

APLIKASI KASIR JASA PANGKAS RAMBUT BERBASIS MOBILE

Shilna Paradisa¹, Azmi Taqiyuddin², Rizqi Ahmad Fauzan³

^{1,2,3,4}Informatika, Fakultas Teknik, Universitas Jenderal Soedirman, Indonesia

NIM : ¹H1D020007, ²H1D020046, ³H1D020085

Email: ¹shilna.paradisa@mhs.unsoed.ac.id, ²azmi.taqiyuddin@mhs.unsoed.ac.id, ³rizqi.fauzan@mhs.unsoed.ac.id
(Artikel dikirimkan tanggal : dd mmm yyyy)

Abstrak

Hampir semua pekerjaan yang ada telah dipermudah dengan adanya teknologi yang semakin berkembang seiring berjalannya waktu. Salah satunya yaitu adanya perkembangan teknologi *smartphone* yang sangat pesat dan juga banyak penggunaannya. Dengan adanya teknologi tersebut, profesi yang ada kini dapat lebih berkembang dan lebih inovatif lagi dalam pelaksanaannya. Salah satunya yaitu profesi pangkas rambut yang banyak dibutuhkan oleh orang saat ini dari berbagai usia dan kalangan. Dalam prosesnya, transaksi pangkas rambut membutuhkan adanya aplikasi pengelolaan kasir yang dapat menyimpan data transaksi pelanggan untuk mengatasi hilangnya data data transaksi. Oleh karena itu, penulis membuat aplikasi kasir jasa pangkas rambut ini dengan harapan aplikasi ini dapat mempermudah segala aktivitas yang dilakukan khususnya dalam pemesanan jasa pangkas rambut yang mudah agar lebih efisien dan efektif dengan pemanfaatan teknologi yang ada. Pembuatan aplikasi kasir jasa pangkas rambut ini menggunakan bahasa pemrograman java dan *database* akan disimpan dalam *firebase*. *Firebase* akan digunakan untuk menyimpan data yang diinputkan kedalam aplikasi yang dibuat.

Kata kunci: Aplikasi android, Pangkas Rambut, Kasir, Teknologi.

MOBILE BASED BARBERSHOP CASHIER APPLICATION

Abstract

Almost all existing jobs have been made easier by the existence of technology that is growing over time. One of them is the development of smartphone technology which is very rapid and also many users. With this technology, the existing profession can now be more developed and more innovative in its implementation. One of them is the profession of barbershop which is needed by people of all ages and backgrounds. In the process, barbershop transactions require a cashier management application that can store customer transaction data to overcome the loss of transaction data. Therefore, the author made this barbershop cashier application with the hope that this application can facilitate all activities carried out, especially in ordering easy barber services to make it more efficient and effective by utilizing existing technology. The making of this barbershop cashier application uses the Java programming language and the database will be stored in Firebase. Firebase will be used to store data that is input into the application that is made.

Keywords: Android Apps, Barbershop, Cashier, Technology.

1. PENDAHULUAN

Seiring dengan perkembangan zaman saat ini, teknologi telah merambat di berbagai aspek kehidupan kita, tak terkecuali dengan pekerjaan-pekerjaan yang ada. Hampir semua pekerjaan yang ada telah dipermudah dengan adanya teknologi yang semakin berkembang seiring berjalannya waktu. Salah satunya yaitu adanya perkembangan teknologi *smarthphone* yang sangat pesat dan juga banyak penggunaannya. Banyaknya pengguna *smartphone* dikarenakan harganya yang terjangkau dan biaya komunikasinya yang juga semakin murah, membuat para peneliti memutuskan untuk membuat sebuah prototipe sistem mobile ebisnis berbasis android[1].

Menurut Abidin, populasi aplikasi yang dibuat pada sistem operasi android semakin meningkat[2]. Seiring perkembangan teknologi *mobile*, banyak jasa-jasa online yang kian muncul seperti jasa antar makanan online, antar penumpang, dan sebagainya. Dengan adanya teknologi tersebut, profesi yang ada kini dapat lebih berkembang dan lebih inovatif lagi dalam pelaksanaannya.

Profesi pangkas rambut merupakan profesi yang banyak dibutuhkan oleh berbagai kalangan saat ini. Pangkas rambut sangat dibutuhkan guna menjaga penampilan dari tiap individu. Di zaman modern ini, berpenampilan rapi dan menarik menjadi kebutuhan sehari-hari bagi masyarakat di segala kalangan. Baiknya penampilan juga dapat

meningkatkan kepercayaan diri, menarik perhatian orang atau lawan jenis dan meningkatkan kepercayaan orang lain[3]. Oleh karenanya, usaha pangkas rambut ini dapat menjadi sebuah profesi yang akan terus berkelanjutan untuk kedepannya.

Walaupun banyak kebutuhan pangkas rambut ini, sayangnya masih sangat jarang penyedia jasa layanan pangkas rambut yang menggunakan aplikasi kasir dalam proses transaksi. Data Para pelanggan tidak tersimpan dengan baik, dan tidak dapat diketahui keuntungan yang diperoleh untuk setiap harinya. Hal tersebut tentu kurang efektif dilakukan sehingga menimbulkan risiko kehilangan data pelanggan dan proses transaksi tidak efisien[3].

Adanya fenomena teknologi informasi ini harus dicermati dengan baik, terutama untuk dapat mempermudah segala aktivitas yang dilakukan oleh manusia khususnya dalam pemesanan jasa pangkas rambut yang mudah dan cepat agar lebih efisien dan efektif[4]. Oleh karena itu, penulis membuat aplikasi kasir jasa pangkas rambut ini dengan harapan aplikasi yang dibangun dapat mempermudah dan mempercepat proses transaksi pada jasa pangkas rambut, dengan adanya aplikasi ini data pelanggan akan tersimpan rapi dan mempermudah untuk mengetahui keuntungan yang didapatkan setiap harinya dari adanya jasa pangkas rambut.

