharapkan dapat meningkatkan efektifitas dan efisiensi proses administrasi pada kantor, serta sebagai langkah digitalisasi kantor pada PT Ronadamar Sejahtera.

2. Metode Penelitian

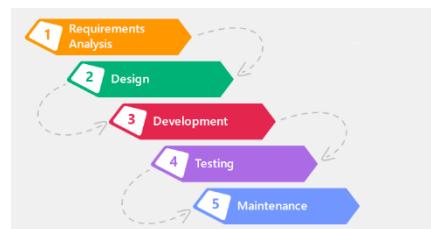
Perancangan aplikasi administrasi kantor berbasis *Android* menggunakan metode penelitian kualitatif dengan melakukan survei di salah satu kantor cabang Jakarta dari PT Ronadamar Sejahtera. Survei dilakukan dengan menganalisa infrastruktur dan teknologi yang tersedia pada kantor cabang Jakarta. Setelah dilakukan survei, wawancara bersama staf *IT* di kantor cabang Jakarta juga dilakukan untuk memahami lebih dalam mengenai kendala-kendala dan kebutuhan yang diharapkan dapat ditambahkan pada aplikasi yang akan dikembangkan. Dengan dilakukannya survei dan wawancara, perancangan aplikasi yang dihasilkan sesuai dengan kebutuhan yang diharapkan perusahaan.

Untuk pengembangan aplikasi, metode yang digunakan adalah metode *System Development Life Cycle (SDLC)*. Berdasarkan buku yang ditulis oleh Kundu [4], *Software Development Life Cycle (SDLC)* adalah *Framework* yang menjelaskan tugas-tugas yang harus dilakukan di setiap tahap pada pengembangan *software*. *SDLC* memiliki bergabagai macam pendekatan, namun pendekatan yang akan digunakan untuk merancang aplikasi administrasi kantor berbasis *Android* adalah

pendekatan *Waterfall*. Setelah itu akan dilakukan perancangan *UML* dan basis data.

2.1. Pendekatan *Waterfall*

Berdasarkan buku yang ditulis oleh Tsui *et al.*[5], pendekatan *Waterfall* adalah pendekatan pada proses pengembangan *software* dengan metode *SDLC* yang bersifat sekuensial yang berarti proyek dilakukan secara bertahap secara jelas dengan setiap tahap harus diselesaikan sebelum lanjut ke tahap berikutnya. Pemilihan metode ini diharapkan dapat memaksimalkan hasil pada setiap tahapan pengembangan serta menghasilkan sistem akhir yang sesuai dengan rancangan awal dengan dokumentasi yang lengkap setiap tahapnya.



Gambar 1 Gambar Pendekatan *Waterfall*

Gambar 1 menunjukkan bahwa ada 5 tahap yang akan dilewati pada pendekatan waterfall. Berikut adalah tahap-tahap serta penjelasannya:

1. Requirement Analysis

Menurut Senarath [6], Tahap *Requirement Analysis* adalah tahap yang dilakukan untuk menganalisa dan mengetahui kebutuhan dari aplikasi yang ingin dirancang seperti desain tampilan, fungsi-fungsi pada aplikasi, spesifikasi dari aplikasi dan sebagainya. Pada aplikasi administrasi kantor berbasis *Android*, tahap ini dilakukan dengan survei dan wawancara langsung bersama beberapa karyawan di kantor PT Ronadamar Sejahtera untuk mendapatkan informasi seperti kebutuhan fitur dan spesifikasi perangkat.

2. Design

Tahap selanjutnya adalah tahap desain yang menurut Ramadan et al. [7] adalah tahap yang dilakukan untuk mengubah hasil analisa kebutuhan dari tahap sebelumnya menjadi sebuah rancangan atau blueprint dari sebelum tahap pengembangan dimulai. Tahap ini dilakukan untuk merancang arsitektur sistem dari aplikasi dengan membuat beberapa *Unified Modeling Language*. *Unified Modeling Language* yang akan dibuat adalah *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*. Selain arsitektur sistem, perancangan basis data juga dilakukan pada tahap ini.

