

BAB II

KAJIAN LITERATUR

2.1 Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu. Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem. Adapun karakteristik dari sistem yaitu [1] :

1. Komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan.

2. Batasan sistem (*boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan luar sistem (*environment*)

Lingkungan luar sistem (*environment*) adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tepat dijaga dan yang merugikan yang harus dikendalikan, kalau tidak akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem akan menjadi masukan (*input*) untuk subsistem lain melalui penghubung.

5. Masukkan sistem (*input*)

Masukkan adalah *energy* yang dimasukkan ke dalam sistem, yang dapat berupa perawatan (*maintenance input*), dan masukkan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Contoh dalam sistem komputer program adalah *maintenance input* sedangkan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran sistem (*output*)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh komputer menghasilkan panas yang merupakan sisa pembangunan, sedangkan informasi adalah keluaran yang dibutuhkan.

7. Pengolahan sistem

Suatu sistem menjadi bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, sistem akuntansi akan mengolah data menjadi laporan-laporan keuangan.

8. Sasaran sistem

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan *input* yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

Dalam konteks pengembangan *Web Ultra Laundry*, fungsi pengolahan berperan sentral dalam mentransformasi masukan berupa data layanan menjadi keluaran informatif yang mampu menjangkau sasaran promosi secara efisien [2]. Implementasi karakteristik sistem dalam pengembangan situs *Web Ultra Laundry* memastikan bahwa setiap elemen teknis dan operasional bekerja sebagai satu kesatuan yang terintegrasi untuk mencapai tujuan bisnis. Melalui pendekatan ini, berbagai komponen seperti basis data dan antarmuka pengguna dihubungkan oleh *interface* yang memungkinkan aliran data atau *input* berupa informasi layanan *Laundry* diproses secara sistematis. Dengan penetapan batasan (*boundary*) dan pengawasan terhadap lingkungan luar digital, sistem dapat menghasilkan *output* akurat berupa media promosi yang efektif bagi pelanggan. Pada akhirnya, seluruh pengolahan di dalam sistem ini diarahkan secara konsisten untuk mencapai sasaran utama, yaitu memperluas jangkauan informasi dan meningkatkan profesionalisme layanan *Ultra Laundry* di mata publik.

2.2 Website

Website adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia, dan lainnya pada jaringan Internet. Melihat aplikasi *web* yang sedang berkembang, dapat diklasifikasikan menjadi beberapa bentuk atau kelompok. Berikut ini penjelasan beberapa kelompok aplikasi *web* dan contohnya [3]:

1. *Web Bisnis* yaitu aplikasi *web* yang didalamnya terdapat proses bisnis seperti jual beli, sewa-menyewa, penggunaan jasa, dan sebagainya. Contohnya situs yang menggunakan *website* ini adalah *e-bay.com*.

2. *Web Berita dan Informasi* yaitu *web* yang menyediakan konten informasi berbayar maupun gratis. Contohnya situs yang menggunakan *web* berita dan informasi adalah *kompas.com*, *detik.com*
3. *Web Profil* yaitu *web* yang mendeskripsikan tentang profil suatu perusahaan, lembaga, ataupun personal. Contohnya *web* ini adalah *jogjakota.go.id*
4. *Web Service* yaitu *web* yang menyediakan layanan pengolahan data dan sebagainya. Perbedaan umum *web service* tidak memiliki antarmuka, namun dapat diakses melalui alamat internet. Contohnya *web service* seperti *amaze.com*
5. *Web Social Networking* yaitu *web* yang memberikan fasilitas pertemanan, tempat, dan dapat juga menjadi wadah suatu kelompok. *web social networking* seperti *facebook.com*
6. *Web Banking* yaitu *web* yang di dalamnya terdapat proses keuangan pada perbankan secara umum, seperti transfer dana, pembayaran, pembelian dan lainnya. Contohnya *Web banking* seperti *klikbca.com*
7. *Web Search Optimiza (SEO)* yaitu *web* yang di dalamnya terdapat proses pencarian pada Internet. Contohnya *Web SEO* seperti *google.com*

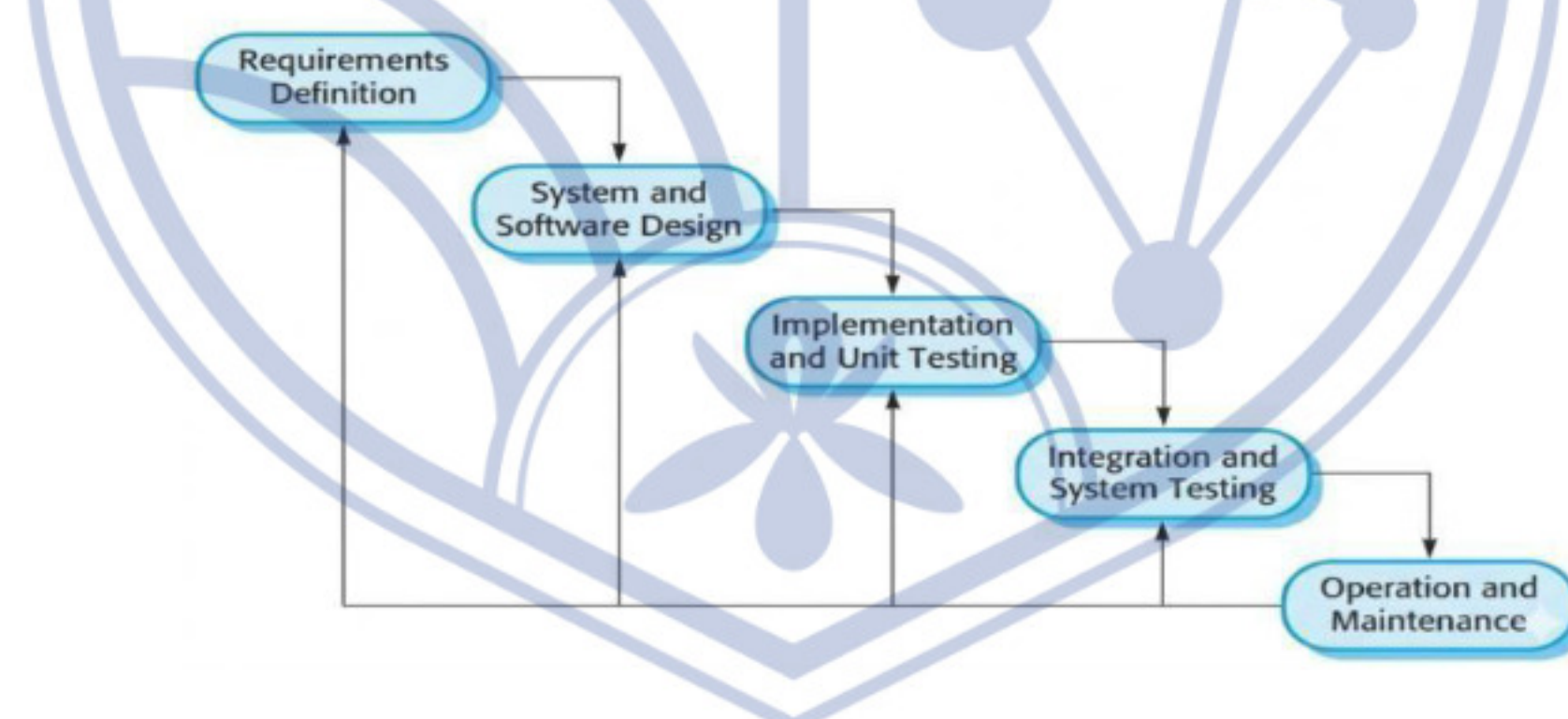
Berdasarkan pengoperasiannya, secara mendasar *website* dibagi menjadi dua jenis, yaitu *website static* dan *website dynamic* [4]:

1. *Website Static* adalah *website* yang memiliki halaman *front end*, yaitu halaman yang dapat dilihat oleh pengunjung *website*, karena fasilitas yang sangat terbatas, isi dari halaman *website static* bersifat tetap atau tidak berubah. Untuk mengganti sebuah halaman *web static* harus dilakukan secara manual dan harus mengganti semua kode- kode HTML yang merupakan unsur utama dari *website* tersebut. *website static* biasa digunakan untuk membuat *company profile* (profil perusahaan), yaitu jenis *website* pengumuman berupa browser *online* yang sangat sederhana dan tidak bisa diubah atau dimodifikasi.
2. *Website Dynamic* adalah *website* yang dapat diubah atau di-update. Dalam *website dynamic* biasanya terdapat dua halaman, yaitu halaman *fornt-end* dan *back-end*. Halaman *front-end* merupakan halaman yang dapat diakses semua *user*, sedangkan *back-end* merupakan halaman yang hanya bisa diakses oleh admin yang bersangkutan. *Back-end* biasa disebut dengan CMS (*Content Management System*) atau dalam WordPress biasa disebut dengan halaman *dashboard*. Fungsi dari halaman *back-end* adalah untuk mengatur *front-end*, sebagai contoh untuk pengaturan isi artikel. pengaturan tampilan *front-end*, hingga tambahan untuk menghias *front-end*. Beberapa contoh jenis CMS yang biasa digunakan adalah WordPress, Joomla, PHPNuke.

Pengembangan *Website* untuk Ultra Laundry dikategorikan sebagai aplikasi *web* bisnis dan profil yang berfungsi secara dinamis untuk mengintegrasikan proses promosi jasa dengan penyampaian informasi perusahaan secara *real-time* [5]. Melalui pemanfaatan teknologi *website* dinamis, sistem ini memungkinkan pengelola melakukan pembaruan konten melalui halaman *back-end* untuk memastikan data layanan yang ditampilkan pada *front-end* tetap relevan bagi pelanggan [6]. Implementasi media informasi berbasis internet ini secara strategis bertujuan untuk memperluas jangkauan pemasaran dan meningkatkan transparansi profil operasional bisnis di era digital.

2.3 Metode Waterfall

Metode Waterfall merupakan metode yang umum digunakan dalam pengembangan perangkat lunak dengan aliran sistem yang linier, di mana keluaran dari tahapan sebelumnya dijadikan sebagai masukan untuk tahap berikutnya [7]. Tahapan dalam metode Waterfall harus dikerjakan secara berurutan dan tahapan berikutnya hanya bisa dimulai setelah tahapan sebelumnya telah selesai. Oleh karena itu, metode Waterfall bersifat rekursif di mana setiap tahapannya dapat diulang terus-menerus sampai tahapan tersebut benar-benar selesai [8]. Metode Waterfall cocok digunakan ketika gambaran produk akhir sudah jelas dan kebutuhan sudah terdefinisi dengan baik dan tidak berubah-ubah [9]. Sebagian besar proyek yang menerapkan metode Waterfall adalah proyek kecil dengan tim kurang dari sepuluh orang dan berlangsung selama kurang dari enam bulan [10].



Gambar 2. 1 Metode Waterfall [7]

Dalam pengembangannya, metode Waterfall memiliki beberapa tahapan, yaitu:

1. *Requirements Definition*

Pada tahap ini dilakukan proses pengumpulan kebutuhan secara lengkap yang bisa diperoleh melalui wawancara, diskusi, atau survei langsung, yang kemudian dianalisis untuk mendapatkan kebutuhan yang dibutuhkan oleh pengguna [7] [11].

Terdapat dua kebutuhan pada tahap ini, yaitu kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional adalah uraian mengenai fungsi-fungsi yang dapat dijalankan oleh sistem, sedangkan kebutuhan non-fungsional adalah kebutuhan sistem yang meliputi kelengkapan operasi pada fungsi yang ada, kesesuaian dengan lingkungan penggunaannya dan kinerjanya [12].

Dokumen yang dihasilkan dari tahapan ini disebut *user requirement*, yang akan diserahkan kepada pengguna untuk dikaji ulang dan dijadikan acuan sistem analisis dalam tahap pengembangan perangkat lunak [7].

2. *System and Software Design*

Tahap selanjutnya adalah desain perangkat lunak, yaitu langkah yang berfokus pada desain pembuatan suatu perangkat lunak berdasarkan spesifikasi kebutuhan yang telah dikumpulkan dari tahapan sebelumnya. Tahapan ini mencakup pembuatan desain antarmuka dan perancangan basis data. Dokumen yang dihasilkan dari tahapan ini disebut *software requirement*, yang akan digunakan dalam membangun perangkat lunak dalam tahapan selanjutnya [7].

3. *Implementation and Unit Testing*

Pada tahap ini, penerapan rancangan dan model yang telah dibuat sebelumnya ke dalam bentuk program kecil yang disebut unit, kemudian digabungkan dalam tahap kegiatan berikutnya [13]. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut *unit testing* [11]. Setiap penerapan dan fitur-fitur sesuai kebutuhan selesai dilakukan uji coba dan evaluasi terkait fungsional sehingga jika terjadi eror tidak sesuai dengan kebutuhan maka segera untuk dilakukan perbaikan [13].

4. *Integration and System Testing*

Seluruh unit yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing unit. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan [11]. Setelah sistem sudah selesai dibangun, mungkin perlu dilakukan beberapa perubahan, perbaikan kesalahan, dan penyempurnaan sesuai kebutuhan. Oleh karena itu, tahap ini merupakan proses penanganan masalah-masalah tersebut [14].

Penerapan metode Waterfall dalam pengembangan situs *web* Ultra Laundry yang dibatasi hingga tahap *integration and system testing* merupakan langkah strategis untuk memastikan bahwa sistem yang dibangun telah mencapai standar fungsionalitas yang matang sebelum dioperasikan secara penuh. Proses ini diawali dengan identifikasi mendalam pada

tahap *Requirements Definition* untuk menangkap kebutuhan spesifik layanan *Laundry*, yang kemudian diterjemahkan ke dalam struktur basis data dan antarmuka pada tahap *System and Software Design*. Melalui tahap *Implementation and Unit Testing*, setiap modul fitur dikodekan dan diuji secara parsial untuk meminimalisasi kegagalan teknis pada level terkecil. Dengan menjadikan *Integration and System Testing* sebagai titik akhir pengembangan dalam konteks penelitian ini, fokus utama diarahkan pada pembuktian bahwa seluruh elemen *web* mulai dari informasi paket hingga fitur promosi telah terintegrasi secara harmonis dan bebas dari kesalahan logika sistem. Hal tersebut memberikan jaminan bahwa kerangka kerja digital yang dihasilkan sudah siap digunakan sebagai media informasi yang valid dan reliabel bagi publik.