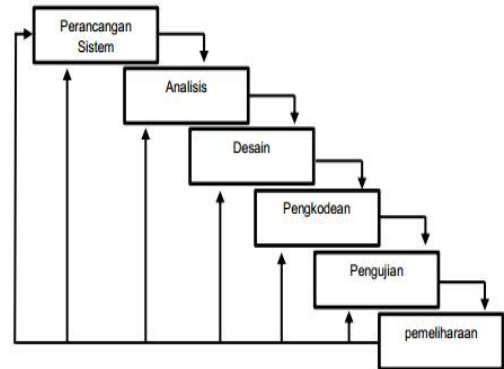
2. METODE

2.1 Model Waterfall

Metode yang digunakan pada pengembangan aplikasi jasa pangkas rambut berbasis mobile ini memakai model *waterfall*. Menurut pressman, model *waterfall* adalah model unggul yang bersifat analitis, berturut-turut dalam menciptakan *software* [5]. Model ini melakukan pendekatan secara sistematis dan berurutan. Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan [6]. Alasan menggunakan metode *waterfall* karena metode ini tahapan dan juga urutan dari metode yang dilakukan berurutan dan berkelanjutan, seperti layaknya sebuah air terjun. [7].

Metode *waterfall* merupakan metode pengembangan perangkat lunak tertua sebab sifatnya yang natural. Metode *waterfall* merupakan pendekatan SDLC paling awal yang digunakan untuk pengembangan perangkat lunak. Urutan dalam Metode *waterfall* bersifat serial yang dimulai dari proses perencanaan, analisa, desain, dan implementasi pada sistem. Metode ini dilakukan dengan pendekatan yang sistematis, mulai dari tahap kebutuhan sistem lalu menuju ke tahap *analysis*, *design*, *coding*, *testing/verification*, dan *maintenance*. Langkah demi langkah yang dilalui harus diselesaikan satu per satu (tidak dapat meloncat ke tahap berikutnya) dan berjalan secara berurutan, oleh karena itu disebut *waterfall* (Air Terjun)[8]. Roger S.Pressman memecah model ini menjadi 6 tahapan

meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya, berikut gambar dan penjelasan dari tahap-tahap yang dilakukan di dalam model menurut pressman:



Gambar 1. Tahapan metode waterfall

Model *waterfall* diawali dengan perancangan sistem. Pada bagian ini aplikasi yang akan dibuat diawali dengan mencari kebutuhan dari keseluruhan sistem yang akan diaplikasikan kedalam bentuk *software*[9]. Selanjutnya ada tahap analisis untuk mengetahui sifat dari program yang akan dibuat misalnya fungsi yang akan dibutuhkan, user interface, dan lainnya[10]. Yang ketiga ada tahap desain yang digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi dalam bentuk "*blueprint*" *software* sebelum *coding* dimulai[11]. Tahapan yang keempat yaitu pengkodean atau proses *coding* untuk dapat dimengerti oleh sistem, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin yaitu ke dalam bahasa pemrograman melalui proses *coding*[12]. Tahap berikutnya yaitu *testing* atau pengujian. Pada tahap ini semua fungsi *software* harus diujicobakan, agar *software* bebas dari *error* dan hasilnya benar benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya. Tahap terakhir yaitu pemeliharaan, pada tahap ini *software* yang sudah jadi dioperasikan pengguna dan dilakukan pemeliharaan. Pemeliharaan memungkinkan pengembang untuk melakukan perbaikan atas kesalahan yang tidak terdeteksi pada tahap-tahap sebelumnya[13]. Pemeliharaan meliputi perbaikan kesalahan, perbaikan implementasi unit sistem, dan peningkatan dan penyesuaian sistem sesuai dengan kebutuhan.

2.2. Rancangan Aplikasi

Perancangan aplikasi kasir jasa pangkas rambut ini akan menggunakan bahasa pemrograman java dengan IDE Android Studio. Java merupakan bahasa pemrograman yang digunakan untuk membangun sebuah *website*, desktop, maupun *mobile programming*. Java merupakan bahasa

pemrograman yang bersifat umum, dan khusus didesain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena kemampuannya untuk memungkinkan aplikasi Java berjalan Beberapa *platform* sistem operasi yang berbeda[15].

2.3. Firebase Realtime Database

Aplikasi kasir pangkas rambut ini akan menggunakan Firebase sebagai database-nya. Firebase merupakan suatu model layanan yang berjalan di belakang layar dan menghubungkan *mobile app* ke *cloud storage*[16]. Firebase akan digunakan untuk menyimpan data yang diinputkan kedalam aplikasi yang dibuat. Setelah proses pemrograman dilakukan akan dihubungkan dengan *database* untuk bisa menjalankan aplikasi.

