

PERANCANGAN SISTEM INFORMASI PENGELOLAAN LAYANAN PESAN ANTAR PADA LAUNDRY BERBASIS MOBILE

Abdul Latief Aminullah ¹, Muhammad Ibrahim Al Farisi ², Anggoro Satrio Bhakti ³,
Zatin Niqotaini

¹²³⁴Sistem Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jakarta

Email: 2410512024@mahasiswa.upnvj.ac.id ¹, 2410512026@mahasiswa.upnvj.ac.id ²,
241051203x@mahasiswa.upnvj.ac.id ³, zatinniqotaini@gmail.com ⁴

ABSTRAK

Perkembangan teknologi sangat berpengaruh terhadap efisiensi operasional sebuah bisnis, termasuk dalam bidang jasa laundry. Laundry masih menggunakan proses pengelolaan secara manual, yang menyebabkan kurangnya efektivitas dalam pencatatan transaksi, pemantauan proses laundry, serta pelayanan terhadap pelanggan. Oleh karena itu, diperlukan suatu perancangan sistem informasi berbasis aplikasi mobile yang dapat mendukung proses digitalisasi layanan. Penelitian ini menghasilkan rancangan sistem pengelolaan laundry menggunakan pemodelan Unified Modeling Language (UML) yang dituangkan dalam bentuk use case diagram, activity diagram, sequence diagram, dan class diagram. Sistem dirancang dalam bentuk mockup aplikasi mobile dengan dua jenis pengguna utama, yaitu pelanggan dan petugas laundry. Pelanggan dapat melakukan pemesanan, memantau status cucian, serta melihat riwayat transaksi melalui aplikasi. Sementara itu, petugas dapat mengelola data transaksi, status layanan, dan informasi pelanggan. Diharapkan rancangan sistem ini dapat mempermudah proses operasional laundry dan meningkatkan kualitas pelayanan kepada pelanggan.

Kata Kunci: Sistem Informasi, Laundry, Pesan Antar, UML.

ABSTRACT

The development of technology has a significant impact on the operational efficiency of a business, particularly in the laundry services sector. Laundry still employs a manual management process, which results in a lack of effectiveness in recording transactions, monitoring the laundry process, and serving customers. Therefore, it is necessary to design a mobile application-based information system that can support the service digitization process. This research resulted in the design of a laundry management system using the Unified Modeling Language (UML), as outlined in the form of Use Case Diagrams, Activity Diagrams, Sequence Diagrams, and Class Diagrams. The system is designed in the form of a mobile application mock-up with two main types of users, namely customers and laundry officers. Customers can place orders, monitor laundry status, and view transaction history through the application. Meanwhile, officers can manage transaction data, service status, and customer information. It is hoped that this system design can simplify the laundry operational process and improve the quality of service to customers.

Keywords: Information System, Laundry, Delivery, UML.

1 PENDAHULUAN

Memasuki era globalisasi, penggunaan Teknologi Informasi (TI) meningkat secara signifikan, seiring dengan peningkatan akses dan pemakaian internet. Internet telah membawa perubahan drastis dalam kehidupan masyarakat Indonesia serta mentransformasi aktivitas bisnis. Teknologi diyakini sebagai kunci peningkatan efisiensi dan efektivitas operasional perusahaan dalam mencapai target dan tujuan [14]. Oleh karena itu, banyak perusahaan dan instansi mulai mengintegrasikan proses bisnis dengan TI untuk mempermudah pencapaian target bisnis [6]. Hal ini menuntut perusahaan untuk menyesuaikan cara kerja mereka agar tidak tertinggal dari kompetitor [8].

Salah satu sektor bisnis yang terdampak adalah bisnis laundry, yakni jasa pencucian pakaian menggunakan mesin cuci, mesin pengering, serta bahan pembersih dan pewangi khusus [13]. Secara umum, pelanggan harus mengantarkan sendiri pakaian kotor ke tempat laundry. Namun, cara ini kurang praktis bagi pelanggan yang sibuk dan tidak memiliki banyak waktu untuk mengantar, mengambil, dan membayar cucian mereka [12].

Untuk menjawab permasalahan ini, beberapa laundry mulai menawarkan layanan pesan antar melalui *WhatsApp*, dimana pelanggan dapat menghubungi pihak laundry dan kurir akan menjemput cucian. Meski cukup membantu, layanan ini masih memiliki kekurangan, yaitu tidak tersedianya pemantauan status cucian secara real-time. Pelanggan harus menghubungi admin secara berulang melalui *WhatsApp* untuk mengetahui status cucian, yang dianggap tidak praktis.

Berdasarkan permasalahan tersebut, dirancang solusi berupa sistem berbasis mobile yang mendukung layanan laundry pesan antar. Sistem ini memfasilitasi pemesanan online, pembayaran online, dan pemantauan status cucian secara real-time tanpa perlu interaksi langsung antara pelanggan dan pihak laundry. Dengan sistem ini, pelanggan tidak perlu mengantar cucian ke tempat laundry dan bisa memantau perkembangan status cucian mereka hanya melalui ponsel.

Penelitian ini bertujuan menghasilkan rancangan sistem informasi dan rancangan UI Aplikasi untuk layanan pesan antar laundry yang lebih efektif dan efisien. Sistem ini juga diharapkan meningkatkan daya tarik layanan laundry melalui kemudahan yang ditawarkan kepada pelanggan. Fitur utama dalam sistem ini mencakup login, pemesanan, status laundry, pembayaran, serta pengelolaan laundry, penjemputan, dan pengiriman cucian. Jika sistem ini berhasil diimplementasikan, diharapkan dapat membantu bisnis laundry dalam mencapai bahkan melampaui target bisnisnya.

Efektivitas sistem informasi telah dibuktikan dalam berbagai bidang. Dalam pendidikan, sistem informasi pada proses Penerimaan Peserta Didik Baru (PPDB) terbukti memudahkan calon siswa dalam mengunggah dokumen serta membantu panitia dan admin dalam mengelola data pendaftaran [5]. Studi lain menunjukkan sistem informasi perpustakaan di MAN 22 Jakarta mempermudah penyimpanan dan pengelolaan e-book serta proses peminjaman dan pengembalian buku [7]. Meski berada di bidang berbeda, penelitian-penelitian ini menguatkan relevansi penelitian terkait sistem informasi laundry.

Perancangan sistem dilakukan menggunakan metode visualisasi pemodelan *Unified Modeling Language (UML)*, yang menggambarkan interaksi dan relasi antara aktor dan fungsi dalam sistem. Diagram UML yang digunakan meliputi *Use Case*, *Activity*, *Sequence*, dan *Class Diagram*. Implementasi UML ini diharapkan menghasilkan rancangan sistem informasi laundry pesan antar yang efektif dalam mengatasi kendala pada layanan laundry konvensional.

2 METODE PENELITIAN

2.1. Metode Penelitian

Penelitian ini dilakukan dengan mengadopsi Metode R&D atau dikenal sebagai *Research and Development*. R&D adalah metode yang dipakai untuk menghasilkan sebuah luaran berupa Produk baru, sekaligus menguji Efektivitas dari Produk tersebut. Untuk menghasilkan sebuah Produk Baru, maka diperlukan sebuah Metodologi Penelitian yang bersifat analitis terhadap kebutuhan User dan juga dapat menguji tingkat Efektivitas dari Produk tersebut [11]. Pada umumnya, berikut adalah tahapan-tahapan yang harus ditempuh dalam mengaplikasikan Metodologi R&D.

2.1.1. Analisis Permasalahan dan Pengumpulan Informasi

Pada tahap ini, dilakukan Pengumpulan Pengetahuan mengenai Proses Bisnis Pesan Antar Laundry yang konvensional. Pengetahuan dapat diraih dengan melakukan pengamatan secara tidak langsung terhadap Proses Pelayanan Laundry dan juga dengan melakukan sesi diskusi dengan Customer. Setelah mendapatkan pengetahuan yang cukup terhadap Proses Bisnis yang berjalan, maka dilakukan proses Identifikasi dari permasalahan dan kekurangan yang tersimpan dalam Proses Bisnis tersebut.

2.1.2. Perencanaan

Pada fase ini, perlu dilakukan penyusunan rencana terhadap apa saja yang perlu dilakukan pada tahap-tahap selanjutnya. Perencanaan bisa dilakukan dengan berdiskusi dan mencatat apa saja yang harus dibuat, atau dengan menyusun sebuah *Flowchart* untuk memvisualisasikan tahapan apa saja yang harus dilalui beserta urutannya secara kronologis.

2.1.3. Perancangan/Pengembangan

Tahap Perancangan dilakukan berdasarkan hasil Analisis Masalah yang telah dilakukan yang berbasis Pengetahuan tentang Permasalahan Proses Bisnis yang telah didapat dan diketahui. Pada fase ini, disusun Diagram-Diagram yang diperlukan yaitu *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan terakhir *Class Diagram*.

2.1.4. Revisi

Pada tahap ini dilakukan kembali Analisis yang detail terhadap *Output* dari tahap Perancangan untuk menemukan kesalahan dan kekurangan pada Diagram-Diagram yang telah dibuat. Jika ditemukan, maka akan dilakukan Penyesuaian dan Revisi terhadap kesalahan dan kekurangan tersebut. Ini dilakukan demi menjaga kualitas dari *Output* yang dihasilkan dari Penelitian ini.

2.1.5. Publikasi

Terakhir, dilakukan pelampiran Rancangan Sistem Informasi Layanan Pesan Antar Laundry di dalam Jurnal untuk dipublikasikan yang nantinya dapat diakses oleh semua pihak dan dapat dijadikan acuan untuk dilakukan *Development* lebih lanjut ataupun direalisasikan menjadi sebuah Sistem nyata yang siap pakai, mengingat ruang lingkup Penelitian ini hanya mencapai sebatas rancangan saja.

2.2. Kajian Teori

2.2.1. Sistem Informasi

Sistem Informasi merupakan gabungan dari Kata “Sistem” dan Kata “Informasi”. Sistem adalah kumpulan dari Elemen-Elemen fungsional dengan Tugas dan Fungsinya masing masing yang saling terkait kemudian secara bersama bekerja untuk memenuhi suatu proses dalam rangka mencapai sebuah tujuan yang sama [1].