2.4 Unified Modeling Language (UML)

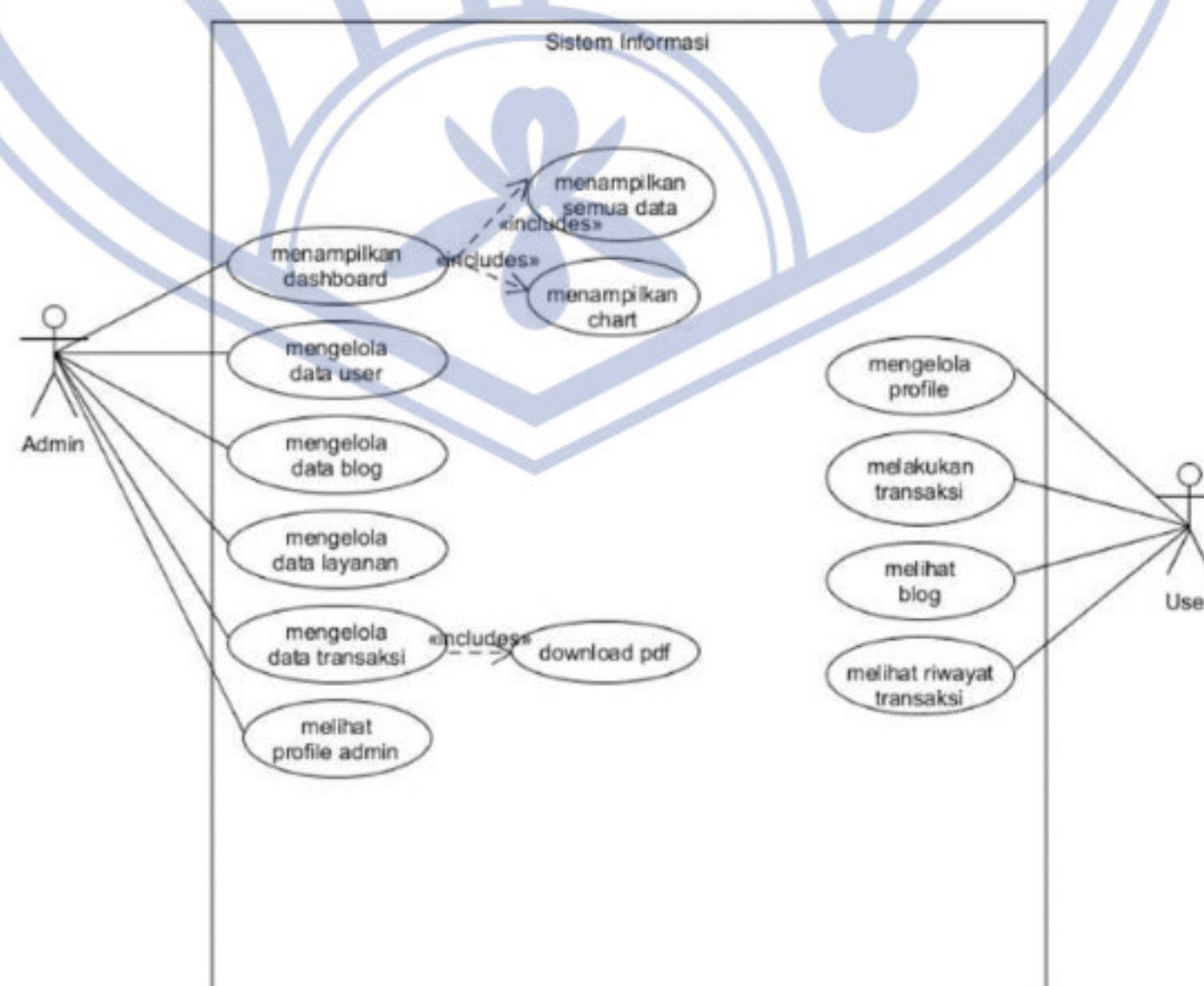
Model piranti lunak dapat dianalogikan seperti pembuatan *blueprint* pada pembangunan gedung. Membuat model dari sebuah sistem yang kompleks sangatlah penting karena tidak semua orang dapat memahami sistem semacam itu secara menyeluruh. Semakin rumit sistem tersebut, semakin krusial penggunaan teknik pemodelan yang cermat. Dengan model ini, diharapkan pengembangan piranti lunak bisa memenuhi segala kebutuhan pengguna dengan keseluruhan dan keakuratan yang optimal, termasuk aspek-aspek seperti skalabilitas, ketahanan, keamanan, dan lainnya. *Unified Modelling Language* (UML) menjadi standar industri untuk visualisasi, merancang, dan mendokumentasikan sistem piranti lunak. UML menawarkan standar untuk merancang model sistem, memungkinkan pembuatan model untuk segala jenis aplikasi piranti lunak yang dapat beroperasi di berbagai platform, sistem operasi, dan jaringan, serta ditulis dalam berbagai bahasa pemrograman [15]. *Unified Modelling Language* (UML) adalah bahasa pemodelan visual yang umum digunakan untuk spesifikasi, visualisasi, pembangunan, dan dokumentasi sistem perangkat lunak. UML berfungsi sebagai alat untuk merekam keputusan dan pemahaman tentang sistem yang akan dibangun. UML biasanya digunakan untuk memahami, merancang, mengeksplorasi, mengkonfigurasi, merawat, dan mengontrol informasi terkait sistem. UML dirancang untuk kompatibel dengan berbagai metode pengembangan, tahapan siklus hidup, domain aplikasi, dan media yang ada, dengan tujuan utama untuk mengintegrasikan pengalaman masa lalu tentang teknik pemodelan dan praktik terbaik dalam pengembangan perangkat lunak ke dalam suatu pendekatan standar. Terdiri dari bagian-bagian statis, dinamis, lingkungan, dan organisasional. UML membantu merepresentasikan informasi tentang struktur statis dan perilaku dinamis dari sebuah sistem, dengan sistem dipandang sebagai kumpulan objek yang berinteraksi secara diskret untuk mencapai tujuan pengguna. Pendekatan pemodelan dari berbagai sudut pandang

memungkinkan pemahaman yang lebih baik terhadap sistem. UML juga menyediakan struktur organisasional untuk mengelola model dalam paket yang memungkinkan bagi tim pengembangan perangkat lunak untuk membagi sistem menjadi bagian yang dapat dikelola, memahami ketergantungan antar paket, dan mengelola versi model dalam lingkungan pengembangan yang kompleks, serta menyediakan konstruksi untuk merepresentasikan keputusan implementasi dan mengatur elemen-elemen *runtime* ke dalam komponen [16].

UML merupakan "bahasa universal" rekayasa perangkat lunak yang unggul dalam mengatasi hambatan komunikasi teknis antara pengembang dan klien, tak tertandingi alat pemodelan lain. Penerapan UML sejak tahap perancangan menghasilkan dokumentasi komprehensif serta meminimalkan risiko kesalahpahaman kebutuhan sistem. Pada penelitian situs *web* Ultra Laundry, diagram UML berfungsi sebagai "peta" panduan keseluruhan proses pembangunan, memastikan setiap fitur yang diimplementasikan tepat mencerminkan kebutuhan bisnis yang teridentifikasi akurat.

2.4.1 Use Case Diagram

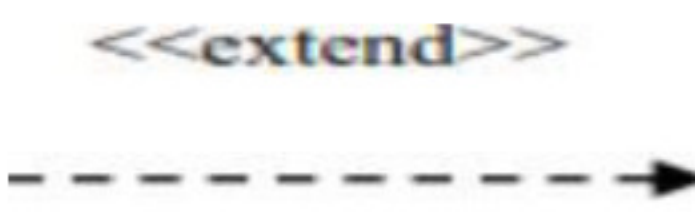
Use Case Diagram adalah representasi grafis dari interaksi antara pengguna dan *system* [17]. *Diagram* ini menggambarkan berbagai skenario di mana pengguna berinteraksi dengan fitur-fitur dalam sistem, membantu mengidentifikasi kebutuhan dan fungsi utama yang harus disediakan oleh sistem. Pada pengembangan *web* sebagai media informasi dan promosi pada Ultra Laundry. *Use Case Diagram* yang digunakan terdapat pada Gambar 2.2

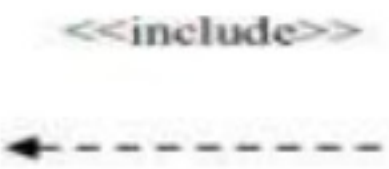


Gambar 2. 2 Use Case Diagram [18]

Use Case Diagram dalam pengembangan situs *web* Ultra Laundry berfungsi sebagai instrumen strategis untuk memetakan interaksi antara aktor, seperti admin dan pelanggan, dengan berbagai fitur yang tersedia di dalam sistem informasi tersebut. Dalam konteks analisis kebutuhan fungsional, diagram ini digunakan sebagai kerangka acuan untuk menetapkan batasan operasional sistem, sehingga setiap fungsi mulai dari pengelolaan konten promosi hingga riwayat transaksi dapat terdefinisi secara jelas sebelum masuk ke tahap teknis berikutnya. Dengan adanya visualisasi ini, arah pengembangan *web* menjadi lebih terukur dalam mengakomodasi kebutuhan pengguna, sekaligus memastikan bahwa sistem yang dibangun mampu berjalan efektif sebagai media informasi yang responsif dan sesuai dengan tujuan bisnis Ultra Laundry.