3. HASIL DAN PEMBAHASAN

3.1 Analisis Kebutuhan

Proses analisis kebutuhan yang dilakukan mencakup pencarian studi literatur terkait, wawancara secara langsung dengan calon customer, dan juga observasi secara langsung tentang fitur apa saja yang dibutuhkan untuk aplikasi KangCukur. Studi literatur diambil dari internet yang memiliki pembahasan cukup relevan untuk dijadikan sebagai referensi dalam membangun aplikasi KangCukur. Setelah menemukan studi literatur terkait, selanjutnya penulis melakukan analisis kebutuhan pengguna dengan dua cara, yang pertama yaitu dengan observasi secara langsung yaitu peneliti mengamati dan menganalisis kondisi lingkungan sekitar dan melakukan diskusi mengenai fitur apa saja yang nantinya dibutuhkan sesuai dengan kondisi lingkungan dan kebutuhan *customer*. Cara yang kedua yaitu dengan wawancara secara langsung kepada calon *customer*. Pada proses ini penulis melakukan wawancara dari berbagai kalangan dari remaja hingga orang dewasa yang hasilnya nantinya akan dijadikan sebagai acuan untuk membangun sebuah aplikasi android KangCukur berbasis mobile. Subsystem pada aplikasi KangCukur ini terdiri dari admin dan *user*. Admin berperan untuk mengelola data pada aplikasi KangCukur. Hak akses yang dimiliki admin yaitu *login*, melihat, mengedit, menghapus, menambahkan data karyawan, data paket harga yang tersedia, dan data pelanggan. Sementara itu *user* dapat melakukan *login*, melihat paket yang tersedia, dan melakukan transaksi jasa cukur rambut yang disediakan pada aplikasi KangCukur. Adapun identifikasi kebutuhan *user* disajikan pada tabel 1 berikut.

Tabel 1. Spesifikasi Kebutuhan Pengguna

No	Pengguna	Kebutuhan
1	Admin	Melakukan login Menambahkan karyawan

Melihat data karyawan
Mengedit data karyawan
Menghapus data karyawan
Melihat daftar paket
Menambahkan daftar paket
Mengedit data daftar paket
Menghapus data paket
Menambahkan pesanan
Mengedit pesanan
Menghapus pesanan
Melihat riwayat pesanan
Dapat Logout

Berdasarkan Tabel 1 diatas dapat diketahui mengenai hak akses apa saja yang dapat dilakukan oleh admin pada aplikasi KangCukur.. Selain adanya identifikasi kebutuhan pengguna, pada tahap analisis juga dibuat identifikasi kebutuhan sistem. Adapun identifikasi kebutuhan sistem disajikan pada Tabel 2 berikut.

Tabel 2. Spesifikasi Kebutuhan Sistem

No.	Pengguna	Fitur	Kebutuhan
1	Admin	Login	Sistem dapat melakukan login
		Main menu	Sistem dapat menampilkan menu yang berisi fitur data daftar karyawan, daftar paket. Masing masing menu dapat menyimpan data, menghapus data, serta mengedit data
		Daftar karyawan	Sistem dapat menghapus, menampilkan data karyawan yang di ubah dan dapat menambahkan data karyawan
		Daftar Paket	Sistem dapat menampilkan daftar paket yang sudah ditambahkan Sistem dapat menghapus data daftar paket

Order Paket Sistem dapat menampilkan data customer yang melakukan pemesanan. Sistem dapat memilih tanggal pemesanan. Sistem dapat memilih paket yang telah tersedia. Sistem dapat melihat riwayat order.

Riwayat Order Sistem dapat menampilkan daftar customer yang telah melakukan pemesanan.

Logout Sistem dapat melakukan logout dan akan mengarahkan ke halaman splash aplikasi KangCukur.

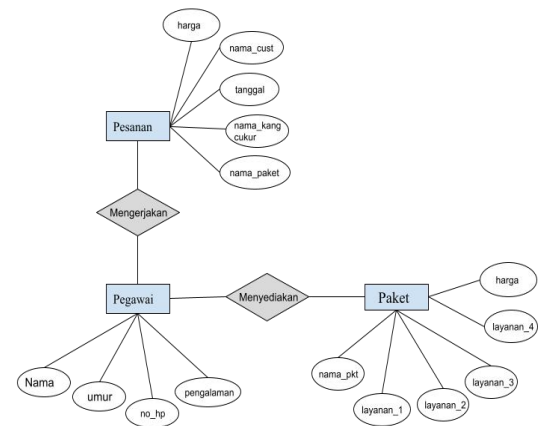
Berdasarkan Tabel 2 diatas dapat diketahui mengenai kebutuhan apa saja yang tersedia dan dapat dijalankan pada aplikasi KangCukur.

3.2 Desain

Pada tahap ini peneliti melakukan pembuatan rancangan desain pada aplikasi KangCukur yang meliputi ERD (*Entity Relationship Diagram*), UML (*Unified Modeling Language*), dan desain *interface*. Desain yang dibuat nantinya digunakan sebagai acuan untuk membuat aplikasi KangCukur.

3.2.1 ERD (*Entity Relationship Diagram*)

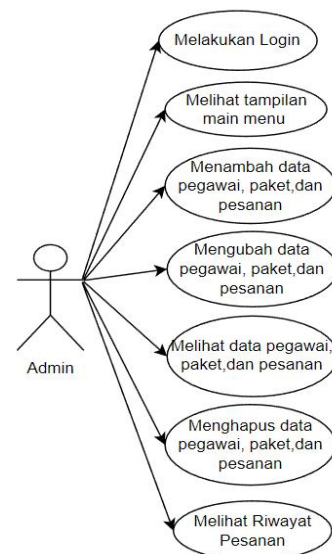
ERD (*Entity Relationship Diagram*) merupakan relasi antar entitas yang digambarkan melalui diagram. Dengan adanya ERD ini struktur logis dari suatu aplikasi data dapat digambarkan dengan jelas dan sederhana. ERD dari aplikasi KangCukur dapat dilihat pada gambar 2 berikut.