3. Development

Penelitian oleh Ridwan et al. [8] menjelaskan bahwa tahap ini merupakan tahap pengembangan aplikasi dengan pembuatan kode dan membuat rancangan desain dari tahap-tahap sebelumnya. Proses ini

mencakup pembuatan kode program sesuai dengan desain sistem dan antarmuka pengguna, serta pengolahan data untuk mendukung transaksi dan fungsi-fungsi utama dalam aplikasi.

4. Testing

Tahap testing atau pengujian merupakan tahap terakhir sebelum meluncurkan aplikasi yang dibuat ke pengguna. Menurut Gupta et al. [9], Tahap ini adalah tahap yang dilakukan untuk menguji apakah aplikasi yang dibuat sudah sesuai dengan kebutuhan pengguna.

5. Deployment and Maintenance

Berdasarkan penelitian yang dilakukan oleh Risener [10], tahap *Deployment* adalah tahap untuk meluncurkan aplikasi yang sudah dibuat dan sesuai dengan spesifikasi yang diharapkan pengguna untuk digunakan secara publik. Bersama dengan tahap *Deployment* adalah tahap *Maintenance* yang dilakukan untuk menjaga dan memelihara aplikasi secara berkala untuk memastikan aplikasi dapat digunakan secara berkelanjutan dan tetap relevan dengan kebutuhan pengguna di masa mendatang.

3. Hasil dan Pembahasan

Perancangan aplikasi administrasi kantor berbasis *Android* menghasilkan rancangan-rancangan yang dibutuhkan untuk pengembangan aplikasi. Rancangan yang dihasilkan berupa *Unified Modeling Language (UML)*, perancangan basis data, dan perancangan *Wireframe*. Rancangan-rancangan tersebut telah disesuaikan untuk kebutuhan dan kenyamanan pengguna sehingga mempermudah pengembangan selanjutnya. Perancangan ini akan diimplementasikan menggunakan *framework Flutter* dengan bahasa pemrograman *dart* agar hasil implementasi bisa lebih cepat selesai dan lebih maksimal.

3.1 Fitur Aplikasi Administrasi Kantor Berbasis Android

Sebelum melakukan perancangan aplikasi, fitur-fitur atau proses yang ingin diimplementasikan harus ditentukan terlebih dahulu. Perancangan aplikasi administrasi kantor berbasis *Android* dilatarbelakangi oleh keberadaan website Integrasi-RDS dari PT Ronadamar Sejahtera yang telah digunakan untuk menangani seluruh proses administrasi di semua kantor. Beberapa fitur yang tersedia pada website tersebut akan dikembangkan dalam versi *Android* agar dapat diakses dengan lebih mudah melalui ponsel atau tablet oleh para karyawan. **Tabel 1** menampilkan fitur-fitur yang akan dirancang dalam aplikasi sesuai dengan kebutuhan pengguna.

Tabel 1 Daftar Fitur Pada Aplikasi Administrasi Kantor Berbasis Android

No	Fitur Aplikasi Administrasi Kantor Berbasis Android	Aktor/Pengguna
1.	Bantuan IT	Seluruh Karyawan, Staf IT
2.	Izin	Seluruh Karyawan, Admin
3.	Cuti	Seluruh Karyawan, Admin
4.	Permintaan ATK	Seluruh Karyawan
5.	Sales Order	Staf Sales
6.	Absen Mandiri	Seluruh Karyawan

3.1.1 Fitur Bantuan IT

Fitur ini merupakan fitur yang digunakan untuk melaporkan masalah-masalah terkait IT di kantor. Bantuan IT dapat digunakan oleh seluruh karyawan dan staf IT sebagai pengelola hasil laporan permasalahan IT yang terjadi.

Cara kerja fitur bantuan IT pada aplikasi yaitu :

1. Karyawan :

- a. Karyawan harus login terlebih dahulu dan pergi ke halaman daftar tiket dengan menekan tombol bantuan pada halaman utama.
- b. Karyawan dapat membuat laporan masalah terkait IT dengan membuat tiket terlebih dahulu dan memberikan informasi yang dibutuhkan pada formulir pelaporan masalah IT.
- c. Karyawan juga dapat mengubah informasi tiket atau menghapus tiket yang sudah dibuat pada tiket yang belum berstatus selesai.

2. Staf IT :

- a. Staf IT dapat menentukan prioritas dari tiket yang dibuat.
- b. Staf IT dapat mengubah status dari tiket yang dibuat. Kemudian akan diminta memberikan komentar atau feedback melalui pop-up input pada aplikasi.