Tabel 2. 1 Simbol *Use Case Diagram* [19]

Simbol	Nama Simbol	Keterangan
	Aktor	Aktor adalah entitas manusia atau sistem lain yang berinteraksi dengan <i>use case</i> dan tidak selalu berupa orang ditandai dengan penggunaan kata benda di awal frasa.
	<i>Use Case</i>	Berfungsi sebagai unit-unit yang saling bertukar pesan dengan unit lain atau aktor, biasanya ditandai dengan kata kerja.
	<i>Association</i>	Memiliki fungsi sebagai penghubung antara aktor dengan <i>use case</i>
	<i>Extend</i>	Memiliki fungsi sebagai fungsional tambahan dari <i>use case</i> jika suatu kondisi terpenuhi.
	<i>Generalization</i>	Memiliki fungsi sebagai spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> lain.

	<i>Include</i>	Menunjukkan bahwa suatu <i>use case</i> merupakan fungsionalitas dari <i>use case</i> lainnya.
---	----------------	--

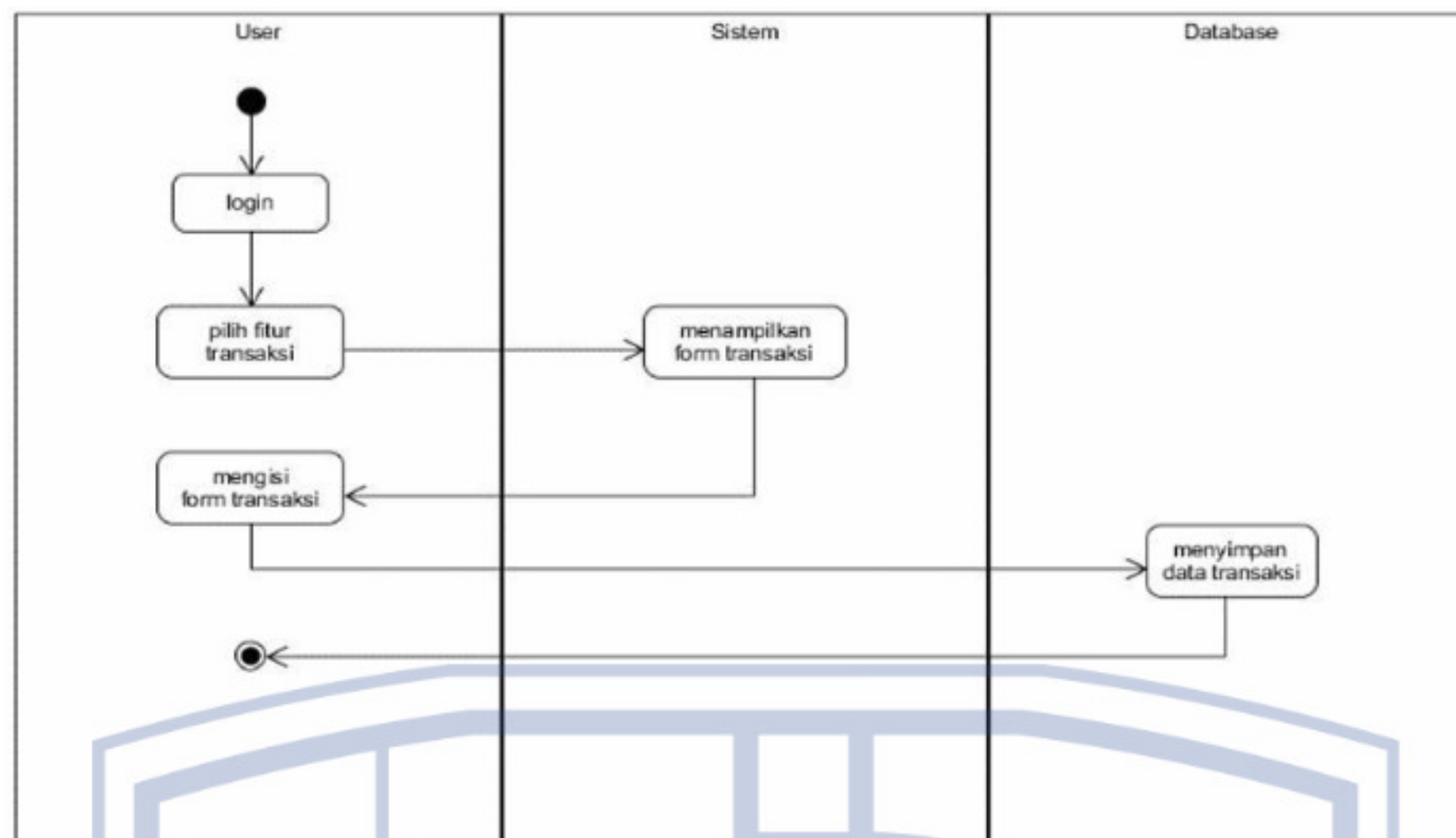
Tabel simbol ini menggunakan aktor sebagai representasi entitas eksternal, baik manusia maupun perangkat, yang berinteraksi langsung dengan fungsi-fungsi sistem di dalam *Use Case*. Hubungan komunikasi antara aktor dan fungsionalitas tersebut dipetakan melalui garis *association*, sedangkan diferensiasi peran aktor secara lebih spesifik digambarkan menggunakan relasi generalisasi. Selain itu, keterkaitan antar fungsionalitas diatur secara terperinci melalui dua jenis relasi: *include*, yang menunjukkan komponen fungsi wajib sebagai bagian dari prosedur utama, dan *extend*, yang merepresentasikan fungsionalitas opsional yang dieksekusi hanya saat kondisi prasyarat tertentu terpenuhi.

Use Case Diagram merupakan artefak perancangan paling bernilai di fase awal penelitian ini, karena merangkum kebutuhan fungsional sistem secara visual dan ringkas. Bagi pemilik Ultra Laundry tanpa latar belakang teknis, diagram ini paling mudah dan intuitif untuk memahami rancangan tim pengembang, sehingga memungkinkan masukan, koreksi, atau persetujuan sebelum coding dimulai. Dengan begitu, *Use Case Diagram* efektif mencegah *scope creep* yang berpotensi membengkakkan biaya dan waktu pengembangan.

2.4.2 Activity Diagram

Activity Diagram adalah representasi visual dari alur kerja atau aktivitas dalam suatu sistem [20]. Diagram ini bertujuan untuk mengklarifikasi konsep sistem yang akan dikembangkan dengan menunjukkan urutan dan kondisi dari berbagai langkah dalam proses. Pada pengembangan sistem manajemen layanan *Laundry*, *Activity Diagram* membantu dalam memahami dan mengoptimalkan alur kerja dari setiap proses yang ada.

Pada Gambar 2.3 *Activity Diagram* menggambarkan tahapan yang akan dilalui atau alur aktivitas dalam suatu sistem, serta terdiri atas beberapa notasi yang masing-masing memiliki makna tersendiri.