Sedangkan Informasi memiliki definisi yaitu hasil dari proses olahan suatu Data yang dianggap sebagai suatu Objek penting bagi para Penerimaanya [3]. Dari dua Definisi di ini, dapat disimpulkan bahwa Sistem Informasi adalah kumpulan dari *Hardware*, *Software*, *Brainware* yang saling bekerja sama untuk mengolah sebuah Data menjadi Informasi yang merupakan Objek berharga bagi para penerimaanya [10].

2.2.2 UML

Unified Modeling Language (disingkat UML) adalah sebuah Bahasa Universal dan metode pengembangan Sistem *Object Oriented* yang digunakan oleh para *System Analyst* dan *System Architect* dalam mendefinisikan, memvisualisasikan, dan membangun Rancangan *Software* [2]. Dalam Penelitian ini, terdapat 3 Jenis Diagram UML yang digunakan yaitu *Use Case Diagram*, yang kemudian dikembangkan menjadi *Activity Diagram*, *Sequence Diagram*, dan disusun menjadi *Class Diagram*.

Use Case Diagram adalah Diagram yang menggambarkan fungsi-fungsi dalam Sistem maupun Kelas, menjelaskan bagaimana Sistem berinteraksi dengan Lingkungan luar Sistem serta menggambarkan Fungsionalitas *Software* atau Sistem yang dapat diakses oleh pengguna/*User* [9]. *Use Case* juga menggambarkan relasi interaksi yang terbentuk antara Pengguna/*User* dengan Sistem [9].

Use Case Diagram tadi dapat dikembangkan ke tahap selanjutnya yaitu menjadi *Activity Diagram*. Ini adalah sebuah Diagram yang memvisualisasikan alur aktivitas antar entitas/aktor dalam sebuah Sistem. *Activity Diagram* berfungsi untuk menunjukkan urutan proses yang terjadi dalam sebuah Sistem [9].

Selanjutnya, dibuat *Sequence Diagram* yang menggambarkan secara lebih detail lagi urutan pertukaran Informasi antar objek di dalam Sistem, termasuk operasi dan fungsi apa yang bekerja, dan informasi apa saja yang diminta oleh sebuah Fungsi atau Objek untuk bekerja. Diagram ini disusun dalam urutan Kronologis.

Terakhir, dari Diagram-Diagram sebelumnya maka dapat dibuat sebuah *Class Diagram*. *Class Diagram* mengilustrasikan struktur statis dari sebuah *Software* atau Sistem. *Class Diagram* terdiri dari beberapa *Class* yang berisi Atribut dan Operasi/*Method*, Nama *Class* itu sendiri, serta relasi antar *Class* [4], [9].

2.2.3. Layanan Pesan Antar (Delivery Service)

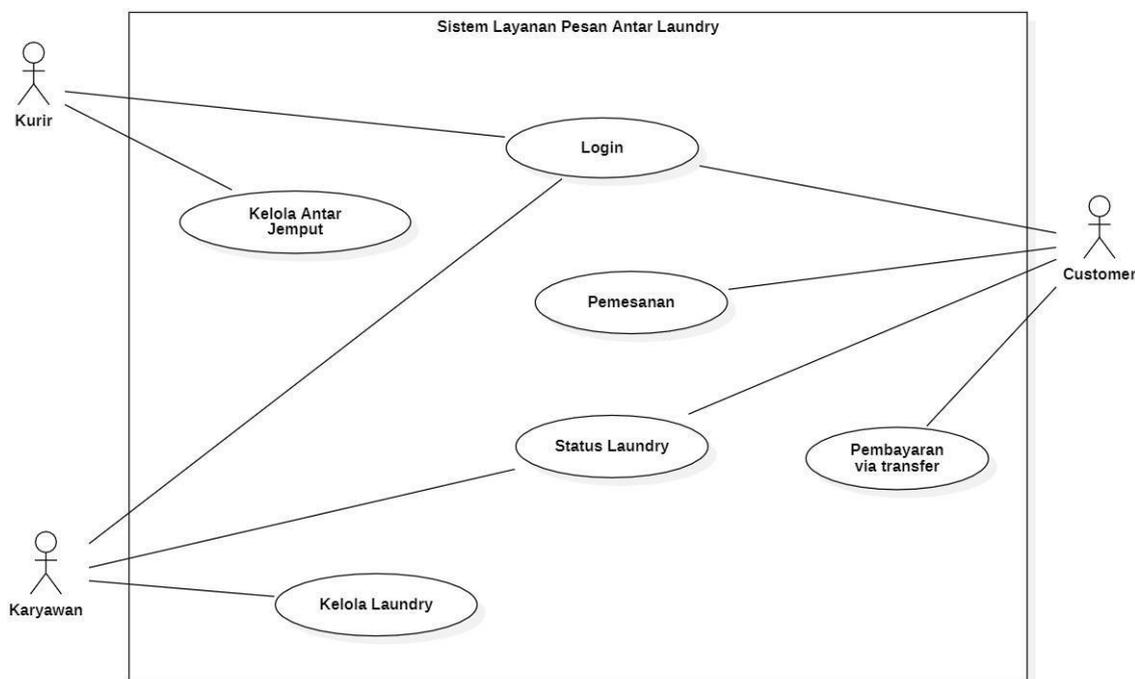
Layanan pesan antar merupakan evolusi dari model bisnis yang berfokus pada kenyamanan pelanggan. Model bisnis ini, yang sering dikategorikan sebagai *on-demand service*, memungkinkan pelanggan untuk memesan produk atau jasa dari lokasi mereka dan menerimanya tanpa harus mengunjungi lokasi fisik penyedia jasa [15]. Proses bisnis intinya melibatkan serangkaian aktivitas mulai dari pemesanan, konfirmasi, penjemputan (*pickup*), pengantaran (*delivery*), hingga penyelesaian transaksi.

3 HASIL DAN PEMBAHASAN

Berikut adalah Hasil dari Analisis yang kami lakukan dan kami tuangkan dalam bentuk Rancangan Sistem berbasis *Unified Modeling Language* (UML). Diagram-Diagram yang dibuat adalah *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*. Di bawah ini adalah Diagram-Diagram tersebut:

3.1. Use Case Diagram

Output pertama yang disusun menggunakan aplikasi StarUML adalah *Use Case Diagram*. Diagram ini adalah Induk dari semua Diagram-Diagram yang dibuat seterusnya karena *Use Case Diagram* merupakan *Blueprint* yang menjelaskan bagaimana Sistem berinteraksi dengan Lingkungan luar Sistem serta menggambarkan Fungsi atau Sistem yang dapat diakses oleh pengguna/User. Diagram ini juga menggambarkan relasi interaksi yang terbentuk antara Pengguna/User dengan Sistem. Berikut adalah *Use Case Diagram* yang dimaksud:

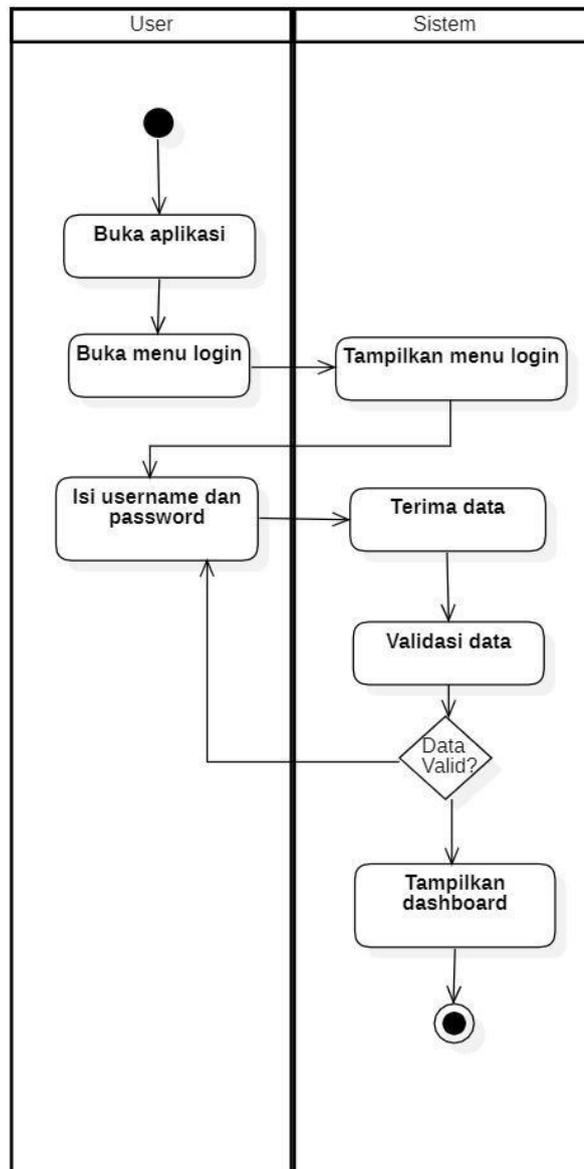


Gambar 1: Use Case Diagram

Use Case Diagram ini mendeskripsikan sistem laundry berbasis *mobile* yang melibatkan tiga peran penting (*Customer*, *Kurir*, dan *Karyawan*) dengan alur utama mulai dari pemesanan, penjemputan, pengolahan cucian, hingga pengiriman kembali ke pelanggan. Sistem ini mempermudah proses operasional laundry secara digital dan efisien.