3.1.2 Fitur Izin

Fitur cuti merupakan fitur yang digunakan untuk mengajukan pengambilan cuti karyawan. Fitur ini dapat digunakan oleh seluruh karyawan. Selain itu, fitur ini juga memiliki sistem *approval* yang hanya dapat digunakan oleh admin.

Cara kerja fitur cuti pada aplikasi yaitu :

1. Karyawan :

- a. Karyawan harus login terlebih dahulu dan pergi ke halaman daftar cuti dengan menekan tombol cuti pada halaman utama.
- b. Karyawan dapat mengajukan cuti dengan menekan tombol buat cuti dan memberikan informasi yang dibutuhkan pada formulir pengajuan cuti.
- c. Karyawan juga dapat mengubah informasi cuti atau menghapus cuti yang diajukan selama status cuti bukan selesai.

2. Admin :

- a. Admin dapat meninjau cuti yang diajukan dengan menekan tombol cuti yang ingin ditinjau. Tombol tersebut akan membuka halaman detail pengajuan cuti.
- b. Admin dapat menerima atau menolak pengajuan cuti dengan menekan tombol terima atau tombol tolak pada halaman detail pengajuan cuti.
- c. Pengajuan cuti akan diproses menggunakan sistem approval berlapis, di mana pengajuan tersebut harus disetujui oleh seluruh admin yang berwenang sesuai dengan level-nya masing-masing. Apabila terdapat satu penolakan, maka pengajuan cuti akan dianggap ditolak.

3.1.3 Fitur Cuti

Fitur izin adalah fitur yang dibuat untuk membuat permohonan izin karyawan. Izin ini dapat berupa izin pergi sebentar, izin pulang awal, izin tidak masuk, dan izin tidak memakai seragam. Fitur ini dapat digunakan oleh seluruh karyawan. Sistem approval juga dimiliki pada fitur ini dan hanya admin yang bisa mengakses sistem tersebut.

Cara kerja fitur izin pada aplikasi yaitu :

1. Karyawan :

- a. Karyawan harus login terlebih dahulu dan pergi ke halaman daftar izin dengan menekan tombol izin pada halaman utama.
- b. Karyawan dapat membuat permohonan izin dengan menekan tombol buat izin dan memberikan informasi yang dibutuhkan pada formulir permohonan izin.
- c. Karyawan juga dapat mengubah informasi permohonan izin atau menghapus permohonan

izin yang dibuat selama status permohonan izin bukan selesai.

2. Admin :

- a. Admin dapat meninjau permohonan izin yang diajukan dengan menekan tombol permohonan izin yang ingin ditinjau. Tombol tersebut akan membuka halaman detail permohonan izin.
- b. Admin dapat menerima atau menolak permohonan izin dengan menekan tombol terima atau tombol tolak pada halaman detail permohonan izin.
- c. Permohonan izin akan diproses menggunakan sistem approval berlapis, di mana permohonan tersebut harus disetujui oleh seluruh admin yang berwenang sesuai dengan level-nya masing-masing. Apabila terdapat satu penolakan, maka permohonan izin akan dianggap ditolak.

3.1.4 Fitur Permintaan ATK

Fitur ini merupakan fitur yang dapat digunakan karyawan untuk membuat pengajuan permintaan alat tulis kantor (ATK). Fitur ini tidak memiliki sistem approval. Maka dari itu, fitur ini dibuat agar permintaan barang ATK di kantor dapat tercatat dengan rapih di satu database.

Cara kerja fitur pengajuan permintaan ATK yaitu :

1. Karyawan harus login terlebih dahulu dan pergi ke halaman daftar pengajuan permintaan ATK dengan menekan tombol permintaan ATK pada halaman utama.
2. Karyawan dapat membuat pengajuan permintaan ATK pada halaman daftar pengajuan permintaan ATK dan mengisi formulir permintaan ATK.
3. Karyawan dapat menambah barang baru pada satu pengajuan yang sama dengan mengisi formulir tambah barang pada halaman detail pengajuan permintaan ATK.
4. Karyawan juga dapat mengubah jumlah barang ATK yang diminta atau menghapus barang ATK pada suatu pengajuan.
5. Karyawan dapat menghapus pengajuan yang dibuat.