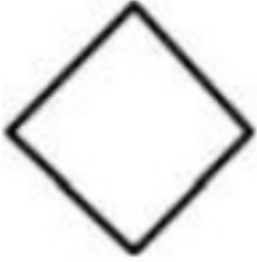
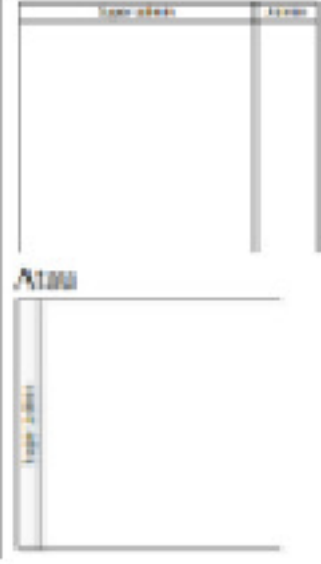


Gambar 2. 3 Activity Diagram Menjelaskan tatacara *user* dalam melakukan transaksi pada Website [18]

Proses diawali dengan *User* melakukan *Login* dan memilih fitur transaksi, yang kemudian direspon oleh sistem dengan menampilkan *form* transaksi kepada pengguna. Setelah *form* tersebut muncul, *User* mengisi data yang diperlukan dan mengirimkannya sehingga alur berpindah ke bagian *Database* untuk melakukan penyimpanan data transaksi secara permanen. Rangkaian aktivitas ini akhirnya ditutup kembali di sisi *User* yang menandakan bahwa proses transaksi telah berhasil diselesaikan sepenuhnya

Tabel 2. 2 Simbol Activity Diagram[21]

Simbol	Nama Simbol	Keterangan
	Status Awal / <i>Initial State</i>	Status awal atau <i>initial state</i> adalah suatu keadaan awal pada saat sistem mulai hidup.
	Status Akhir / <i>Final State</i>	Status akhir atau <i>final state</i> adalah suatu keadaan akhir dari daur hidup.
	Aktivasi	Aktivasi adalah suatu kegiatan yang dilakukan didalam sistem, biasanya diawali dengan kata kerja.

	Percabangan / <i>Decision</i>	Percabangan adalah suatu kegiatan dimana terdapat pilihan kegiatan didalamnya.
	Penggabungan / <i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabung menjadi satu.
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
	<i>Transition State</i>	Menggambarkan hubungan antara dua <i>state</i> , dua <i>activity</i> ataupun antara <i>state</i> dan <i>activity</i>

Tabel simbol ini mengilustrasikan komponen utama *Activity Diagram* yang dimulai dengan *Start State* dan diakhiri oleh *End State*. Alur kerja disusun menggunakan *Activity* sebagai representasi tugas dan *Transition State* sebagai penghubung antar elemen, sementara logika percabangan diatur melalui *Decision*. Selain itu, penggunaan *Fork* dan *Join* memungkinkan penggambaran proses paralel yang efisien, dengan seluruh aktivitas dikelompokkan ke dalam *Swimlane* untuk menegaskan batasan tanggung jawab setiap aktor dalam sistem.

Activity Diagram menjadi jembatan *krusial* antara konsep bisnis dan implementasi teknis pada penelitian *web* Ultra Laundry. Diagram ini mendorong pengembang berpikir sistematis dan menyeluruh tentang setiap langkah proses sebelum *coding* dimulai. Dengan rancangan cermat, *Activity Diagram* meningkatkan kualitas kode secara signifikan, karena kondisi serta skenario pengguna sudah diantisipasi awal, sehingga meminimalkan risiko *bug* yang mengganggu pengalaman pengguna situs Ultra Laundry.

2.5 PIECES

PIECES merupakan kerangka analisis yang terdiri dari enam dimensi utama, meliputi *performance* (kinerja), *information* (informasi), *economy* (ekonomi), *control* (pengendalian), *efficiency* (efisiensi), dan *services* (layanan). Instrumen ini berfungsi untuk melakukan evaluasi menyeluruh terhadap sistem informasi, mulai dari penilaian kecepatan akses dan akurasi data, hingga aspek manajemen biaya operasional, proteksi keamanan data, optimalisasi sumber daya, serta mutu layanan yang dirasakan oleh pengguna. [22]

Pendekatan yang dikenal sebagai *PIECES Analysis* ini memungkinkan organisasi untuk memetakan kendala teknis maupun fungsional secara akurat demi terciptanya solusi sistem yang lebih efektif [23]:

1. *Performance* (Kinerja),

Variabel ini mengukur kemampuan sistem dalam menyelesaikan tugasnya secara cepat dan tepat. Fokus utamanya adalah pada *throughput* (jumlah pekerjaan atau transaksi yang dapat diselesaikan dalam jangka waktu tertentu) dan *response time* (waktu tunggu yang dibutuhkan mulai dari saat *input* dimasukkan hingga hasil atau respon muncul). Sistem yang berkinerja buruk biasanya ditandai dengan proses yang lambat dan penumpukan antrean data.

2. *Information* (Informasi)

Evaluasi ini menitikberatkan pada kualitas *output* yang dihasilkan oleh sistem. Informasi yang baik harus memenuhi kriteria akurat, relevan, tepat waktu, dan lengkap. Selain itu, aspek ini juga menilai kualitas *input* (apakah data yang dimasukkan mudah atau justru membingungkan) dan penyimpanan data (apakah data tersusun rapi dan mudah diakses kembali untuk kebutuhan laporan).

3. *Economy* (Ekonomi)

Variabel ini menilai aspek finansial dan keuntungan yang diperoleh dari penggunaan sistem. Analisis ekonomi membandingkan antara biaya (*cost*) yang dikeluarkan (seperti biaya pengembangan, perangkat keras, dan operasional) dengan manfaat (*benefit*) yang diterima. Manfaat ini bisa berupa penghematan biaya operasional atau peningkatan keuntungan bisnis berkat efektivitas sistem.

4. *Control* (Pengendalian)

Sistem informasi harus memiliki mekanisme perlindungan terhadap gangguan, penyalahgunaan, atau kesalahan data. Pengendalian yang lemah dapat menyebabkan kebocoran data sensitif, sementara pengendalian yang terlalu ketat justru bisa menghambat

alur kerja. Fokusnya adalah menemukan keseimbangan agar sistem tetap aman namun tetap fleksibel bagi pengguna yang berwenang.

5. *Efficiency* (Efisiensi)

Efisiensi berkaitan dengan bagaimana sistem memanfaatkan sumber daya yang ada secara minimal untuk mendapatkan hasil maksimal. Sumber daya di sini meliputi waktu manusia, kapasitas penyimpanan, daya proses komputer, hingga bahan habis pakai. Sistem yang efisien adalah sistem yang mampu menghilangkan langkah-langkah birokrasi yang tidak perlu atau pemborosan data yang berulang (*redundansi*).

6. *Service* (Layanan).

Variabel terakhir ini menilai sejauh mana sistem memberikan kepuasan kepada penggunanya (*user-friendly*). Layanan yang baik mencakup kemudahan dalam penggunaan sistem, keandalan (sistem jarang eror), fleksibilitas dalam menangani berbagai situasi, serta kemudahan dalam pemeliharaan atau perbaikan jika terjadi kendala teknis.

Indikator-indikator yang berhubungan dengan performa disusun berdasarkan keenam variabel tersebut

Tabel 2. 3 Variabel dan indikator Pieces framework[24]

Faktor	Indeks	Pengertian
Performance	<i>Throughput</i>	Sistem dinilai berdasarkan banyaknya pekerjaan yang dilakukan pada periode waktu tertentu.
	<i>Respon Time</i>	Durasi yang diperlukan oleh sistem ketika melakukan pemrosesan perintah.
	Audibilitas	Kesesuaian performa sistem dengan standar atau persyaratan yang telah ditetapkan.
	Kelaziman Komunikasi	Tingkat kesulitan yang dihadapi pengguna dalam memahami antarmuka sistem.
	Kelengkapan	Kemampuan sistem untuk melaksanakan fungsi atau perannya secara keseluruhan.
	Konsistensi	Keselarasan sistem dalam pengelolaan data persediaan sesuai standar pelaporan barang.
	Toleransi Kesalahan	Jumlah kesalahan yang dilakukan oleh sistem.