3.2. Activity Diagram

3.2.1. Activity Diagram Login



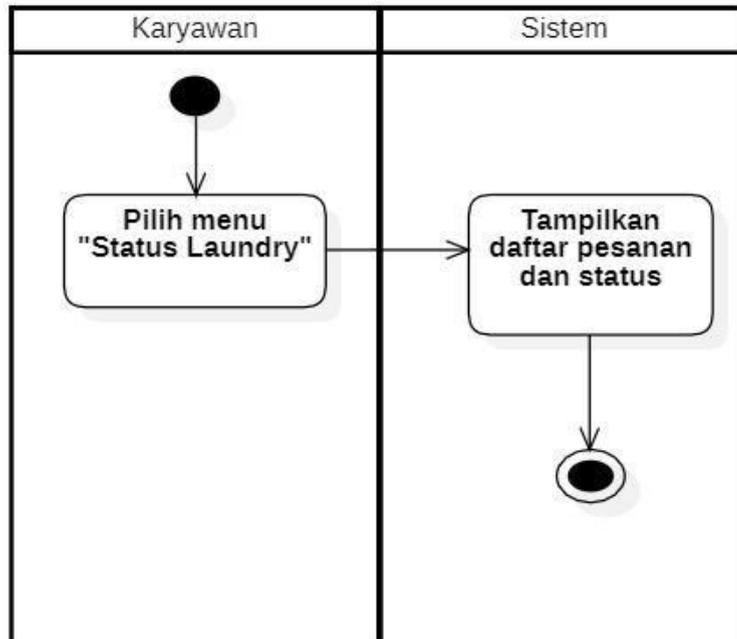
Gambar 2: Activity Diagram (Login)

Aktivitas *Login* melibatkan 2 Aktor yaitu *User* (dalam konteks ini bisa *Customer*, *Karyawan*, maupun *Kurir*) dan *Sistem*. Aktivitas dimulai dengan *User* membuka Aplikasi, kemudian *User* akan membuka atau memilih Menu *Login* yang tersedia di Aplikasi. Setelah itu, maka *Sistem* akan menampilkan Halaman *Login*. Di dalamnya, terdapat beberapa kolom yang harus diisi oleh *User* yaitu *Username* dan juga *Password* dari akun mereka.

Jika Data-Data yang diperlukan sudah diisi, maka *Sistem* akan menerima *Dataset* tersebut. Dengan ini, maka Proses *Validasi Data* dimulai. Dalam proses ini, muncul sebuah *Decision* atau *if-else condition*. Jika Data yang dimasukkan *valid*, maka *Sistem* akan menampilkan Halaman Utama atau *Dashboard*. Sedangkan, jika Data yang diterima oleh *Sistem* tidak *Valid*, maka *Sistem* tidak akan berlanjut ke Halaman Utama tetapi akan kembali ke Halaman *Login* dan meminta *User* untuk

melakukan *input* ulang Data.

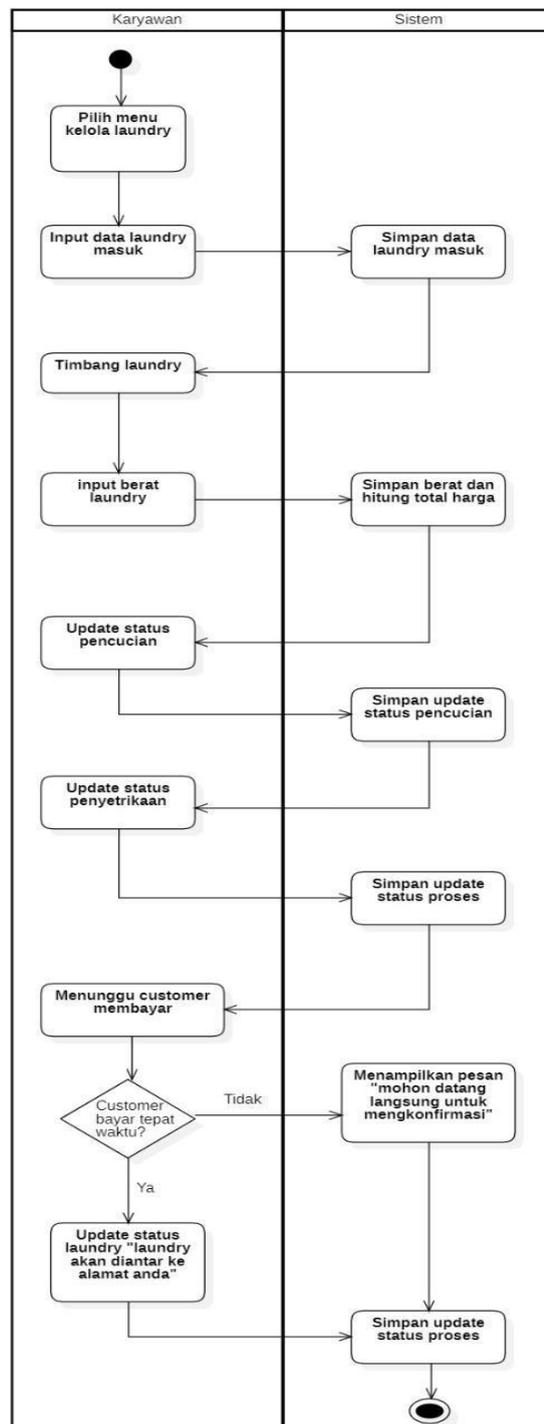
3.2.2. Activity Diagram Status Laundry (Karyawan)



Gambar 3: Activity Diagram (Status Laundry Karyawan)

Diagram yang terlihat pada gambar 3 ini, menggambarkan alur aktivitas yang terjadi ketika seorang karyawan ingin melihat daftar pesanan dan status laundry melalui sistem. Proses dimulai saat karyawan masuk ke sistem dan memilih menu "Status Laundry" yang tersedia. Setelah menu tersebut dipilih, sistem akan secara otomatis menampilkan seluruh daftar pesanan laundry pelanggan lengkap dengan status terkini, seperti apakah laundry sedang dijemput, dalam proses pencucian, telah selesai, atau sedang dalam pengantaran. Aktivitas ini berakhir ketika sistem berhasil menampilkan informasi tersebut kepada karyawan.

3.2.3. Activity Diagram Kelola Laundry



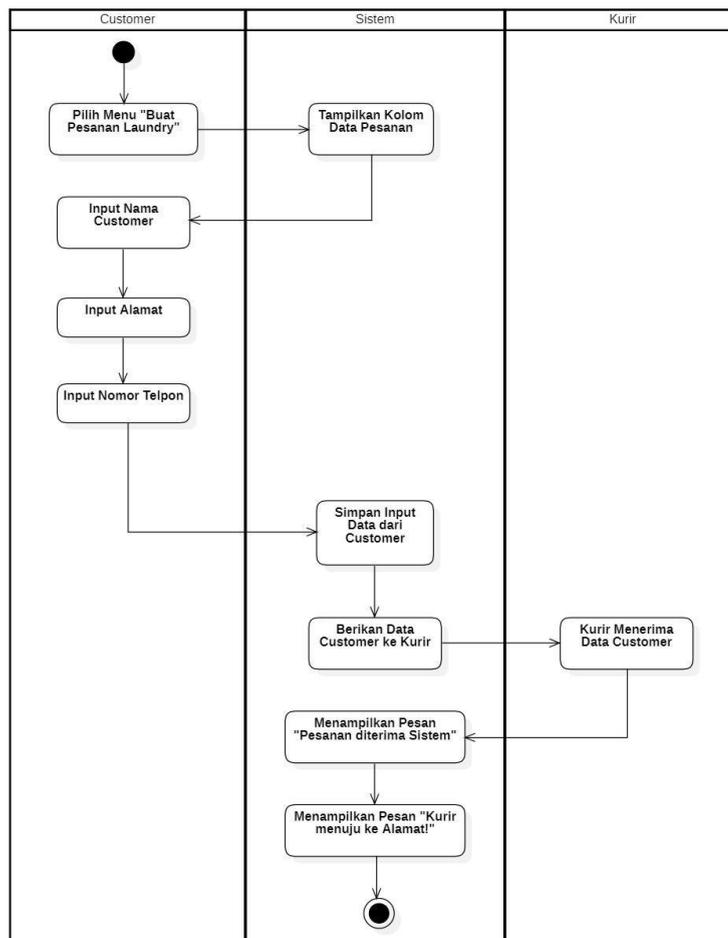
Gambar 4: Activity Diagram (Kelola Laundry Karyawan)

Diagram diatas menggambarkan alur aktivitas seorang karyawan dalam mengelola proses laundry pelanggan, mulai dari pencatatan hingga proses pengantaran. Aktivitas diawali dengan karyawan memilih menu "Kelola Laundry" pada sistem untuk memulai pengelolaan. Setelah itu, karyawan memasukkan data cucian yang baru diterima dari pelanggan, kemudian melakukan

penimbangan laundry. Berat laundry yang telah ditimbang dimasukkan ke dalam sistem untuk menghitung total harga layanan.

Selanjutnya, karyawan memperbarui status proses laundry, mulai dari tahap pencucian hingga penyetrikaan, sesuai perkembangan yang terjadi. Setelah semua proses selesai, sistem akan menunggu konfirmasi pembayaran dari pelanggan. Jika pelanggan belum melakukan pembayaran tepat waktu, sistem akan menampilkan pesan yang meminta pelanggan untuk datang langsung guna melakukan konfirmasi. Namun jika pembayaran sudah dilakukan, maka status laundry akan diperbarui menjadi "laundry akan diantar ke alamat anda". Sepanjang proses ini, sistem secara otomatis menyimpan setiap input dan pembaruan status yang dilakukan oleh karyawan.

3.2.4. Activity Diagram Pemesanan

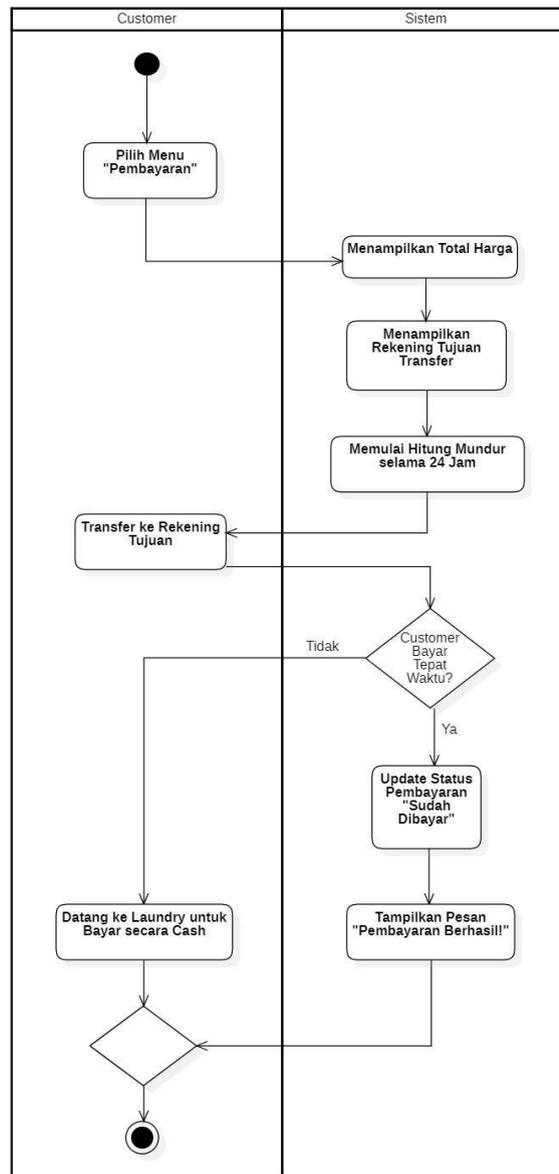


Gambar 5: Activity Diagram (Pemesanan)

Seluruh proses yang terjadi pada Sistem dan yang perlu dilakukan oleh Customer telah tervisualisasikan pada Activity Diagram di atas. Proses Pemesanan dimulai oleh Customer dengan memilih menu “Buat Pesanan Laundry”. Dengan memilih menu ini, Proses selanjutnya adalah Sistem akan menampilkan Halaman untuk membuat Pesanan Laundry. Pada halaman ini, akan ditampilkan kolom Nama, Alamat, dan Nomor Telepon yang harus diisi oleh Customer sebagai Data Diri Customer yang setelahnya akan disimpan di Database.