3.1.5 Fitur Sales Order

Selanjutnya adalah fitur untuk mengelola Sales Order yang digunakan hanya oleh staf Sales Order. Fitur ini dibuat agar proses membuat, mengubah, dan menghapus Sales Order menjadi lebih praktis. Approval untuk Sales Order juga dibutuhkan sehingga admin juga terlibat pada fitur ini. Sebuah Sales Order akan diterima jika seluruh admin menyetujui Sales Order tersebut, Namun jika salah satu admin menolak maka Sales Order tersebut tidak akan disetujui.

Cara kerja fitur pengelolaan Sales Order yaitu :

1. Staf Sales :

- Staf Sales harus login terlebih dahulu dan pergi ke halaman daftar Sales Order dengan menekan tombol Sales Order pada halaman utama.
- Staf Sales dapat membuat Sales Order pada halaman daftar Sales Order dan mengisi formulir pembuatan Sales Order.
- Staf Sales juga dapat mengubah informasi Sales Order atau menghapusnya.

2. Admin :

- Admin dapat meninjau Sales Order yang diajukan dengan menekan tombol Sales Order yang ingin ditinjau. Tombol tersebut akan membuka halaman detil Sales Order.
- Admin dapat menerima atau menolak Sales Order dengan menekan tombol terima atau tombol tolak pada halaman detil Sales Order.
- Sales Order akan diproses menggunakan sistem approval berlapis, di mana Sales Order tersebut harus disetujui oleh seluruh admin yang berwenang sesuai dengan level-nya masing-masing. Apabila terdapat satu penolakan, maka Sales Order akan dianggap ditolak.

3.1.6 Fitur Absen Mandiri

Fitur terakhir adalah fitur absen mandiri yang digunakan oleh seluruh karyawan untuk melakukan absen dimana-pun kapan-pun. Fitur ini bekerja dengan mengecek lokasi karyawan saat melakukan absen saat check-in atau check-out.

Cara kerja fitur absen mandiri yaitu :

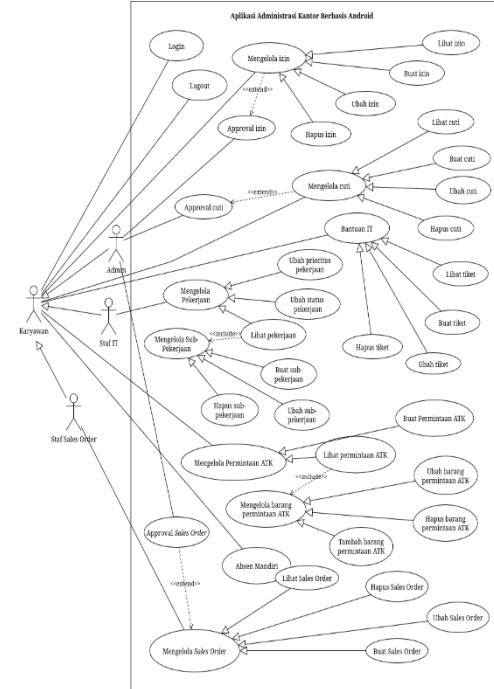
- Karyawan harus login terlebih dahulu dan pergi ke halaman absen dengan menekan tombol absen pada halaman utama untuk melakukan check-in.
- Sistem akan melakukan pengecekan pada lokasi karyawan saat ini dan menyimpannya ke data absen.
- Jika sistem tidak mendeteksi check-in maka karyawan tidak bisa melakukan check-out.
- Karyawan dapat melakukan check-out pada halaman yang sama dengan check-in.

3.2 Unified Modeling Languange(UML)

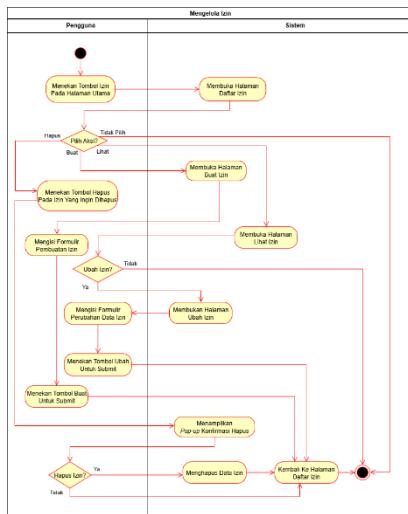
Perancangan aplikasi administrasi kantor berbasis android akan dilakukan dengan membuat *Unified Modeling Language (UML)*. *Unified Modeling Language (UML)* adalah bahasa pemodelan standar yang digunakan untuk membuat dokumentasi perancangan model sistem suatu perangkat lunak atau perangkat keras yang bersifat *Object Oriented* sehingga memudahkan dalam menggambarkan fungsi dan desain sistem secara