Gambar 2. *Entity Relationship Diagram* Aplikasi KangCukur

3.2.2 . UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) merupakan pengganti dari metode analisis berorientasi object dan *design* berorientasi *object* (OOAD&D/*object oriented analysis and design*) yang dimunculkan sekitar akhir tahun 80an dan awal tahun 90 an [16]. Penulis menggunakan *use case diagram* dalam pemodelan ini. Berikut merupakan UML pada aplikasi KangCukur



Gambar 3. *Use Case Diagram* Aplikasi KangCukur

Use case diagram adalah satu dari berbagai jenis diagram UML (*Unified Modelling Language*) yang menggambarkan hubungan interaksi antara sistem dan aktor. *Use case diagram* digunakan untuk memodelkan proses bisnis berdasarkan perspektif pengguna sistem

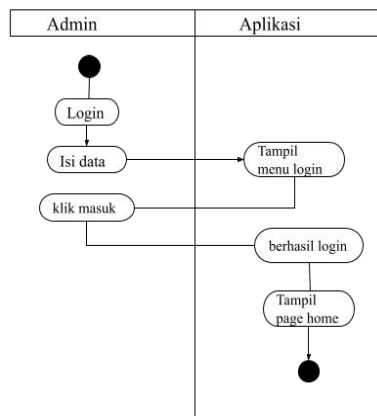
3.2.3 *Activity Diagram*

Activity diagram menggambarkan aktivitas-aktivitas yang dapat dilakukan

oleh *user*. Activity diagram adalah teknik menggambarkan alur kerja atau aktivitas dalam melakukan pekerjaan. *Activity diagram* yang tersedia pada aplikasi KangCukur terdiri dari 4 activity yaitu *activity* login, *activity* daftar kangcukur, *activity* daftar paket, dan *activity* tambah pesanan.

1. Activity Diagram Login

Pada *activity diagram login*, admin dapat langsung melakukan login dengan *email* dan *password* yang sudah didaftarkan pada *database*. Gambar 4 berikut ini menampilkan *activity diagram login* aplikasi KangCukur.

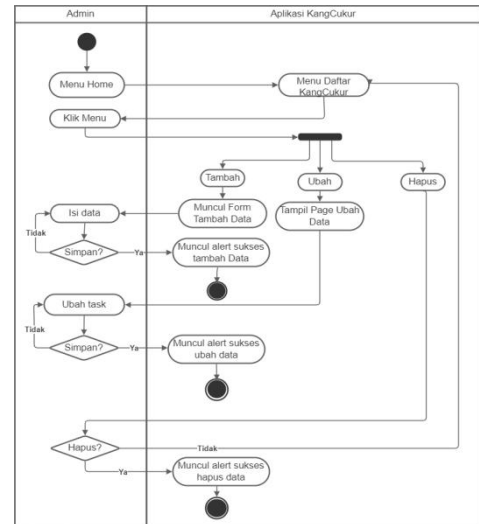


Gambar 4. Activity login KangCukur

2. Activity Diagram Daftar KangCukur

Gambar 4 diatas merupakan *Activity Diagram* dari aplikasi kasir KangCukur yang mempunyai satu subsistem saja yaitu admin yang dapat melakukan pengelolaan data tukang cukur, data paket, dan data pesanan yang tersedia. Data yang diinputkan admin akan disimpan pada *database* dan akan ditampilkan pada aplikasi sebagai sebuah data yang dapat diakses oleh admin.

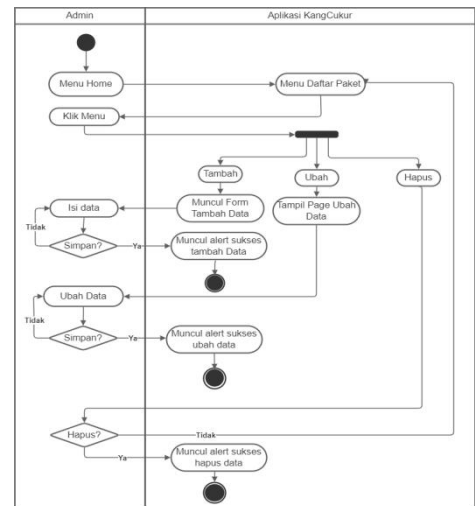
3. Activity Diagram Menu Daftar KangCukur



Gambar 5. Activity Menu Daftar KangCukur

Gambar 5 diatas merupakan *Activity Diagram* menu daftar KangCukur yang didalamnya terdapat beberapa fitur yaitu dapat menampilkan data, menambahkan data, mengedit data, serta menghapus data. Setelah data diinputkan akan muncul *alert* bahwa data tersebut berhasil ditambah, diubah, maupun dihapus sesuai aksi yang dilakukan.

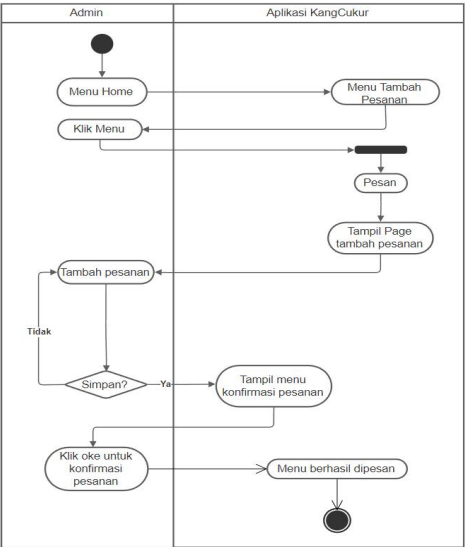
4. Activity Diagram Menu Daftar Paket



Gambar 6. Activity Menu Daftar Paket

Gambar 6 diatas merupakan *Activity Diagram* menu daftar paket yang didalamnya terdapat beberapa fitur yaitu dapat menampilkan data, menambahkan data, mengedit data, serta menghapus data. Setelah data diinputkan akan muncul *alert* bahwa data tersebut berhasil ditambah, diubah, maupun dihapus sesuai aksi yang dilakukan. Paket yang ditambahkan nantinya dapat diinputkan pada halaman pesanan.