Setelah Data Customer tersimpan di Database, maka Data tersebut akan diteruskan kepada Kurir Laundry supaya Kurir bisa segera menuju ke Alamat Customer. Saat Sistem mengirimkan data ke Kurir, maka Notifikasi akan tampil pada Aplikasi Kurir. Kemudian, Sistem akan menampilkan pesan konfirmasi bahwa Pesanan telah diterima sistem dan Kurir sedang menuju ke Alamat yang Customer masukkan ke dalam Sistem. Jika semua proses sudah terlaksana dengan Sukses, maka Sistem akan mengakhiri proses Activity Pemesanan ini.

3.2.5. Activity Diagram Pembayaran via Transfer



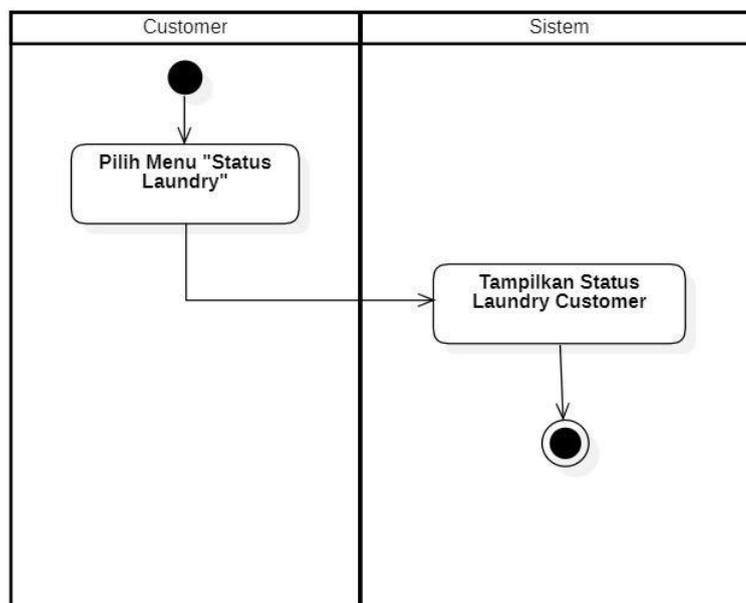
Gambar 6: Activity Diagram (Pembayaran via Transfer)

Setelah melakukan Pemesanan, maka Customer harus melalui Proses Pembayaran via Transfer. Demi meningkatkan kepraktisan dalam Sistem ini, Pembayaran hanya bisa dilakukan melalui Transfer *Mobile Banking* sehingga mengeliminir kebutuhan bagi Customer untuk menyediakan uang pas dan Kurir yang harus membawa uang kembali. Proses dimulai oleh Customer dengan memilih menu “Pembayaran”.

Ketika *Customer* membuka menu “Pembayaran”, Sistem akan melakukan 3 *Activity* secara berurutan yaitu menampilkan Total Harga Cucian sang *Customer*, menampilkan Rekening tujuan Transfer, dan memulai *countdown* selama 24 Jam. Ini dilakukan supaya *Customer* memiliki tenggat waktu untuk menyelesaikan proses Pembayaran ini dan mencegah kemungkinan adanya *Customer* yang terlambat untuk membayar.

Karena jika dalam 24 Jam Sistem tidak mendeteksi adanya Transfer masuk dari *Customer*, maka Cucian tidak akan dikirimkan ke Alamat *Customer*. Untuk mendapatkan kembali Cuciannya, *Customer* harus datang secara langsung ke tempat Laundry untuk menyelesaikan Pembayaran secara Offline. Sedangkan, jika Sistem mendeteksi *Customer* telah melakukan Transfer kepada Laundry dalam kurun waktu 24 jam, maka status pembayaran akan di-update menjadi “Sudah Dibayar” dan Sistem akan menampilkan Pesan “Pembayaran Berhasil!” kepada *Customer*. Setelah itu, maka Kurir akan menuju ke Alamat *Customer* untuk mengantarkan kembali Cucian milik *Customer*.

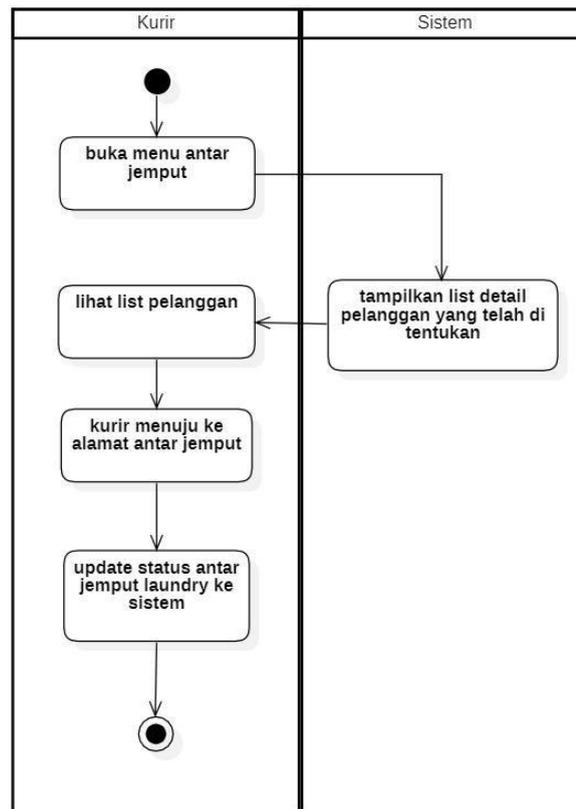
3.2.6. *Activity Diagram* Status Laundry (*Customer*)



Gambar 7: *Activity Diagram* (Status Laundry *Customer*)

Gambar di atas merupakan *Activity Diagram* yang memvisualisasikan alur aktivitas yang terjadi ketika seorang *Customer* ingin melihat daftar Status Laundry mereka melalui Sistem. Proses dimulai saat *Customer* memilih menu "Status Laundry". Setelah menu tersebut dipilih, sistem akan secara otomatis menampilkan Status dari Laundry atau Cucian mereka, yang mana Status yang ditampilkan bersifat *real-time*. Status yang ditampilkan bisa berupa Kurir sedang menuju ke Alamat, Cucian dalam proses pencucian, telah selesai, atau sedang dalam pengantaran kembali ke Alamat *Customer*. Aktivitas ini berakhir ketika sistem berhasil menampilkan informasi tersebut kepada karyawan.

3.2.7. Activity Diagram Kelola Antar Jemput



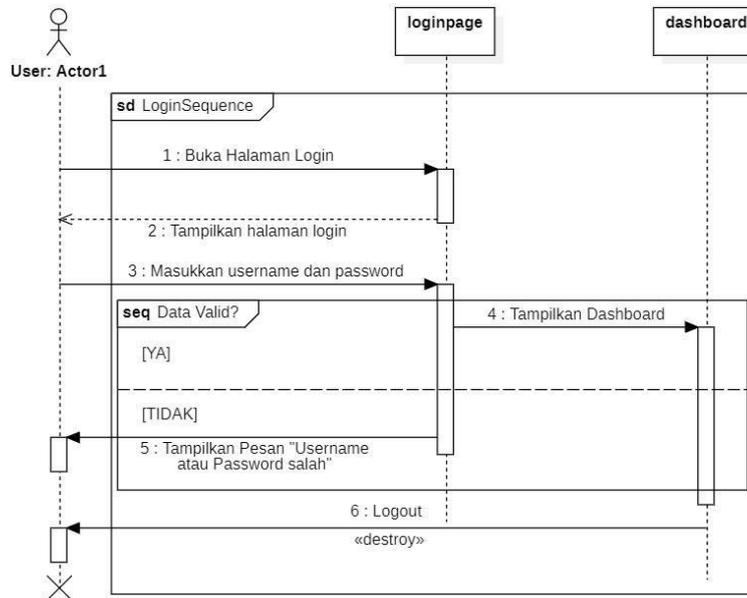
Gambar 8: Activity Diagram (Kelola Antar Jemput)

Activity Diagram di atas merupakan gambaran dari Activity pengelolaan Antar Jemput Cuci Customer yang dilakukan oleh Kurir. Aktivitas diawali dengan Kurir membuka Menu Antar Jemput yang tersedia dalam Aplikasi/Sistem. Sebagai catatan, diasumsikan bahwa Kurir yang dimiliki Laundry berjumlah lebih dari 1 dan Sistem yang membagi pengantaran kepada para Kurir. Berdasarkan penjelasan tersebut, Sistem akan menampilkan Daftar Customer yang perlu ditangani oleh tiap Kurir setelah Menu tersebut dibuka.

Setelah menerima Data Alamat Customer, maka Kurir akan menuju ke Alamat untuk menjemput Cuci atau mengantar kembali Cuci milik Customer. Selain itu, Kurir juga harus update Status pengantaran mereka ketika sedang menjemput atau mengantar Cuci. Ini dilakukan supaya Sistem bisa mengetahui Status dari Kurir dan bisa menampilkan Informasi tersebut kepada Customer.