Information dan Data	Akurasi	Informasi yang dihasilkan memiliki tingkat ketepatan yang tinggi.
	Relevansi Informasi	Kesesuaian kebutuhan pengguna terhadap informasi yang dihasilkan.
	Penyajian Informasi	Informasi disajikan dalam bentuk yang tepat dan mudah dipahami.
	Fleksibilitas Data	Kemudahan dalam mengakses suatu data yang ada pada sistem.
Economics	Reusabilitas	Sejauh mana suatu program atau bagian program dapat digunakan kembali dalam aplikasi lain.
	Sumber Daya	Jumlah sumber daya yang diperlukan dalam proses pengembangan sistem, melibatkan baik manusia maupun sumber daya ekonomi.
Control and Security	Integritas	Sistem memiliki pembatasan hak akses bagi Administrator terkait dengan pengoperasian tertentu.
	Keamanan	Memiliki mekanisme untuk menjaga keamanan data dalam sistem.
Efficiency	Usabilitas	Usaha yang dilakukan oleh pengguna saat memahami dan mengoperasikan sistem.
	Maintanabilitas	Usaha pengguna untuk menangani kesalahan yang muncul dalam sistem (perbaikan sistem).
Service	Akurasi	Ketepatan yang terjadi dalam proses kerja sistem.
	Reliabilitas	Kepercayaan terhadap performa sistem sesuai dengan yang diharapkan.
	Kesederhanaan	Kemudahan bagi pengguna dalam memahami sistem.

Metode ini dikembangkan untuk mengidentifikasi kelemahan pada sistem yang sedang berjalan serta menentukan kebutuhan fungsional bagi pengembangan sistem baru agar lebih optimal. Dalam prosesnya, variabel *performance* digunakan untuk mengukur kecepatan respons sistem, sementara variabel *information* menilai kualitas data yang dihasilkan. Variabel *Economy* meninjau nilai manfaat ekonomis, *Control* berfokus pada keamanan data, *Efficiency* mengevaluasi pemanfaatan sumber daya, dan *Service* menitikberatkan pada aspek kenyamanan serta kepuasan pengguna akhir.

Penerapan metode Pieces dalam pengembangan *web* sebagai media informasi dan promosi sangat krusial untuk menjamin bahwa platform yang dibangun mampu menyelesaikan masalah operasional secara mendasar. Dengan membedah setiap variabel, pengembang dapat memastikan bahwa transisi dari sistem manual ke sistem berbasis *web* tidak hanya mengubah cara kerja, tetapi juga meningkatkan nilai tambah bagi bisnis. Hal ini sejalan dengan penelitian terdahulu yang menyatakan bahwa kerangka Pieces mampu memberikan gambaran yang komprehensif mengenai tingkat kepuasan pengguna terhadap sebuah sistem informasi, sehingga pengembang memiliki landasan yang kuat dalam melakukan perbaikan atau perkembangan fitur yang lebih relevan dengan kebutuhan pelanggan. Teori Pieces berfungsi sebagai instrumen diagnostik dan alat ukur standar kualitas untuk memetakan GAP antara kondisi sistem saat ini dengan ekspektasi pengguna. Dalam penelitian ini, teori tersebut digunakan untuk menjamin bahwa *website* Ultra Laundry yang dikembangkan memenuhi kriteria teknis dan fungsional yang handal, efisien, serta mampu memberikan pelayanan informasi yang maksimal bagi para pelanggannya [25]

Kerangka analisis PIECES paling sesuai untuk penelitian ini karena mampu membedah masalah dari enam perspektif berbeda yang saling terkait. Bagi Ultra Laundry, yang selama ini bergantung pada promosi konvensional seperti brosur cetak dan papan nama toko, analisis PIECES menyediakan "bukti ilmiah" kuat bahwa transformasi digital melalui pengembangan *web* bukan hanya keinginan, melainkan kebutuhan mendesak yang dapat diverifikasi secara sistematis dari aspek *Performance*, *Economy*, hingga *Service*.

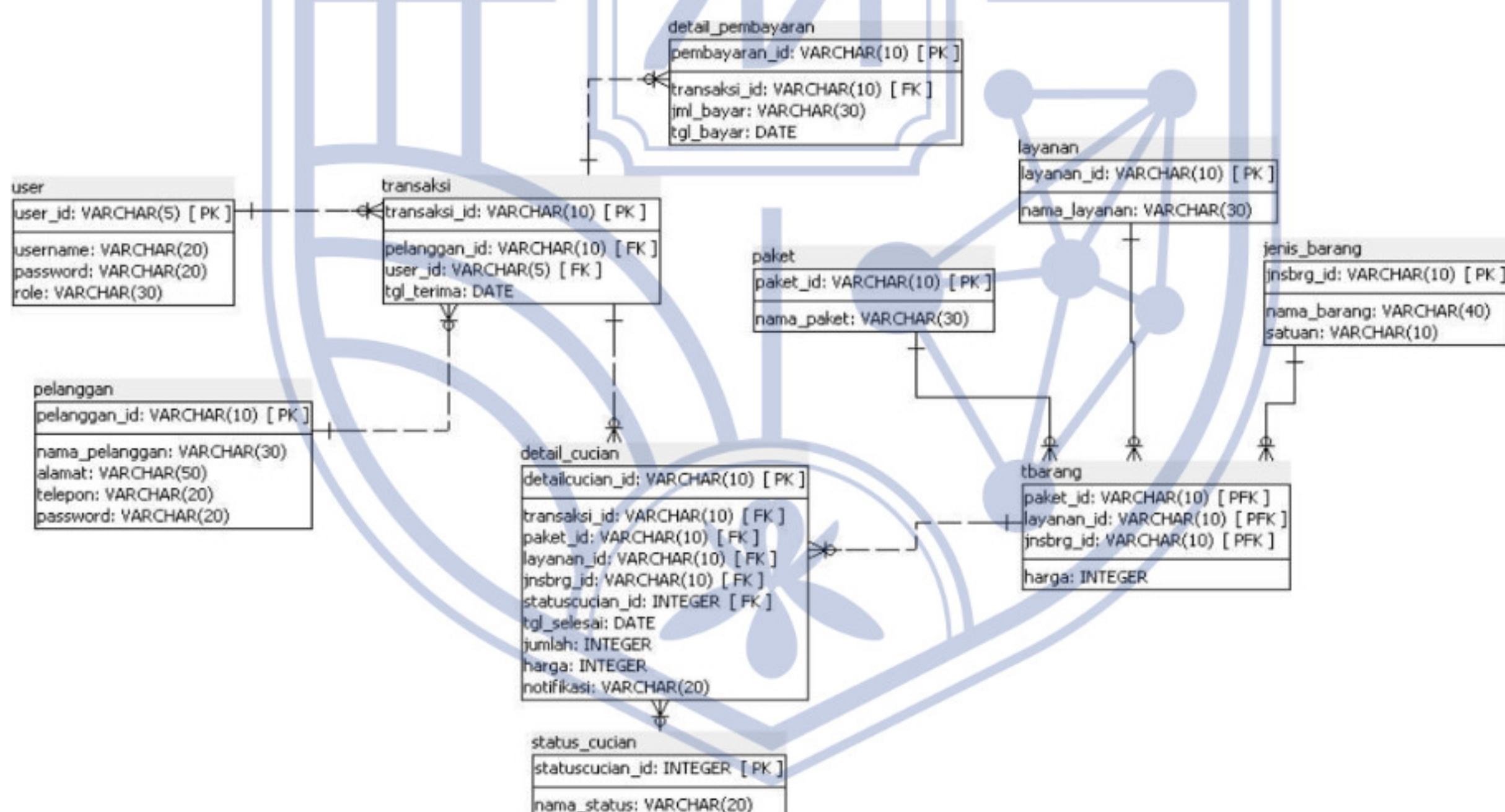
2.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah metode pemodelan yang dipakai untuk menggambarkan kebutuhan data di perusahaan atau organisasi saat tahap pengembangan sistem. Notasi grafis ini merepresentasikan pemodelan data konseptual, di mana berbagai data saling terkait digambarkan secara jelas [26]. ERD memanfaatkan struktur data yang rapi untuk mengimplementasikan basis data secara sistematis. Sebagai dasar perancangan *database* sebelum pembangunan teknis, ERD memungkinkan perencanaan hubungan antar data yang teliti, sehingga mencegah kesalahan struktural yang bisa memicu kegagalan sistem secara keseluruhan.