5. Activity Diagram Menu Tamnah Pesanan



Gambar 7. Activity Menu Tambah Pesanan

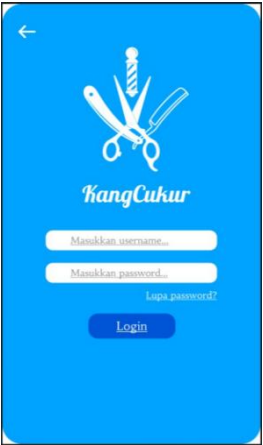
Gambar 7 diatas merupakan *Activity Diagram* menu tambah pesanan yang didalamnya terdapat fitur yaitu dapat menambahkan pesanan dalam aplikasi KangCukur. Admin dapat membuat pesanan dan mengonfirmasi pesanan. Data pesanan tersebut akan ditampilkan pada riwayat pesanan

3.2.4 Desain Interface

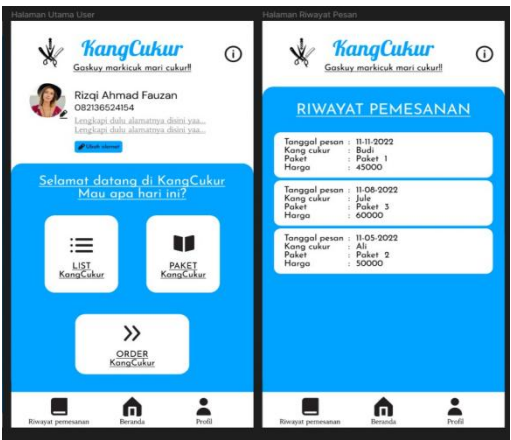
Pembuatan desain tampilan *interface* dibuat dengan *software* Figma. Berikut ini merupakan tampilan *user interface* dari aplikasi KangCukur disajikan pada gambar berikut.



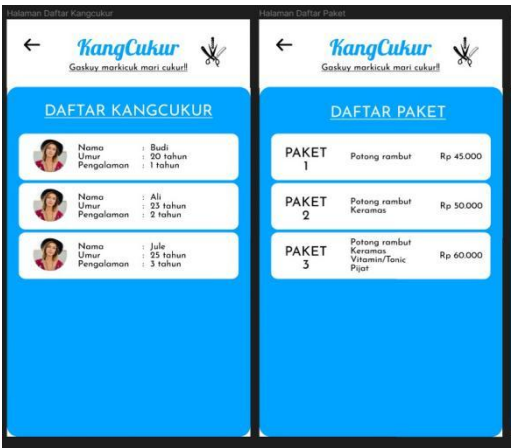
Gambar 8. Gambaran Tampilan *Splash*



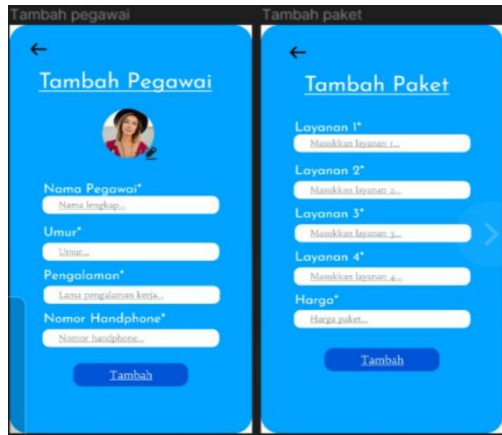
Gambar 9. Gambaran Tampilan *Login*



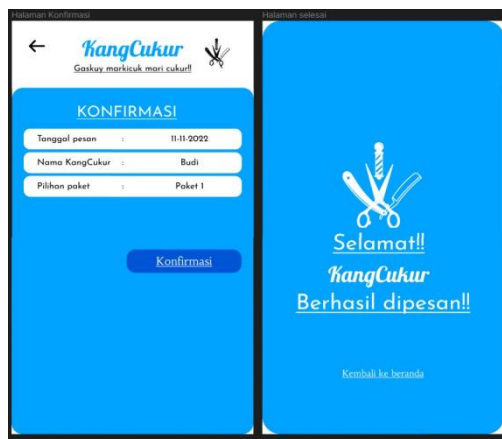
Gambar 10. Gambaran Tampilan Menu utama dan Riwayat Pesanan



Gambar 11. Gambaran Tampilan Daftar KangCukur dan Daftar Paket



Gambar 12. Gambaran Tampilan Tambah karyawan dan Tambah Paket



Gambar 13. Gambaran Tampilan Konfirmasi Pesanan

Gambar tersebut menunjukkan gambaran dari aplikasi KangCukur yang akan dibuat. Desain interface ini nantinya akan dilakukan proses implementasi coding pada android studio. Desain *interface* sangat penting mengingat perannya sebagai layanan komunikasi antara pengguna atau user terhadap sistem operasi yang ada. Dalam hal ini, antarmuka nantinya memberikan layanan yang berupa informasi sesuai kebutuhan pengguna.

3.3 Implementasi Aplikasi

Implementasi aplikasi merupakan tahap dari realisasi aplikasi yang sudah dibuat sekaligus hasil pengujian pada aplikasi. Aplikasi KangCukur ini terdiri dari satu sub sistem yaitu admin. Berikut dibawah ini penjelasan lebih lengkap mengenai fitur-fitur yang terdapat pada aplikasi kasir KangCukur berbasis *mobile*.