3.3 Sequence Diagram

3.3.1. Sequence Diagram Login

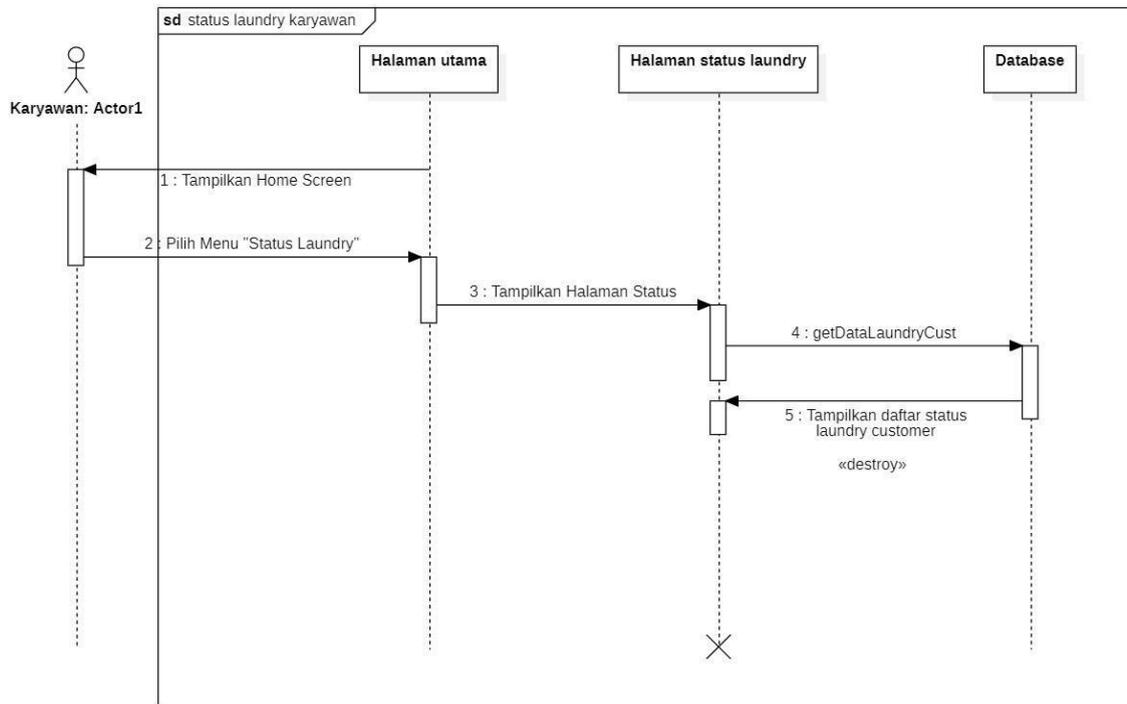


Gambar 9: Sequence Diagram (Login)

Diagram di atas menjelaskan tentang alur Interaksi antara Aktor yaitu User (yang bisa berupa Customer, Karyawan, maupun Kurir) dengan beberapa Objek dalam Sistem untuk menyelesaikan sebuah Aktivitas yaitu Login. Proses Login dimulai dengan User memilih menu Login. Fungsi loginPage akan bekerja dan menampilkan Halaman Login. Selanjutnya, User akan memasukkan Data-Data akun mereka pada Kolom yang tersedia.

Setelahnya, akan dilakukan proses Validasi Data. Jika Data yang dimasukkan valid, maka Sistem akan menampilkan Halaman Utama atau Dashboard. Sedangkan, jika Data yang diterima oleh Sistem tidak Valid, maka Sistem tidak akan berlanjut ke Halaman Utama tetapi akan kembali ke Halaman Login dan meminta User untuk melakukan input ulang Data.

3.3.2. Sequence Diagram Status Laundry Karyawan

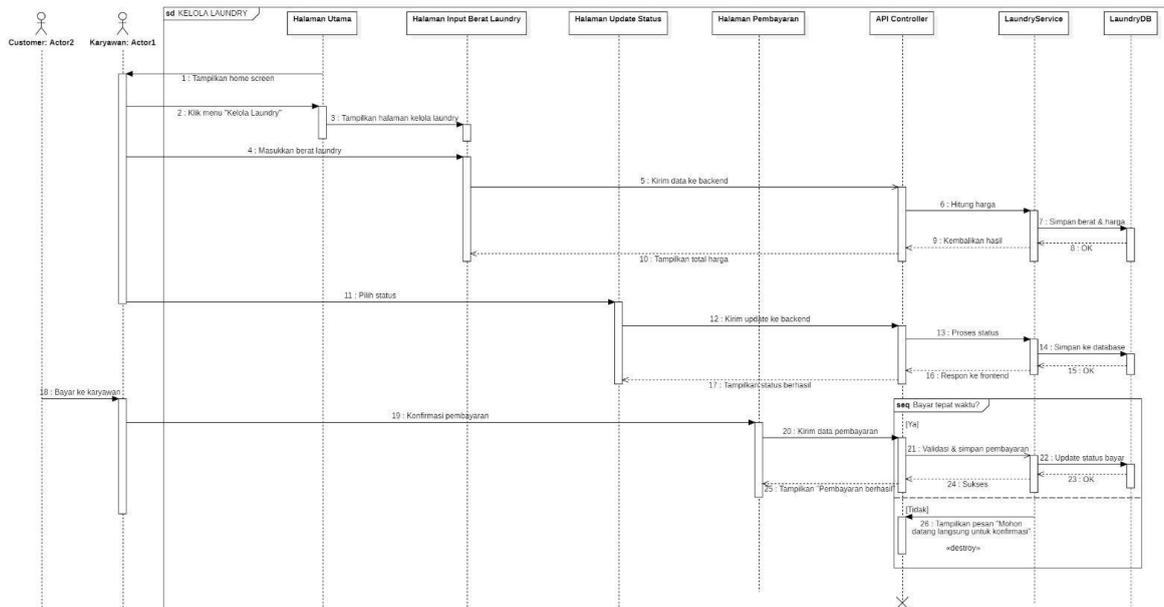


Gambar 10: Sequence Diagram (Status Laundry Karyawan)

Diagram ini menjelaskan alur komunikasi antar objek dalam sistem ketika seorang karyawan ingin melihat status laundry pelanggan. Proses dimulai saat karyawan membuka aplikasi, kemudian sistem secara otomatis menampilkan halaman utama (*Home Screen*). Dari halaman ini, karyawan memilih menu "Status Laundry", yang memicu sistem untuk menavigasi ke halaman khusus yang menampilkan informasi status laundry.

Selanjutnya, halaman status laundry melakukan permintaan data ke database melalui perintah *getDataLaundryCust* untuk mengambil informasi terkait status cucian pelanggan. Setelah data diterima dari database, sistem menampilkan daftar status laundry secara lengkap kepada karyawan. Setelah seluruh informasi berhasil ditampilkan, sesi interaksi pada halaman status laundry dianggap selesai, dan objek halaman tersebut dihancurkan sebagai tanda akhir proses.

3.3.3. Sequence Diagram Kelola Laundry



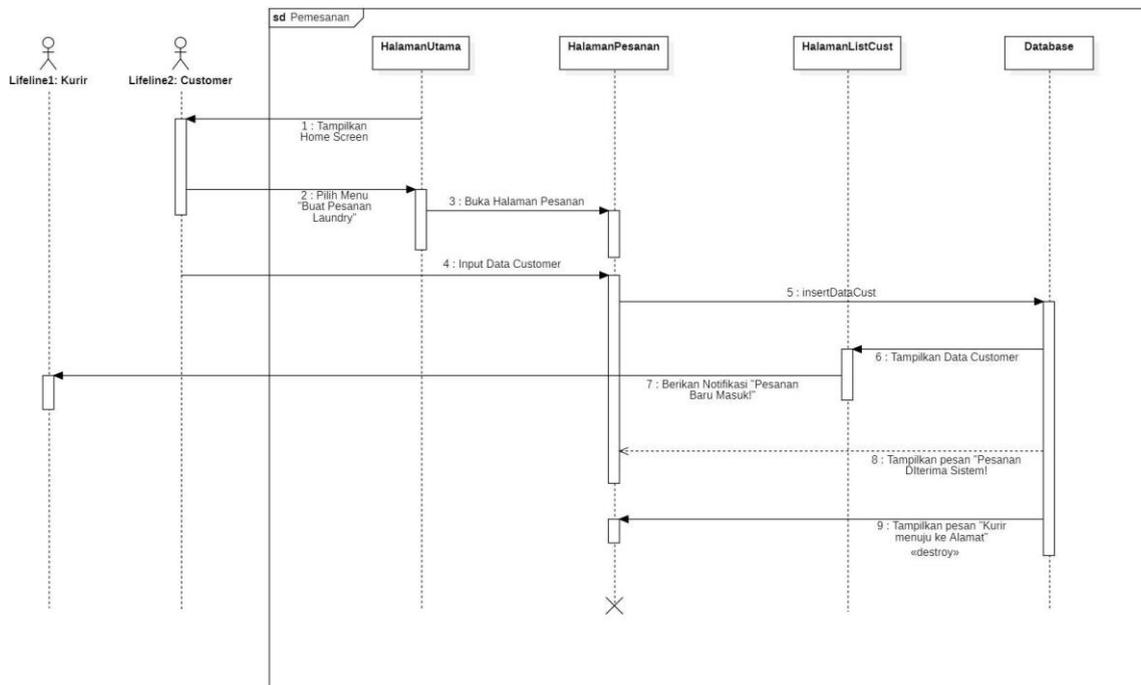
Gambar 11: Sequence Diagram (Kelola Laundry Karyawan)

Diagram ini menggambarkan alur komunikasi antara aktor, yaitu *Karyawan* dan *Customer*, dengan berbagai komponen dalam sistem saat proses pengelolaan laundry berlangsung, dimulai dari input data cucian hingga tahap pembayaran. Proses dimulai ketika karyawan membuka aplikasi dan memilih menu "Kelola Laundry", yang kemudian menampilkan halaman input berat laundry. Setelah itu, karyawan memasukkan berat cucian pelanggan, dan data tersebut dikirim ke backend.

Backend kemudian memproses perhitungan total harga dengan menggunakan layanan *LaundryService*. Hasil perhitungan disimpan ke dalam database (*LaundryDB*) dan dikembalikan ke antarmuka pengguna (frontend) untuk ditampilkan kepada karyawan. Setelah total harga ditampilkan, karyawan memilih status proses laundry, seperti "dicuci" atau "disetrika". Informasi status ini dikirim ke backend dan disimpan dalam database. Sistem lalu menampilkan pesan bahwa status laundry berhasil diperbarui.