Saat menyusun ERD untuk desain *database*, ada poin-poin krusial yang perlu diperhatikan setiap *database* harus mencakup entitas yang saling terkait lewat relasi, dengan tiap entitas dilengkapi atribut berupa *Primary Key* sebagai pengenalan unik serta atribut

deskriptif pendukungnya. Banyak kasus kegagalan pembuatan *database* berawal dari kesalahan saat menggambar ERD, baik secara konseptual, prosedural, maupun teknis. Penyebab utama kesalahan konseptual ini seringkali adalah kurangnya penguasaan komponen ERD seperti entitas, relasi, atribut, dan kardinalitas contohnya, relasi yang seharusnya berbentuk belah ketupat malah digambarkan sebagai persegi panjang, ditambah absennya garis penghubung antar entitas beserta notasi kardinalitas hubungannya [27].

ERD berfungsi sebagai alat pendukung dalam merancang *database* sekaligus memberikan visualisasi cara kerja *database* yang akan dibangun. Tiga elemen dasar dalam ERD adalah entitas, atribut, dan relasi. Khususnya pada pengembangan *web* Ultra laundry sebagai platform informasi dan promosi, ERD memainkan peran vital karena data layanan, pelanggan, transaksi, serta promosi harus terintegrasi secara logis. Proses pembuatan ERD bersifat konseptual, dimulai dari mengidentifikasi entitas yang relevan dikenali melalui kata benda yang merujuk pada manusia, objek, tempat, proses, organisasi, atau konsep lalu dilengkapi dengan atribut sebagai ciri khas masing-masing entitas [26].



Gambar 2. 4 ERD sistem informasi pelayanan 21 Laundry [28]

Pada rancangan ERD sistem ini, terdapat 10 tabel utama, yakni *user*, pelanggan, transaksi, detail pembayaran, detail cucian, paket, layanan, jenis barang, barang, dan status cucian. Hubungan antar tabel-tabel tersebut dapat diamati melalui *diagram* ERD.

ERD lebih dari sekadar instrumen teknis berperan sebagai penghubung komunikasi antara pengembang sistem dan pemilik usaha. Pada proyek *web* Ultra Laundry, ERD memastikan struktur *database* mencerminkan proses bisnis *real* di lapangan, mulai dari pengelolaan data pelanggan, variasi layanan cuci, hingga penyusunan laporan promosi digital. Tanpa ERD yang tepat, risiko redundansi data dan inkonsistensi informasi menjadi sangat besar, yang berpotensi merusak kualitas layanan informasi yang disampaikan lewat situs *web* kepada pelanggan.

Tabel 2. 4 Simbol ERD [29]

Simbol	Nama	Keterangan
	Entitas	Sesuatu di dalam sistem atau di luar sistem yang saling berinteraksi.
	Atribut	Sifat, atau elemen dari setiap entitas.
	<i>Relationship</i>	Hubungan antara entitas1 dengan entitas lainnya.
	<i>Link</i>	Menghubungkan satu entitas dengan entitas lainnya

2.7 Promosi

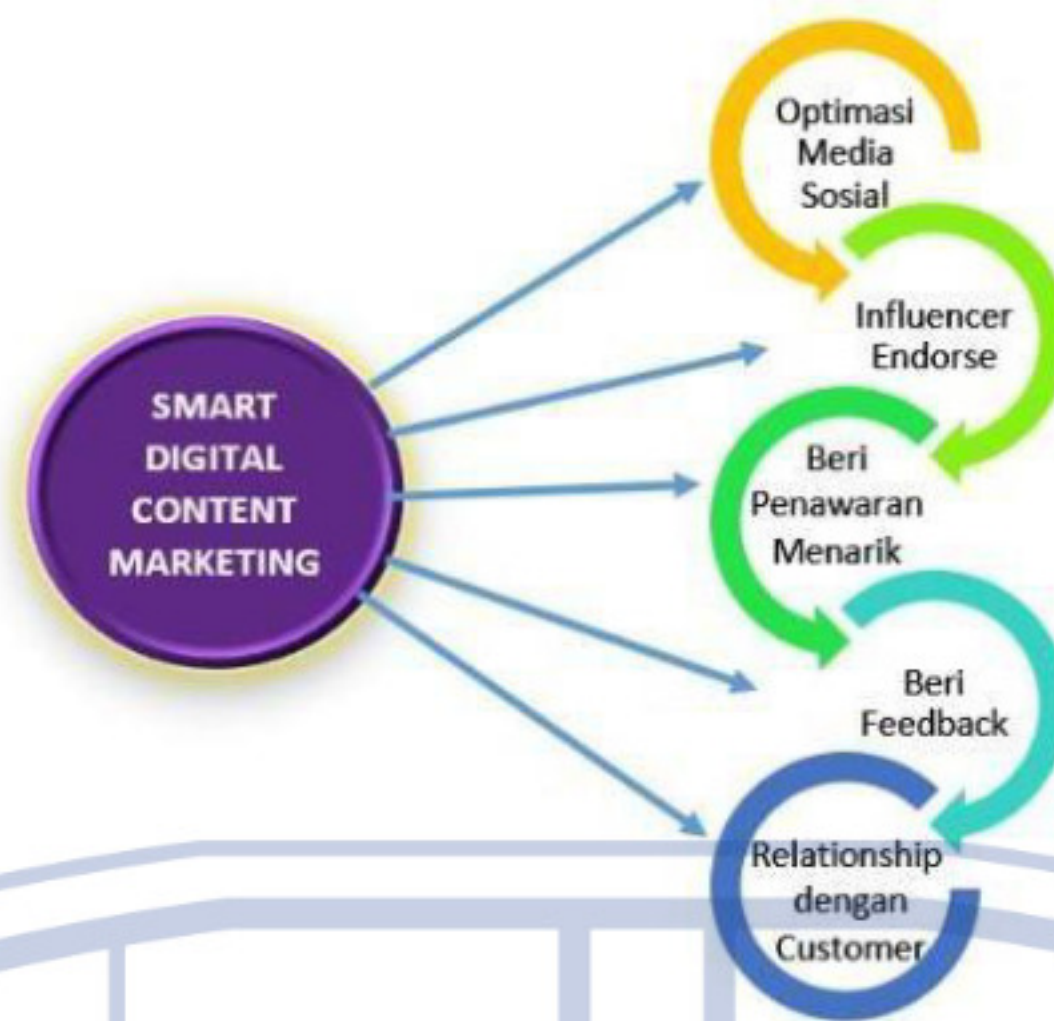
Promosi merupakan salah satu instrumen strategis dalam bauran pemasaran yang bertujuan untuk memberikan informasi, membujuk, serta memengaruhi keputusan pelanggan dalam menggunakan suatu produk atau jasa. Di era digitalisasi, promosi tidak lagi hanya mengandalkan media cetak konvensional, melainkan telah bertransformasi menggunakan platform berbasis *web* yang menawarkan jangkauan lebih luas dan interaksi yang lebih cepat. Melalui promosi digital, sebuah bisnis dapat membangun citra merek yang kuat dan menyampaikan keunggulan layanan secara visual maupun tekstual kepada calon pelanggan potensial tanpa batasan ruang dan waktu. Hal ini menjadi krusial bagi bisnis jasa seperti *Laundry* untuk tetap kompetitif di tengah persaingan pasar yang semakin ketat.

Di tengah persaingan bisnis *laundry* yang kian ketat di Kota Padang, promosi digital via *website* menawarkan investasi jangka panjang dengan ROI jauh lebih tinggi daripada metode konvensional. Situs yang menyajikan informasi promosi lengkap, terkini, dan mudah diakses berfungsi sebagai "tenaga penjual digital" 24/7 tanpa biaya operasional ekstra. Bagi Ultra Laundry dengan program promo harian beragam, *website* ideal untuk menjangkau target pasar lebih luas.