3.3.1 Menu *Splash*

Hal yang pertama kali muncul saat membuka aplikasi KangCukur yaitu menu *splash*. Pada menu ini terdapat logo aplikasi

dan akan mengarahkan pada menu selanjutnya yaitu menu *login*.



Gambar 14. Tampilan menu *splash*

3.3.2 Menu *Login*

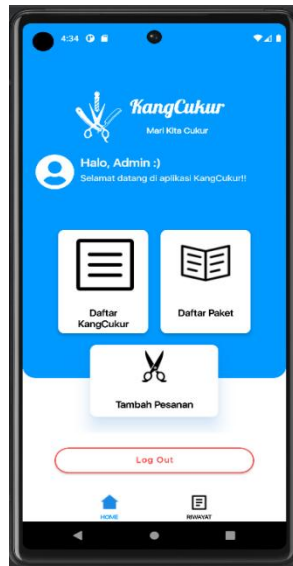
Pada menu ini terdapat data yang harus dimasukkan untuk login yaitu *email* dan *password*. Data ini diambil dari *database* yang telah dimasukkan sebelumnya. Admin harus melakukan *login* terlebih dahulu untuk dapat mengakses aplikasi KangCukur. Setelah melakukan login maka akan masuk ke dalam menu *home* aplikasi.



Gambar 15. Tampilan menu *login*

3.3.3 Menu *Utama*

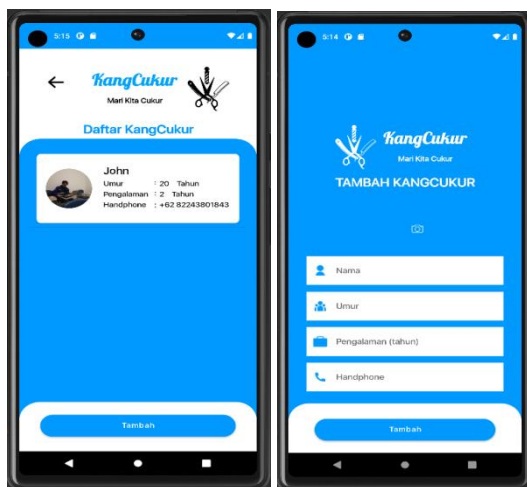
Pada menu ini merupakan menu utama dari admin. Terdapat tiga sub menu yaitu pegawai, paket, pesanan. Masing masing memiliki fitur untuk dapat melihat, menambahkan, menghapus, dan mengedit data.



Gambar 16. Tampilan Menu Utama

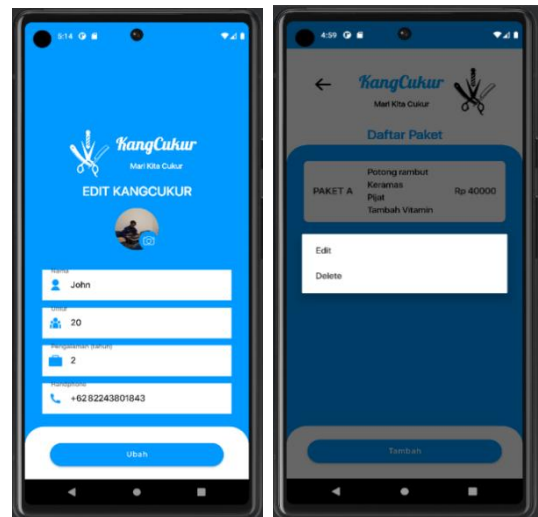
3.3.4 Menu Daftar KangCukur

Pada menu ini admin dapat menambahkan karyawan KangCukur yang berisi nama, umur, pengalaman, dan nomor *handphone*. Pada menu ini juga terdapat fitur edit, dan hapus data tukang cukur, untuk dapat mengedit dan menghapus data.



Gambar 17. Tampilan Daftar KangCukur

Pada gambar 17 akan tampil data yang sudah diinputkan. Ketika tombol tambah diklik maka akan muncul tampilan untuk *input* data. Ketika akan menambahkan data, admin harus mengisi data foto, nama, umur, pengalaman kerja yang dituliskan dalam tahun, dan nomor *handphone*.

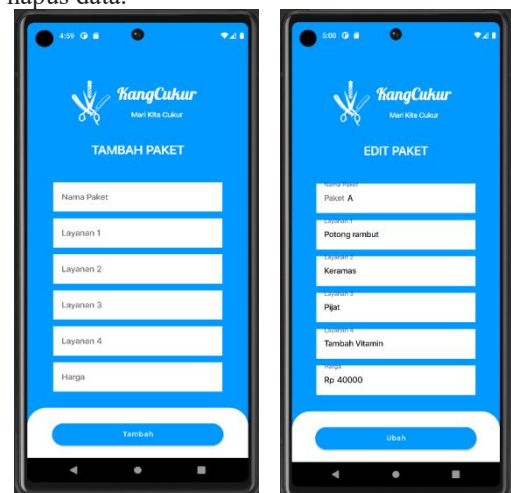


Gambar 18. Tampilan menu tambah KangCukur

Pada gambar 18 diatas adalah tampilan untuk mengubah dan mengedit data. Untuk dapat melakukan aksi ini, admin harus menekan beberapa detik pada menu dan akan muncul *pop up* edit dan hapus data. Ketika tombol ubah diklik maka admin harus menginputkan data yang akan diubah dan klik ubah, maka data akan berubah dan sistem akan menampilkan *alert* data berhasil diubah. Begitu pula pada fitur hapus data.