Tahapan selanjutnya melibatkan pelanggan yang melakukan pembayaran kepada karyawan. Karyawan kemudian melakukan konfirmasi pembayaran, dan sistem akan memvalidasi apakah pembayaran dilakukan tepat waktu. Jika pembayaran dilakukan sesuai waktu yang ditentukan, maka sistem akan memperbarui status laundry menjadi "akan dikirim" dan menampilkan pesan bahwa pembayaran berhasil. Namun, jika pembayaran belum dilakukan tepat waktu, sistem akan menampilkan pesan peringatan yang berbunyi "Mohon datang langsung untuk konfirmasi."

3.3.4. Sequence Diagram Pemesanan

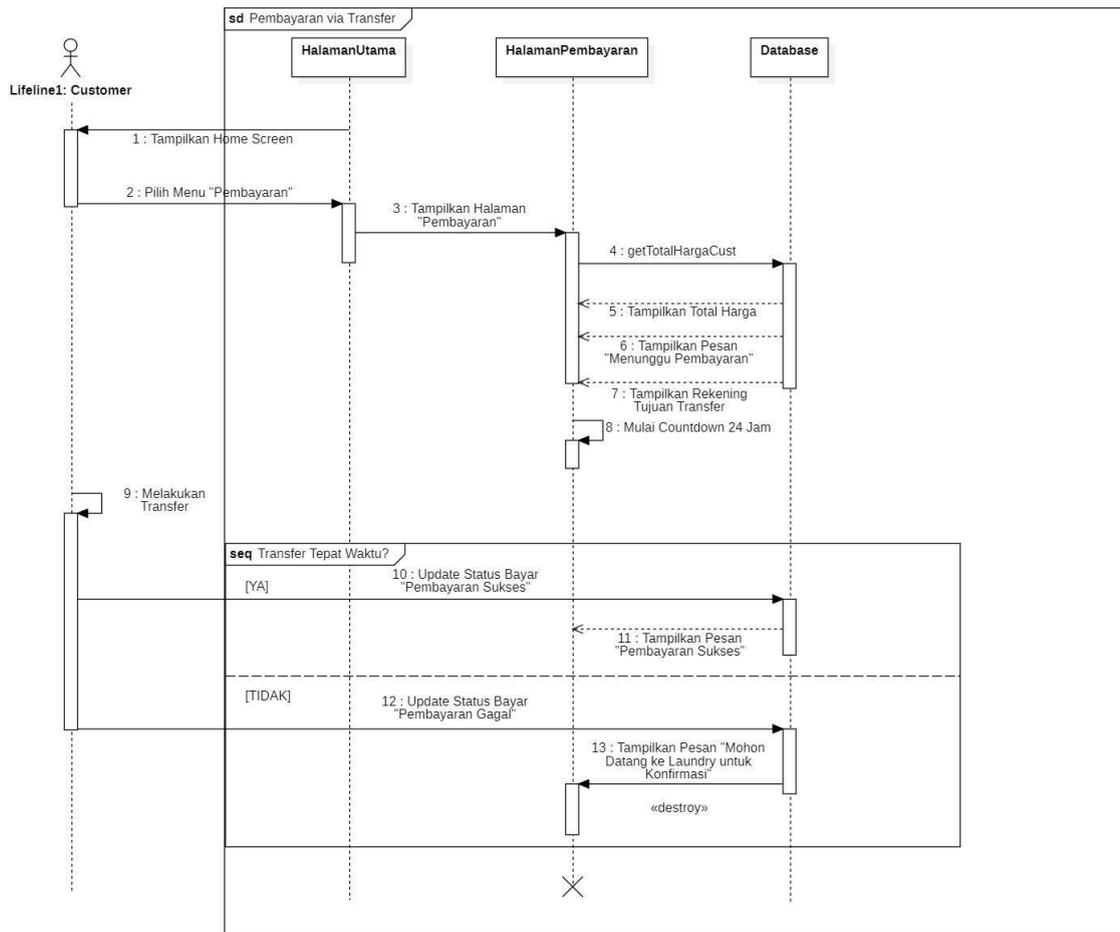


Gambar 12: Sequence Diagram (Pemesanan)

Terlihat bahwa Aktivitas ini melibatkan 2 Entitas di luar sistem yaitu Kurir dan Customer. Layaknya Sequence sebelumnya, keseluruhan Proses diawali oleh fungsi HalamanUtama yang menampilkan Home Screen dan menu-menu yang terdapat pada Sistem. Hal yang terjadi selanjutnya adalah Customer akan membuka menu “Buat Pesanan Laundry”. HalamanUtama akan memanggil Fungsi HalamanPesanan untuk menampilkan Halaman untuk membuat Pesanan dan menangani Input Data yang akan dilakukan oleh Customer.

Ketika Customer selesai melakukan Input Data, maka Data Data akan disimpan di dalam Database sehingga Data tersebut bisa digunakan dan dipanggil kembali pada Proses-Proses lainnya. Ketika Proses penyimpanan Data sukses dilaksanakan, maka HalamanListCust yang merupakan Fungsi dalam Interface Sistem dari view kurir akan mendapatkan Data Customer terbaru dari Database dan akan ditampilkan pada Page List Customer di tampilan Sistem Kurir. Selain itu, Sistem juga akan mengirimkan Notifikasi kepada Kurir bahwa ada Pesanan baru masuk ke dalam Sistem. Ketika Notifikasi ini dibuka, maka Kurir akan mendapatkan Nama, Alamat, dan Nomor Telepon Customer sehingga Kurir bisa segera menuju ke Alamat tersebut. Setelah semua Proses tersebut selesai, maka Sistem akan menampilkan reply message kepada HalamanPesanan yang sedang diakses oleh Customer berupa sebuah Pesan bahwa Pesanan sudah diterima Sistem dan Kurir sedang menuju ke Alamat sang Customer.

3.3.5. Sequence Diagram Pembayaran via Transfer

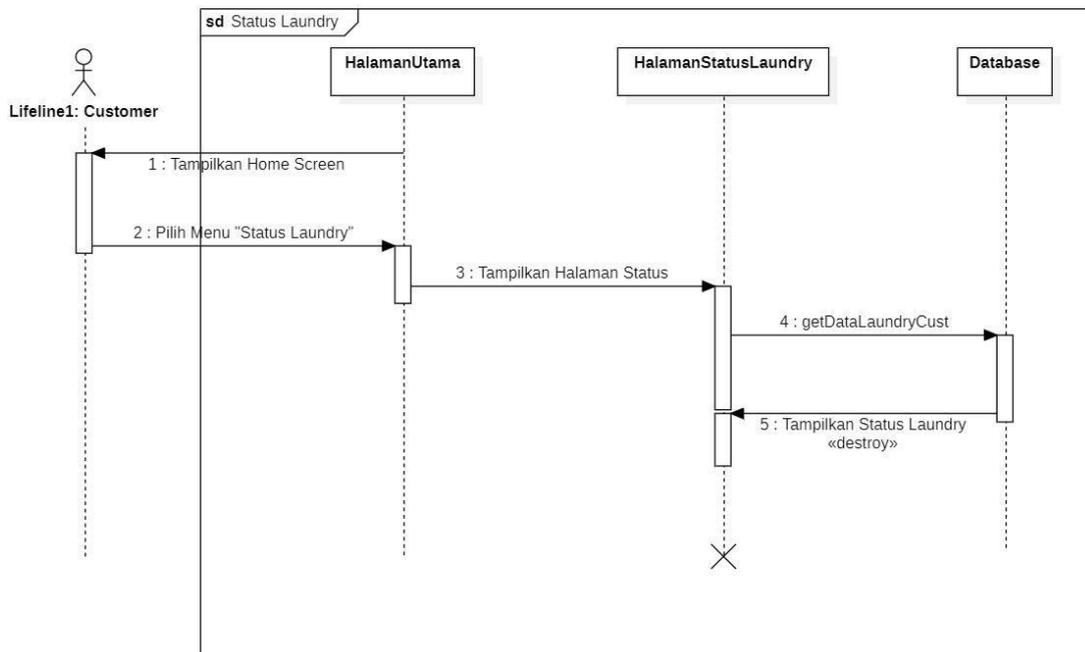


Gambar 13: Sequence Diagram (Pembayaran via Transfer)

Sequence Diagram di atas telah menggambarkan bagaimana Alur Sistem dalam menangani Pembayaran via Transfer yang dilakukan oleh Customer. 2 Peristiwa pertama yang terjadi masih sama seperti Sequence-Sequence sebelumnya yaitu HalamanUtama menampilkan seluruh menu yang dimiliki oleh Sistem, dan Customer akan memilih menu Pembayaran. Selanjutnya, HalamanPembayaran akan mencari Data Total Harga Cucian Customer yang sebelumnya telah ditentukan dan tersimpan di Database. Kemudian, HalamanPembayaran akan menampilkan Total Harga, menampilkan Status Bayar yaitu “Menunggu Pembayaran” dan memulai countdown selama 24 Jam sebagai tenggat waktu bagi Customer untuk menyelesaikan Pembayaran.

Jika Customer melakukan Pembayaran dalam kurun waktu 24 jam tersebut, maka Sistem akan update Status Bayar menjadi “Pembayaran Sukses”, memberikan reply message berupa Konfirmasi bahwa Pembayaran sudah diterima, dan Countdown pun berhenti. Jika pembayaran terjadi di luar kurun Waktu 24 Jam tersebut, maka Status Bayar akan secara otomatis di-update menjadi “Pembayaran Gagal” dan Sistem akan menampilkan Pesan bahwa Customer harus datang secara langsung ke Laundry untuk mengonfirmasi dan menyelesaikan Pembayaran.