jelas dan terstruktur [11]. Sebelum menggunakan *UML*, pemahaman mengenai dasar model konseptual sangat diperlukan karena merupakan fondasi yang akan digunakan pada *UML*. Model konseptual akan menjelaskan objek apa saja yang terlibat dan bagaimana objek tersebut saling berinteraksi. Model ini dibangun berdasarkan konsep-konsep *Object Oriented* yang meliputi objek, kelas, abstraksi, inheritance, polimorfisme, dan enkapsulasi. Dengan menggunakan konsep tersebut, Penggunaan *UML* untuk mendeskripsikan struktur dan perilaku sistem menjadi jelas dan terstruktur.

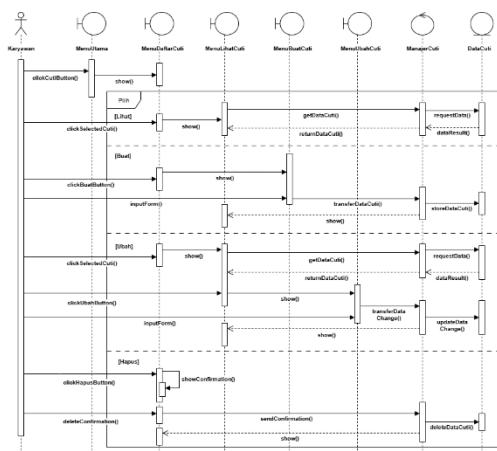
Buku yang ditulis oleh Alex Nordeen [11] juga menjelaskan bahwa *UML* akan digambarkan dalam bentuk diagram yang terbagi menjadi tiga kategori yaitu *Structural Diagram*, *Behavioral Diagram* dan *Interaction Diagram*. Dalam perancangan aplikasi administrasi kantor berbasis *Android*, ketiga kategori diagram akan disusun untuk memastikan bahwa seluruh aspek yang dibutuhkan tercakup, sehingga rancangan sistem dapat digambarkan dengan jelas dan terstruktur. Structural Diagram yang akan digunakan pada perancangan adalah Class Diagram yang dapat dilihat pada **Gambar 5**, kemudian Behavioral Diagram yang digunakan adalah Use Case Diagram yang dapat dilihat pada **Gambar 2** dengan scenario serta Activity Diagram yang dapat dilihat pada **Gambar 3**. Untuk Interaction Diagram, diagram yang digunakan adalah Sequence Diagram yang dapat dilihat pada **Gambar 4**.



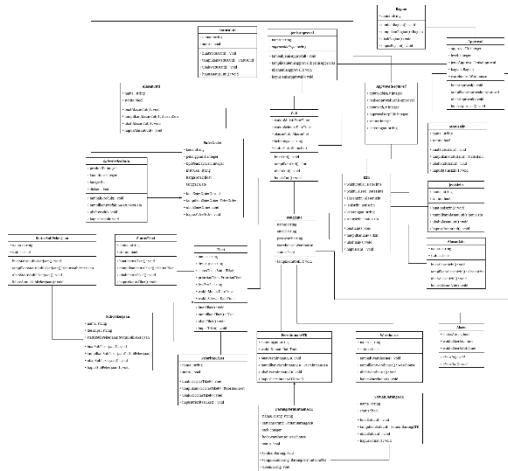
Gambar 2 Use Case Diagram untuk aplikasi administrasi kantor berbasis *Android*