3.3.5 Menu Daftar Paket

Pada menu ini admin dapat menambahkan daftar paket yang terdiri dari nama paket, layanan yang tersedia, dan harga paket. Pada menu ini juga terdapat fitur edit dan juga hapus data. Untuk dapat mengedit dan menghapus data maka admin harus menekan beberapa detik pada menu daftar paket dan akan muncul tampilan untuk edit maupun hapus data.

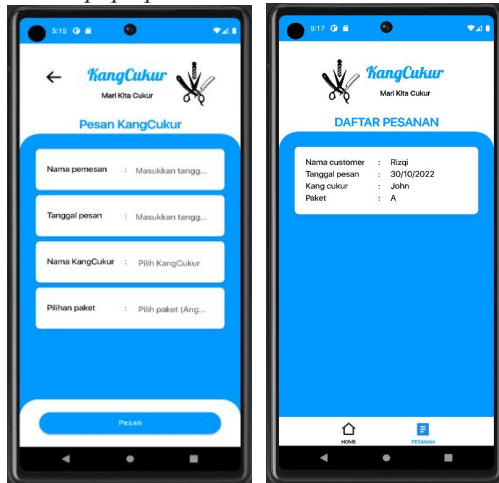


Gambar 19. Tampilan Menu Daftar Paket

3.3.6. Menu Tambah Pesanan

Pada menu ini admin dapat menambahkan pesanan. Akan muncul tampilan

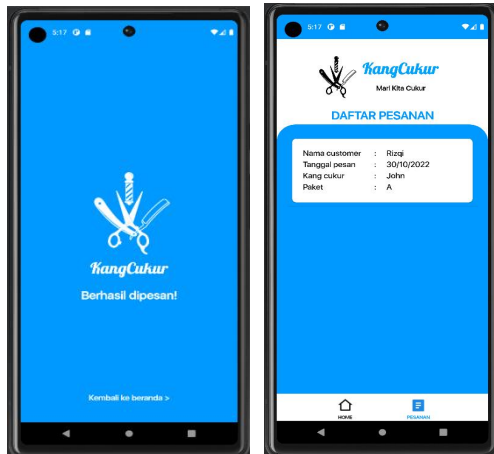
pesanan dan inputkan data berupa nama pemesan, tanggal pesan, nama tukang cukur, dan pilihan paket kemudian klik pesan untuk melanjutkan pesanan. Jika data tidak diisi akan muncul *pop up* data harus diisi.



Gambar 20. Tampilan Menu Tambah Pesanan

3.3.7. Menu Riwayat Pesanan

Setelah menambahkan pesanan akan muncul tampilan bahwa pesanan telah berhasil diinputkan dan admin dapat melihat pesanan yang sudah diinputkan pada menu riwayat pesanan.



Gambar 21. Tampilan Menu Riwayat Pesanan

3.4. Evaluasi Aplikasi

Evaluasi aplikasi merupakan tahap pengujian aplikasi oleh pengembang maupun *user*. Untuk evaluasi aplikasi KangCukur ini menggunakan *Blackbox Testing*.

3.4.1. Uji Blackbox

Pengujian *blackbox* bertujuan untuk mengetahui apakah aplikasi sudah berjalan sesuai dengan yang diharapkan, pengujian ini dilakukan dengan beberapa *test case* pada setiap inputan. Pengujian ini menguji kesalahan dalam aplikasi tanpa melihat *source code*, jika terdapat kesalahan pengembang segera

memperbaiki kesalahan tersebut. Hasil pengujian *blackbox* pada aplikasi Kangcukur ditunjukkan pada tabel 3 berikut.

Tabel 3. Hasil Uji *BlackBox*

Menu	Test case	Hasil yang diharapkan	Hasil Pengujian
Login	Mengkoson gkan salah satu atau kedua <i>form</i>	Gagal login	Valid
	Menginput kan email dan <i>password</i> salah	Gagal login	Valid
	Menginput kan email dan <i>password</i> benar	Berhasil login	Valid
Menu karyawan	Klik menu karyawan	Menampilkan menu karyawan	Valid
	Menginput kan foto, nama, umur pengalaman, dan nomor <i>handphone</i>	Berhasil menginputkan data	Valid
	Klik tambah dan coba untuk menginput kan data	Menampilkan form tambah data, dan Berhasil menambahkan data	Valid
	Klik lama pada halaman makan akan muncul pilihan edit dan hapus	Berhasil menampilkan edit dan hapus	Valid
	Klik edit, dan coba untuk mengganti	Data baru yang diinputkan berhasil	Valid

	inputan	diubah	
	Mencoba menghapus data karyawan yang telah diinputkan	Berhasil menghapus data karyawan yang sudah	Valid
Menu paket	Klik menu paket	Menampilkan halaman menu paket	Valid
	Klik tambah	Menampilkan form ubah data	Valid
	Mengisi form tambah menu paket	Berhasil menambahkan data yang telah diinputkan	Valid
	Mencoba mengubah data	Berhasil mengubah data	Valid
Menu pesanan	Mencoba menghapus data yang baru diinputkan	Berhasil menghapus data	Valid
	Klik menu pesanan	Menampilkan halaman menu pesanan	Valid
	Klik tambah	Berhasil menambahkan pesanal	Valid
	Klik menu konfirmasi pesanan	Berhasil mengonfirmasi pesanan	Valid
	Melihat riwayat pesanan	Berhasil menampilkan data riwayat pesanan yang sudah diinputkan	Valid
Logout	Klik logout	Berhasil logout dari aplikasi KangCukur	Valid

Pada tabel 3 menunjukan hasil uji *blackbox* pada semua *test case* valid. Berdasarkan tabel 3 diatas menunjukan bahwa aplikasi KangCukur ini sudah sesuai dengan yang diharapkan pengembang aplikasi

yang ditunjukan dengan hasil valid pada tabel pengujian menggunakan *blackbox*.