3.3.6. Sequence Diagram Status Laundry Customer

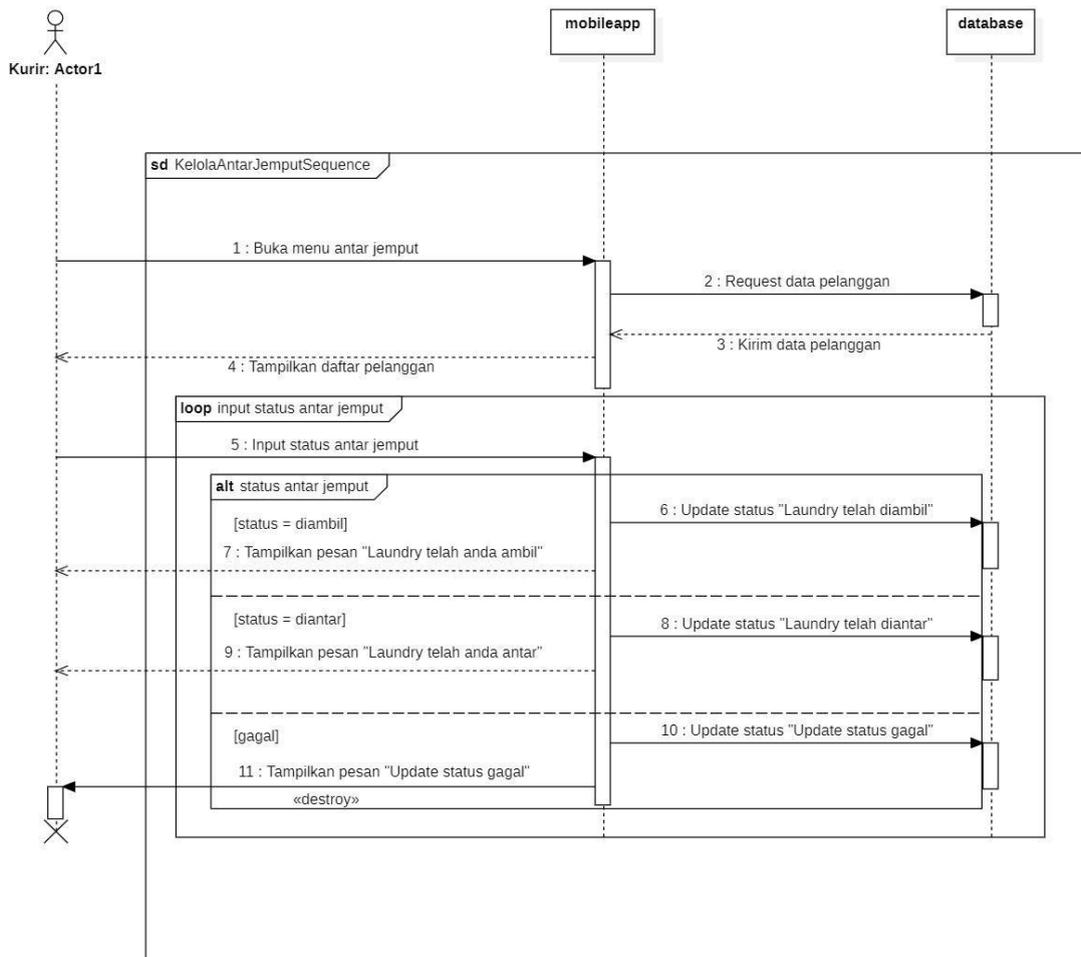


Gambar 14: Sequence Diagram (Status Laundry Customer)

Pada *Sequence Diagram* di atas telah tergambar alur komunikasi yang terjadi antara objek/fungsi dalam Sistem dengan Aktor yaitu seorang *Customer* pada saat mereka ingin melihat Status dari Laundry/Cucian mereka. Keseluruhan *Sequence* diawali oleh Fungsi *HalamanUtama* atau *Home Screen* yang menampilkan *Interface* Halaman Utama dengan berbagai menu yang dimiliki oleh Sistem. Selanjutnya, *Customer* akan memilih menu “Status Laundry”.

Setelah *Customer* memilih menu “Status Laundry”, maka *HalamanUtama* akan memanggil Fungsi *HalamanStatusLaundry* untuk menampilkan Halaman untuk menampilkan Status dari Laundry nya. Setelah itu, Fungsi *HalamanStatusLaundry* akan mencari Data dari Laundry milik *Customer* tersebut di dalam *Database*. Setelah sudah ditemukan, maka Data dari Status Laundry *Customer* akan ditampilkan pada *HalamanStatusLaundry*.

3.3.7. Sequence Diagram Kelola Antar Jemput

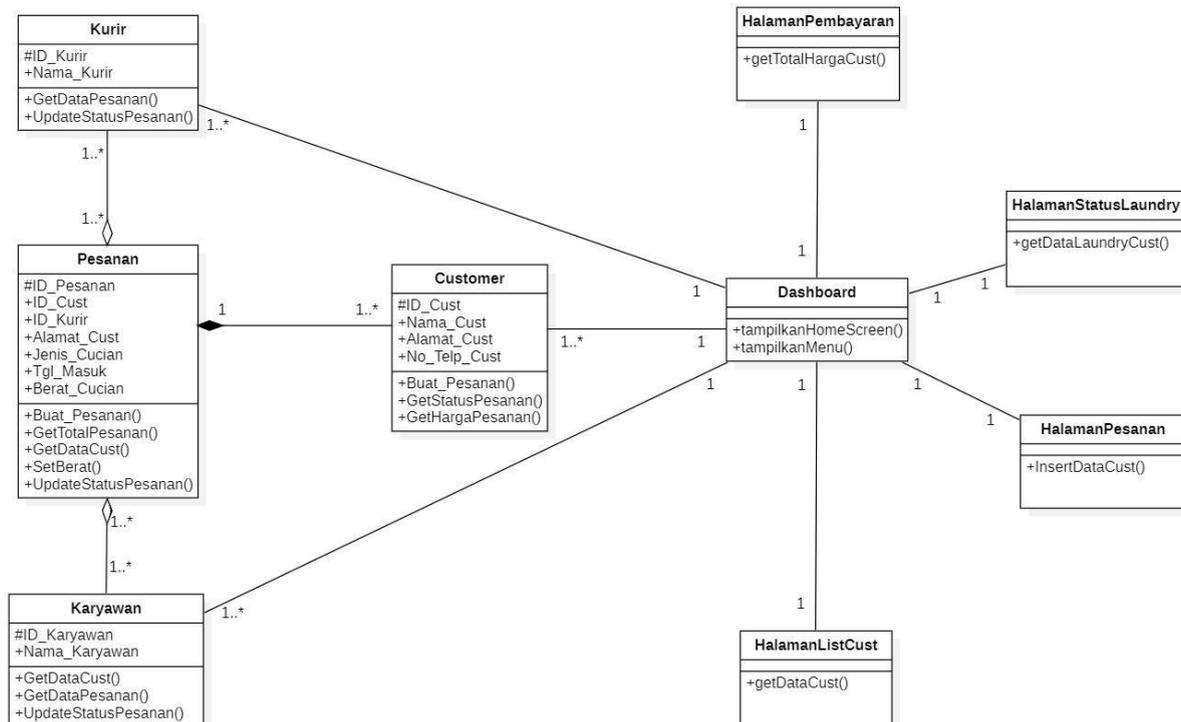


Gambar 15: Sequence Diagram (Kelola Antar Jemput)

Sequence Diagram di atas merupakan gambaran dari Kronologi interaksi dalam aktivitas pengelolaan Antar Jemput Cuci Customer yang dilakukan oleh Kurir. Proses interaksi diawali dengan Kurir membuka Menu Antar Jemput yang tersedia dalam Aplikasi/Sistem. Sebagai catatan, diasumsikan bahwa Kurir yang dimiliki Laundry berjumlah lebih dari 1 dan Sistem yang membagi pengantaran kepada para Kurir. Berdasarkan penjelasan tersebut, Sistem akan menampilkan Daftar Customer yang perlu ditangani oleh tiap Kurir setelah Menu tersebut dibuka.

Setelah menerima Data Alamat Customer, maka Kurir akan menuju ke Alamat untuk menjemput Cuci atau mengantar kembali Cuci milik Customer. Selain itu, Kurir juga harus update Status pengantaran mereka ketika sedang menjemput atau mengantar Cuci. Ini dilakukan supaya Sistem bisa mengetahui Status dari Kurir dan bisa menampilkan Informasi tersebut kepada Customer

3.4. Class Diagram



Gambar 16: Class Diagram

Class Diagram di atas adalah Diagram yang menggambarkan Class apa saja yang terdapat dalam sistem ini, sekaligus menggambarkan Relasi antar Class yang ada. Terdapat dua kelompok besar pada Class Diagram di atas, yaitu ada Class utama dan juga Class Pendukung.

Selain kelas-kelas utama seperti Customer, Pesanan, Kurir, dan Karyawan, sistem ini juga mencakup beberapa kelas pendukung yang berperan dalam tampilan antarmuka dan manajemen data melalui Dashboard dan berbagai halaman (UI pages) seperti HalamanPembayaran, HalamanStatusLaundry, HalamanPesanan, dan HalamanListCust.

Kelas Dashboard berfungsi sebagai pusat navigasi utama dalam sistem. Dashboard menyediakan dua operasi penting, yaitu tampilkanHomeScreen() untuk menampilkan layar utama, serta tampilkanMenu() untuk mengakses berbagai menu yang tersedia. Dashboard memiliki relasi langsung one-to-one terhadap masing-masing halaman UI yang disebutkan di atas, mengindikasikan bahwa dari dashboard, pengguna dapat langsung mengakses setiap halaman tersebut.

Kelas HalamanPembayaran bertanggung jawab untuk menampilkan informasi biaya kepada pelanggan. Melalui operasi getTotalHargaCust(), halaman ini mengambil total harga cucian berdasarkan data dari pesanan yang telah dilakukan oleh pelanggan.

Kelas HalamanStatusLaundry digunakan untuk menampilkan status terkini dari proses laundry. Operasi getDataLaundryCust() mengambil data status laundry milik pelanggan, yang sangat berguna bagi pelanggan maupun karyawan untuk memantau progres cucian.

Kelas HalamanPesanan menyediakan fungsionalitas untuk memasukkan data pelanggan baru ke dalam sistem, ditandai dengan operasi *InsertDataCust()*. Halaman ini kemungkinan besar digunakan saat pelanggan baru ingin mendaftar dan melakukan pemesanan pertama kali.

Kelas HalamanListCust bertugas untuk menampilkan daftar seluruh pelanggan yang terdaftar di sistem. Operasi *getDataCust()* memungkinkan pengambilan data secara menyeluruh, baik untuk keperluan manajemen oleh karyawan maupun pencarian data oleh kurir. Keterkaitan antara *Dashboard* dan semua halaman ini bersifat *one-to-one*, menunjukkan bahwa dalam konteks sistem, setiap halaman diakses secara individual dan spesifik dari *Dashboard*.