Gambar 3 *Activity Diagram* untuk mengelola izin



Gambar 4 *Sequence Diagram* untuk mengelola cuti

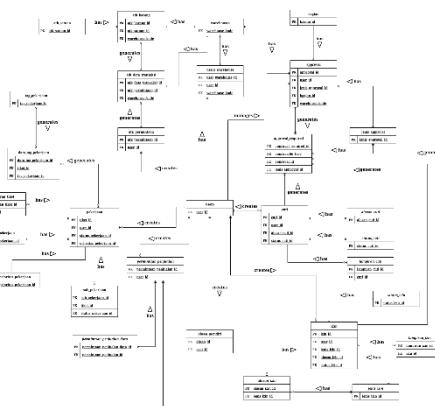


Gambar 5 *Class Diagram* untuk aplikasi administrasi kantor berbasis *Android*

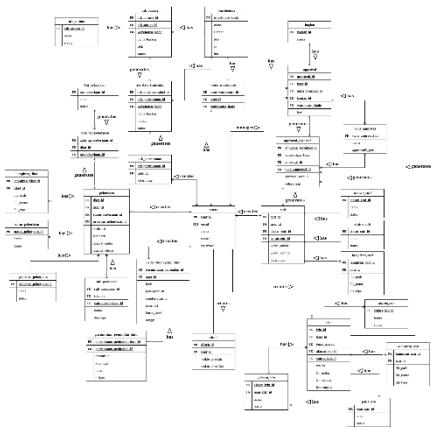
3.3 Perancangan Basis Data

Seiring dengan kemajuan teknologi, metode penyimpanan data atau informasi di perusahaan juga mengalami perkembangan. Dari pencatatan manual di atas kertas, kini beralih ke penyimpanan virtual yang lebih praktis dan efisien. PT Ronadamar Sejahtera sendiri telah menerapkan penyimpanan data atau informasi secara virtual menggunakan sebuah Database yang dapat diakses melalui Website Integrasi-RDS. Berdasarkan buku yang ditulis oleh Connolly dan Begg [12], *Database* adalah kumpulan data yang saling berhubungan, yang pengelolaannya dilakukan oleh sebuah perangkat lunak yang disebut *Database Management System (DBMS)*. *Database Management System (DBMS)* ini merupakan perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara dan mengontrol akses ke *Database*.

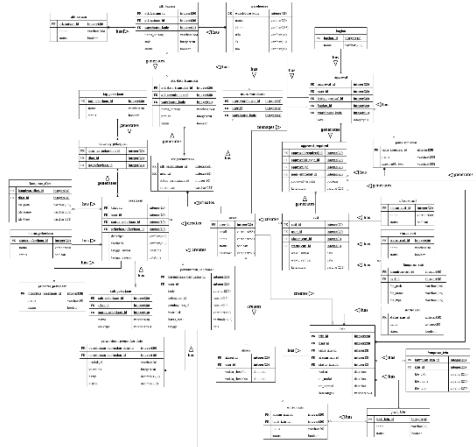
Buku yang ditulis oleh Connolly dan Begg [12] juga menjelaskan bahwa perancangan *Database* adalah proses merancang sistem *Database* yang mampu mendukung misi dan tujuan suatu organisasi. Pada perancangan *Database* terdapat beberapa pendekatan yang dapat digunakan. Perancangan *Database* terdiri dari tiga tahap yaitu pembuatan conceptual database yang dapat dilihat pada **Gambar 6**, logical database yang dapat dilihat pada **Gambar 7**, dan physical database yang dapat dilihat pada **Gambar 8**. Ketiga tahap ini merupakan tahap penting dalam merancang *database* karena tahap-tahap ini akan mendefinisikan entitas, relasi antar data, attribut pada data, dan bagaimana kumpulan data tersebut berinteraksi.



Gambar 6 *Conceptual Database* untuk aplikasi administrasi kantor berbasis *Android*



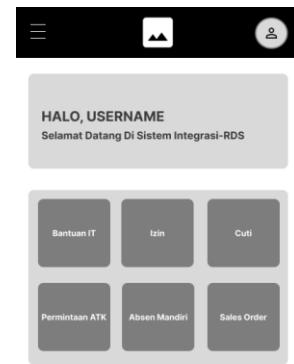
Gambar 7 Logical Database untuk aplikasi administrasi kantor berbasis Android



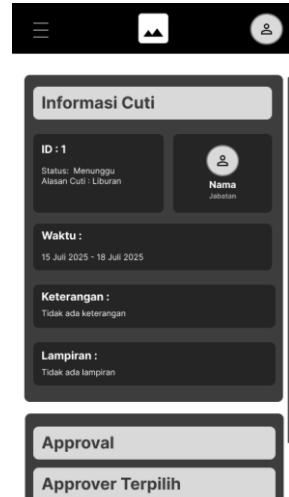
Gambar 8 Logical Database untuk aplikasi administrasi kantor berbasis Android