4. KESIMPULAN

Berdasarkan hasil laporan dapat disimpulkan bahwa:

1. Aplikasi KangCukur memiliki satu subsistem yaitu admin yang dapat mengakses semua fitur yang tersedia
2. Menu yang tersedia pada aplikasi kasir KangCukur terdiri dari menu daftar kancukur yang berisi data karyawan, menu paket yang berisi daftar paket yang tersedia untuk dipesan, dan menu pesanan yang dapat menambahkan pesanan dan data yang ditambahkan tersebut akan ditampilkan pada riwayat pesanan.
3. Hasil pengujian *blackbox* menunjukan hasil valid yang berarti bahwa semua fitur yang ada pada aplikasi KangCukur dapat berjalan sebagaimana yang diharapkan.

DAFTAR PUSTAKA

- [1] A. Herdiansah, D. Nurnaningsih, and H. Rusdianto, "Pemanfaatan Flutter Pada Pengembangan Aplikasi Mobile Ebisnis Penyediaan Bahan Baku Bisnis Katering," *J. TEKNOINFO*, vol. 16, no. 2, pp. 291–303, 2022, [Online]. Available: <https://ejurnal.teknokrat.ac.id/index.php/teknoinfo/index>
- [2] S. F. H. Susanto, Ardi, "Aplikasi Pencarian Jasa Pangkas Rambut Di Bandar Lampung Menggunakan Algoritma A-Star Berbasis Android," *J. Data Min. dan Sist. Inf.*, vol. 3, no. 1, pp. 39–49, 2022.
- [3] M. Fajar Irvansyah, Setiawansyah, "Aplikasi Pemesanan Jasa Cukur Rambut Berbasis Android," 2020.
- [4] A. S. Yoko Andramawan, Khairul Umami, "Rancang Bangun Aplikasi Pemesanan Jasa Perbaikan Komputer, Laptop, dan Smartphone Berbasis Android," *IT J.*, vol. 6, no. 1, 2018, doi: <http://dx.doi.org/10.22303/it.6.1.2018.25-35>.
- [5] D. M. Informatika, F. Teknik, U. N. Surabaya, T. Informatika, F. Teknik, and U. N. Surabaya, "Rancang Bangun Aplikasi Sistem Transaksi Laundry Berbasis Mobile Menggunakan Flutter Kharina Luthfi Asmunin".
- [6] R. I. T. A. H. Haedar, "Perancangan Sistem Informasi Booking Pangkas Rambut Berbasis Android Pada Portal Barbershop Kabupaten Bogor," vol. 3, no. 2, p. 6, 2021.
- [7] D. S. Purnia, A. Rifai, and S. Rahmatullah,

- “Penerapan Metode Waterfall dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android,” *Semin. Nas. Sains dan Teknol.* 2019, pp. 1–7, 2019.
- [8] Y. D. Wijaya and M. W. Astuti, “Sistem Informasi Penjualan Tiket Wisata Berbasis Web Menggunakan Metode Waterfall,” *Semin. Nas. Teknol. Inf. dan Komun.*, pp. 273–276, 2019.
- [9] F. Supandi, W. Desta P, Y. Ambar S, and M. Sudir, “Analisis Resiko Pada Pengembangan Perangkat Lunak Yang Menggunakan Metode Waterfall Dan Prototyping,” *Pros. Semin. Nas. Din. Inform. 2018 (SENADI 2018)*, vol. 2, no. 1, pp. 83–86, 2019.
- [10] P. Damayanti and S. Masripah, “Rancang Bangun Sistem Informasi Penyewaan menggunakan Metode Waterfall,” *J. Sist. Inf. Akunt.*, vol. 2, no. 1, pp. 30–41, 2021, doi: 10.31294/justian.v2i01.295.
- [11] N. R. Naufal and A. Eviyanti, “Perancangan Sistem Informasi Penyewaan Mobil Berbasis Web Menggunakan Framework Codeigniter,” *Zo. J. Sist. Inf.*, vol. 4, no. 1, pp. 31–41, 2022, doi: 10.31849/zn.v4i1.9547.
- [12] P. Pematangsiantar, S. T. Bangsa, and S. Utara, “Rancang bangun sistem aplikasi manajemen hotel berbasis web menggunakan framework codeigniter pada hotel parbina pematangsiantar 1,2,” vol. 1, no. 1, pp. 7–15, 2022.
- [13] A. Mariyus, N. Puwanti, and A. Aziz, “Aplikasi Pengolahan Data Puskesmas (Pusat Kesehatan Masyarakat) Desa Margodadi Kab. Tulang Bawang Barat,” *J. Simada*, vol. 02, no. 01, pp. 15–25, 2019.
- [14] A. M. K. Pandu Pratama, “Pengembangan Sistem Informasi Akademik Berbasis Mobile Menggunakan Flutter Di Universitas Narotama Surabaya Mobile-Based Academic Information System Development Using Flutter At Narotama University Surabaya,” 2021.
- [15] S. Tjandra and G. S. Chandra, “Pemanfaatan Flutter dan Electron Framework pada Aplikasi Inventori dan Pengaturan Pengiriman Barang,” *J. Inf. Syst. Hosp. Technol.*, vol. 2, no. 02, pp. 76–81, 2020, doi: 10.37823/insight.v2i02.109.
- [16] F. D. Astuti and Y. Hermawan, “Pemanfaatan Firebase Realtime Database Pada Aplikasi Pembelajaran Agama Islam Menggunakan Framework Flutter,” *J. Teknol. Inf. DAN ILMU Komput.*, vol. 18, no. 2, pp. 1–11, 2020.