3.5. Desain User Interface

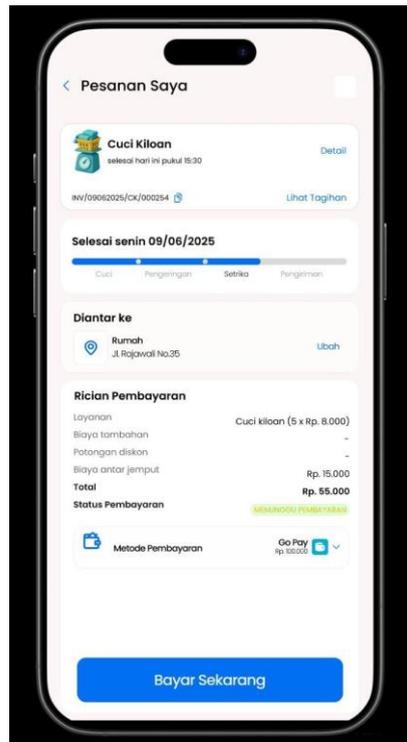
3.5.1. Halaman *Dashboard*



Gambar 17: Mockup UI Halaman *Dashboard*

Dashboard dirancang untuk memuat seluruh menu yang dimiliki oleh Aplikasi ini. Berdasarkan Gambar 17, terdapat beberapa hal utama yaitu Peta yang menampilkan Lokasi *Customer* dan Lokasi Laundry, serta Rute antara kedua lokasi tersebut. Selain itu, juga terdapat kolom untuk meletakkan promosi bagi Laundry, yang mana pada contoh diatas digunakan untuk mempromosikan Voucher Diskon 20% pada setiap pesanan ke-5. Dan terakhir, terdapat 4 Menu utama yang mencakup tentang Proses Bisnis Laundry ini. Pada kolom Persegi Panjang berwarna Biru merupakan *Call To Action* yang mengarahkan *Customer* untuk membuat Pesanan melalui Aplikasi ini. Di bawah dari Kolom tersebut, terdapat 4 pilihan dari Layanan Laundry yaitu Layanan Express, Cuci Kiloan, Cuci Sepatu, dan Setrika Saja. Terakhir, pada bagian bawah dari Halaman terdapat sebuah *NavBar* yang berfungsi sebagai *shortcut* menuju ke berbagai menu lain di Aplikasi ini. Ikon yang tersedia pada *NavBar* merupakan Ikon *shortcut* menuju *Dashboard*, Halaman Pesanan, Chat Karyawan, dan *Profile*.

3.5.2. Halaman Pesanan & Pembayaran



Gambar 18: Mockup UI Halaman pesanan dan Pembayaran

Tampilan Pesanan & Pembayaran merupakan Tampilan lanjutan dari *Dashboard*. Halaman ini berfokus untuk menampilkan detail pesanan berupa Jenis Layanan yang dipilih Customer (Cuci Kiloan/Sepatu/Express/Setrika), ID Pesanan, Status Pesanan dan Perkiraan kapan Laundry selesai diproses, Alamat Pengantaran, serta Rincian Pembayaran yang lengkap dengan harga satuan, biaya layanan, ongkos kirim, total pembayaran, status pembayaran, dan metode pembayaran yang dipilih Customer. Jika Customer menekan tombol “Bayar Sekarang”, maka sebuah *Timer* selama 24 Jam akan berjalan sehingga Customer bisa menyelesaikan Proses Pembayaran melalui pihak ketiga.

3.5.3. Halaman Chat Karyawan



Gambar 19: Mockup UI Halaman Chat Karyawan

Meskipun tidak ada pada *Activity* dan *Sequence Diagram*, fitur *Chat Karyawan* bersifat Opsional dan merupakan tambahan fitur yang akan meningkatkan kualitas pengalaman penggunaan Aplikasi. Pada Halaman *Chat Karyawan* ini, dapat terjadi komunikasi 2 arah secara real time antara *Customer* & *Karyawan*. Dengan ini, maka *Customer* dapat mengkonfirmasi sesuatu secara langsung kepada *Karyawan*, seperti jika ada pakaian tertentu yang memerlukan penanganan khusus. Begitu juga sebaliknya, dimana *Customer* dapat mengkonfirmasi kepada *Customer* jika misal suatu layanan sedang tidak tersedia.

3.5.4. Halaman Profile



Gambar 20: Mockup UI Halaman Profile

Gambar di atas merupakan visualisasi dari menu *Profile* atau *Akun* dari *Customer*. Halaman ini berfungsi untuk menampilkan dan mengubah Data-Data dari *Customer*, seperti Foto Profil, Nama Akun, dan Alamat yang tersimpan pada *Database* Sistem. Berdasarkan gambar tersebut, juga terdapat beberapa hal lain seperti menu untuk memeriksa *History* Pesanan Laundry, dan juga *Lencana* atau *Level* keanggotaan. Semakin sering mencuci pada Laundry ini, maka semakin tinggi juga Poin dan *Level* dari *Customer*.

4 KESIMPULAN

Berdasarkan Hasil dan Pembahasan diatas, Penelitian ini berhasil dalam memenuhi Tujuan dari Perancangan Sistem ini, yaitu untuk menghasilkan *Output* berupa Rancangan Sistem Informasi Pengelolaan Layanan Pesan Antar Laundry berbasis *mobile* dalam bentuk *Use Case*, *Activity*, *Sequence*, *Class Diagram*, serta Desain UI yang ramah bagi pengguna. Sistem ini dirancang untuk dapat menjawab permasalahan-permasalahan di atas, yaitu *Customer* yang sibuk sehingga tidak sempat untuk mendatangi Laundry, dan *Customer* yang tidak bisa memantau Status Laundry mereka secara *real time* melalui Layanan Pesan Antar Laundry konvensional, dengan Efektif dan Efisien. Dengan Desain UI yang modern nan ramah bagi pengguna, ini juga berperan untuk meningkatkan kecepatan dalam mencapai *Business Goals* dari Laundry. Ini membuktikan sekaligus semakin memperkuat Argumentasi bahwa di era Globalisasi dan Digitalisasi ini, Perusahaan/Instansi harus menginkorporasikan Sistem Informasi Modern dalam Proses Bisnis mereka guna menjaga tingkat kompetitif mereka di Pasaran.

REFERENSI

- [1] Alzedan, R. M. (2019). Sistem Informasi Management.
- [2] Binangkit, C. A. A., Voutama, A., & Heryana, N. (2023). Pemanfaatan Uml (Unified Modeling Language) Dalam Perencanaan Sistem Pengelolaan Sewa Alat Musik Berbasis Website.

JATI (Jurnal Mahasiswa Teknik Informatika), 7(2), 1429-1436.

- [3] Bratha, W. G. E. (2022). Literature Review Komponen Sistem Informasi Manajemen: Software, Database Dan Brainware. *Jurnal Ekonomi Manajemen Sistem Informasi*, 3(3), 344-360.
- [4] Daud, M., & Malik, A. A. (2025). Evaluating Combined Influence of Weighted Analysis Class Diagram Metrics on Early Software Size Estimation. *Computing and Informatics*, 44(1), 26-42.
- [5] Nistrina, K., & Sahidah, L. (2022). Unified Modelling Language (Uml) Untuk Perancangan Sistem Informasi Penerimaan Siswa Baru Di Smk Marga Insan Kamil. *J-SIKA| Jurnal Sistem Informasi Karya Anak Bangsa*, 4(1), 17-23.
- [6] Niqotaini, Z. (2020). PERANCANGAN SISTEM INFORMASI PENGAJUAN PERMOHONAN PEMASANGAN LISTRIK BARU PT PLN PERSERO DISTRIBUSI JAWA BARAT DAN BANTEN(Studi Kasus: Subbagian Unit Pelayanan). *SisInfo – Jurnal Sistem Informasi dan Informatika*, 2(1), pp.1–8. <https://repository.unibi.ac.id/45>
- [7] Niqotaini, Z., Arrafi, N. R., Destiana, N., Augustine, A. H., Yoshita, A. B., & Ashfiya, A. N. (2023). Perancangan Sistem Informasi Perpustakaan Man 22 Jakarta. *PROSIDING SEINASI-KESI*, 2(1), 150-158.
- [8] Niqotaini, Z., Kurniyanto, A. D., Gulo, A. K., Mulyawan, V. A. M., & Al Ghifary, M. I. (2023). ANALISIS DAN PERANCANGAN SISTEM INFORMASI PENJUALAN PADA APOTEK. *PROSIDING SEINASI-KESI*, 2(1), 143-149.
- [9] Niqotaini, Z., Yulistiawan, B. S., Krisnanik, E., & Amalia, R. D. (2023). ANALISA DAN PERANCANGAN SISTEM INFORMASI DENGAN UNIFIED MODELLING LANGUAGE. In *Indie Press* (Vol. 11, Issue 1).
- [10] Saputri, H., Kusnaedi, U., & Asmana, Y. (2023). Pengaruh Sistem Informasi Akuntansi Terhadap Kualitas Laporan Keuangan Perusahaan Jasa di Jakarta Utara. *Madani: Jurnal Ilmiah Multidisiplin*, 1(4), 102-109.
- [11] Slamet, F. A. (2022). Model Penelitian Pengembangan (R n D). *Malang: Institut Agama Islam Sunan Kalojogo Malang*.
- [12] SUWITO, R. R. (2021). *APLIKASI JASA PESAN ANTAR LAUNDRY BTP BERBASIS ANDROID* (Doctoral dissertation, ISB ATMA LUHUR).
- [13] Tanjung, A. S., & Serli, R. K. (2022). Perancangan Sistem Informasi Jasa Laundry Berbasis Web Pada Laundry Cucimania Depok. *Jurnal Informatika Upgris*, 8(1), 116-119.
- [14] Ummah, L. D. (2018). Rancang Bangun E-Commerce Pada Toko Kerudung Nuri Collection Berbasis Customer Relationship Management. *Nuansa Informatika*, 12(2).
- [15] Turban, E. (2017). *Electric Commerce 2018: A Managerial and Social Networks Perspective* (9th ed.). Springer. 10.1007/978-3-319-58715-8