3.4 Perancangan Antarmuka Sistem

Perancangan antarmuka sistem dari aplikasi administrasi kantor berbasis *Android* dibuat dengan menggunakan metode *Design Thinking*. Menurut Lais *et al.* [13], metode *Design Thinking* adalah metode yang berfokus pada penyelesaian masalah yang disesuaikan dengan kebutuhan pengguna melalui survei atau wawancara dengan tujuan untuk mempermudah identifikasi masalah dan membuat proses pengembangan menjadi lebih mudah dan sesuai dengan kebutuhan pengguna. Perancangan antarmuka sistem pada aplikasi administrasi kantor berbasis *Android* menghasilkan rancangan desain dalam bentuk *Wireframe* seperti pada **Gambar 9** yang menampilkan *wireframe* halaman utama dan **Gambar 10** yang menampilkan halaman lihat cuti.



Gambar 9 Wireframe untuk halaman utama



Gambar 10 Wireframe untuk halaman lihat cuti

3.5 Hasil Pengujian

Perancangan aplikasi administrasi kantor berbasis *android* pada PT Ronadamar Sejahtera telah melalui dua tahap pengujian. Pengujian pertama yang dilakukan adalah *Blackbox*. Pengujian ini diuji oleh perancang aplikasi dengan menguji seluruh fungsi fitur dan alur setiap fiturnya. Pengujian ini memastikan keseluruhan aplikasi sudah siap untuk memasuki tahap selanjutnya bersama pengguna aplikasi. Berdasarkan pengujian *Blackbox*, aplikasi berjalan sesuai harapan perancang dengan seluruh fitur yang berfungsi dengan baik. Oleh karena itu, aplikasi akan lanjut ke tahap pengujian selanjutnya yang bernama *User Acceptance Test (UAT)*.

User Acceptance Test (UAT) akan dilakukan bersama pengguna yang terdiri dari beberapa karyawan di kantor cabang Jakarta dari PT Ronadamar Sejahtera. *Staf IT*, *staf sales*, *staf HR*, dan *admin sales* merupakan pengguna dari aplikasi yang telah dirancang dan telah menguji sesuai dengan fitur yang akan digunakan oleh masing-masing penguji nantinya. Berdasarkan *User Acceptance Test (UAT)* yang telah dilakukan, perancangan aplikasi

administrasi kantor berbasis *android* dapat membantu meningkatkan efektifitas dan efisiensi proses administrasi di kantor yang sebelumnya terasa kurang nyaman digunakan pada *website* yang tampilan belum optimal. Beberapa pengujian juga memberikan *feedback* positif mengenai fitur absen mandiri yang akan sangat membantu karyawan dalam melakukan absensi terutama *staf sales* yang terkadang tidak bisa absen di kantor karena sedang melakukan tugas kerja di luar kantor.

Secara keseluruhan, hasil dari kedua pengujian telah membuktikan bahwa perancangan aplikasi administrasi kantor berbasis *android* menghasilkan perancangan yang efektif dapat digunakan pada PT Ronadamar Sejahtera sesuai dengan kebutuhan para karyawan atau pengguna. Selain itu, *feedback* dari para pengujian pada *User Acceptance Test (UAT)* juga memberikan hasil yang positif karena menghadirkan fitur yang dibutuhkan, serta mengoptimalkan fitur-fitur yang sebelumnya telah tersedia dalam *website* namun terasa kurang optimal jika digunakan pada ponsel. Perancangan aplikasi administrasi kantor berbasis *android* akan berfungsi sesuai kebutuhan pada implementasinya, serta membantu meningkatkan efektifitas dan efisiensi proses administrasi dengan mengoptimalkan aplikasi berbasis *website* untuk digunakan pada ponsel.

4. Kesimpulan

Berdasarkan hasil perancangan yang telah dibuat, Perancangan aplikasi administrasi kantor berbasis Android dapat menjadi sebuah solusi atas permasalahan yang dimiliki pada aplikasi yang telah dibuat sebelumnya. Contoh yang dapat digunakan adalah fitur absen mandiri yang sebelumnya tidak tersedia dalam aplikasi berbasis website akan tersedia pada aplikasi berbasis android sesuai dengan rancangan yang dibuat. Dengan adanya fitur ini, absen karyawan semakin lebih fleksibel dan memudahkan karyawan untuk absen ketika sedang berhalangan hadir di kantor akibat urusan pekerjaan.

Selain contoh tersebut, hasil pengujian yang dilakukan bersama dengan pengguna aplikasi juga menunjukkan hasil yang positif sehingga pengembangan aplikasi berbasis android ini kedepannya akan sangat meningkatkan efektifitas dan efisiensi kerja karyawan. Dengan begitu dapat disimpulkan bahwa perancangan aplikasi administrasi kantor berbasis android ini akan sangat membantu seluruh karyawan di perusahaan untuk melakukan segala proses administrasi yang sebelumnya kurang efektif dan efisien jika dilakukan pada aplikasi berbasis Website.

REFERENSI

- [1] Y. D. Restanti, Administrasi Perkantoran. Malang: Media Nusa Creative, 2024, diakses pada 8 Agustus 2025.
- [2] N. F. Choirunnissa and N. Oktarina, "Peran digitalisasi dalam meningkatkan pelayanan administratif kantor," in Bookchapter Administrasi Perkantoran, 2025, vol. 1, pp. 77–95, diakses pada 8 Agustus 2025.

- [3] S. Beaty and N. Oktarina, "Digitalisasi administrasi kantor: Meningkatkan efisiensi dan kualitas layanan di era digital," in Bookchapter Administrasi Perkantoran, 2025, vol. 1, pp. 137–157, diakses pada 8 Agustus 2025.
- [4] S. Kundu, Modern Software Engineering Guidebook. Bpb Publications, 2024, diakses pada 8 Agustus 2025.
- [5] F. Tsui, O. Karam, and B. Bernal, Essentials of Software Engineering. Jones & Bartlett Learning, 2022, diakses pada 8 Agustus 2025.
- [6] U. S. Senarath, "Waterfall methodology, prototyping and agile development," ResearchGate, 2021, diakses pada 8 Agustus 2025.
- [7] A. R. Ramadan, A. Junaidi, and M. A. Azis, "Kai commuter employee development application using the waterfall method," Informatics and Software Engineering, vol. 1, no. 2, pp. 44–50, 2023, diakses pada 8 Agustus 2025.
- [8] I. Fitri, Benrahman, and M. Ridwan, "Rancang bangun marketplace berbasis website menggunakan metodologi systems development life cycle (sdlc) dengan model waterfall," Jurnal JTAK (Jurnal Teknologi Informasi dan Komunikasi), vol. 5, no. 2, 2021, diakses pada 8 Agustus 2025.
- [9] A. Gupta, A. Rawal, and Y. Barge, "Comparative study of different sdlc models," International Journal of Research in Applied Science and Engineering Technology, vol. 9, no. 11, pp. 73–80, 2021, diakses pada 8 Agustus 2025.
- [10] K. Risener, "A study of software development methodologies," ResearchGate, 2022, diakses pada 8 Agustus 2025.
- [11] A. Nordeen, Learn UML in 24 Hours. Guru99, 2020, diakses pada 16 September 2025. [Online]. Available: <https://books.google.co.id/books?id=rBoGEAAAQBAJ>
- [12] T. M. Connolly and C. E. Begg, Database Systems: A Practical Approach to Design, Implementation, and Management, 6th ed. Pearson Education Limited, 2015, diakses Pada 13 September 2025.
- [13] N. Lais, R. Chandra, M. Hamdandi, F. Bachtiar, D. A. Sastika, and M. R. Pribadi, "Perancangan UI/UX pada aplikasi bapakkost dengan metode design thinking," in MDP Student Conference, vol. 1, no. 1, 2022, pp. 392–397, diakses pada 10 Agustus 2025.

Nicolas, mahasiswa S1 yang sedang menjalankan studi pada program studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara.

Tony, memperoleh gelar S.Kom. pada tahun 2005 dari Universitas Tarumanagara, M.Kom. pada tahun 2010 dari Universitas Indonesia, dan Ph.D. pada tahun 2021 dari Curtin University. Saat ini sebagai staf pengajar di Fakultas Teknologi Informatika, Universitas Tarumanagara.