2.8 Content Marketing

Content marketing merupakan pendekatan pemasaran strategis yang berfokus pada penciptaan dan pendistribusian konten yang bernilai, relevan, dan konsisten untuk menarik serta mempertahankan audiens yang telah ditentukan secara jelas. Di era digital saat ini, strategi ini menjadi esensial karena konsumen cenderung mencari informasi yang bermanfaat sebelum melakukan keputusan pembelian. Melalui konten yang berkualitas, sebuah bisnis tidak hanya mempromosikan produk atau jasa secara langsung, tetapi juga membangun otoritas dan kepercayaan di mata konsumen. Dengan demikian, *content marketing* berfungsi sebagai jembatan komunikasi yang mampu mengubah persepsi audiens dari sekadar pembaca menjadi pelanggan yang loyal melalui penyampaian pesan yang edukatif dan solutif.

Pemanfaatan *content marketing* dalam ekosistem digital *marketing* menjadi keharusan bagi pelaku usaha, terutama UMKM, untuk menghadapi perubahan cara berkomunikasi dan perilaku pasar. Berdasarkan strategi pemasaran digital yang mencakup penggunaan *website* dan media sosial memungkinkan pelaku usaha untuk mempromosikan produk secara lebih luas dan interaktif. Konten yang disajikan dalam media internet tersebut menjadi instrumen utama dalam mengenalkan keunggulan layanan kepada masyarakat secara cepat dan efisien. Dalam konteks Ultra Laundry, penerapan strategi konten yang tepat pada *website* akan membantu menyampaikan informasi mendalam mengenai kualitas layanan, sehingga mampu meningkatkan daya tarik bisnis di tengah persaingan pasar yang semakin dinamis. Teori *content marketing* ini berfungsi sebagai landasan strategis dalam merancang narasi dan visualisasi informasi pada *website* Ultra Laundry agar mampu menarik perhatian target audiens secara efektif. Selain itu, teori ini menjadi acuan dalam menyusun strategi komunikasi digital yang bertujuan untuk membangun identitas merek (*brand identity*) yang kuat, sehingga *website* tidak hanya menjadi pusat informasi tetapi juga menjadi mesin penggerak promosi yang mampu meningkatkan keterlibatan pelanggan secara berkelanjutan.



Gambar 2. 6 Smart Content Digital Marketing [32]

Formula *Smart Content Digital Marketing* (SCDM) dianggap sangat tepat diterapkan oleh pelaku usaha ritel *online* dalam skema B2C (*business to konsumen*) di Indonesia, karena strategi ini sudah menyesuaikan dengan ciri khas pelanggan. Formula SCDM mencakup lima strategi inti berikut:

1. Memaksimalkan optimasi konten media sosial, meliputi:
 - a. Optimasi foto profil, baik foto bisnis / perusahaan maupun foto produk / jasa. Gunakan foto yang menarik dengan warna cerah, gambar tajam, serta mampu menyampaikan pesan produk / jasa secara efektif.
 - b. Optimasi konten video, yang dibuat secara kreatif tapi tetap relevan dengan narasi sesuai karakter konsumen milenial.
2. Memanfaatkan peran influencer endorse secara optimal, yaitu:
 - a. *Product / jasa review*, dengan mempekerjakan selebriti, selebgram, atau *YouTuber* (figur berpengaruh di dunia maya) untuk *mereview*, *unboxing*, dan memberikan testimoni produk yang telah mereka coba.
 - b. *Product unboxing*, melalui video *unboxing* dari selebriti, selebgram, atau *YouTuber* atas produk / jasa yang sudah digunakan.
 - c. *Mention / hashtag*, saat *review* dan *unboxing* untuk mengarahkan *followers* mereka ke toko *online* atau produk / jasa yang dipromosikan.
 - d. Menampilkan testimoni dari *influencer endorse* maupun pelanggan, yang kemudian diposting di semua akun media sosial bisnis *online*.

3. Menyediakan beragam konten promosi, seperti gratis ongkir, *voucher* diskon, atau penawaran serupa yang terbukti memikat minat konsumen untuk *Laundry*.
4. Memberikan umpan balik kepada pelanggan, yaitu:
 - a. *Memberi* ruang bagi pelanggan untuk menyampaikan kritik atau saran terkait strategi pemasaran yang dilakukan.
 - b. Mengajak pelanggan melalui *challenge* untuk ikut membuat konten *marketing* ini dianggap sebagai bentuk penghargaan, karena melibatkan dan memberdayakan mereka dalam promosi produk / jasa.
5. Memperkuat hubungan dengan pelanggan melalui interaksi

Content marketing menjadi pembeda utama *web* Ultra Laundry dalam penelitian ini dibandingkan *laundry* biasa yang hanya menyajikan informasi statis. Sementara kebanyakan *laundry* terbatas pada daftar layanan dan harga, integrasi konten edukatif berkualitas dan konsisten memposisikan Ultra Laundry sebagai "ahli *laundry* terpercaya" di mata konsumen Kota Padang. Kepercayaan dari konten informatif ini akhirnya mendorong pertumbuhan pelanggan organik tanpa biaya iklan besar.

2.9 Ultra Laundry

Ultra Laundry merupakan penyedia jasa pencucian pakaian yang beroperasi di dua lokasi strategis di Kota Padang, yaitu Jalan Raya Balai Gadang Air Dingin (Samping Perumahan Melati Mas) dan Jalan Tui Raya (Depan Bengkel Yamaha Belimbing). Dalam upaya meningkatkan loyalitas pelanggan, Ultra Laundry menghadirkan sistem poin yang dapat ditukarkan dengan berbagai diskon atau hadiah menarik. Selain itu, terdapat penawaran khusus bagi mahasiswa berupa potongan harga sebesar 20% hanya dengan menunjukkan kartu mahasiswa yang aktif. Pelanggan juga diberikan kemudahan melalui program keanggotaan (*Membership*) berbasis saldo. Tersedia empat pilihan paket, mulai dari paket *silver* hingga *diamond*, yang memberikan keuntungan berupa saldo tambahan lebih besar dari nominal yang dibayarkan. Dari sisi layanan, Ultra Laundry menyediakan kategori yang fleksibel, mulai dari layanan *express* (3-6 jam), layanan reguler (1-2 hari), hingga layanan khusus cuci lipat dan setrika. Tersedia pula layanan spesialis *Baby Care* yang menggunakan detergen khusus agar pakaian bayi tetap steril dan nyaman.

Keunggulan lain dari Ultra Laundry adalah adanya promo harian yang beragam, seperti bonus berat cucian pada hari Senin dan Sabtu, serta diskon khusus untuk item tertentu seperti perlengkapan tidur, perlengkapan ibadah, hingga perawatan sepatu dan helm pada hari-hari

yang telah ditentukan. Namun, di balik lengkapnya fasilitas tersebut, strategi pemasaran yang diterapkan saat ini masih bersifat konvensional karena hanya mengandalkan brosur fisik dan papan nama toko. Ketergantungan pada media tradisional ini menyebabkan informasi layanan menjadi kurang terorganisir dan sulit diakses oleh calon pelanggan secara cepat. Kondisi tersebut menghambat optimalisasi jangkauan pasar di era digital, sehingga potensi untuk menarik pelanggan yang lebih luas belum dapat dimanfaatkan secara maksimal. Maka akan dilakukan pengembangan *website* menjadi solusi kunci (jembatan) untuk memutus rantai masalah ini dengan mengorganisir informasi layanan secara profesional dan memperluas jangkauan pasar secara otomatis (1)



Gambar 2. 7 Alamat Ultra Laundry(1)

Kedua gambar tersebut merupakan brosur promosi dari Ultra Laundry *express* yang menampilkan daftar layanan serupa seperti cuci kiloan, satuan, sepatu, helm, tas, boneka, *stroller*, gorden, dan karpet dengan fasilitas gratis antar-jemput, namun keduanya mempromosikan dua cabang yang berbeda. Gambar pertama menunjukkan lokasi cabang di Jl. Tui Raya, depan Bengkel Yamaha Belimbing, dengan nomor WhatsApp 0812-6226-0157, sedangkan gambar kedua mempromosikan cabang di Jl. Raya Balai Gadang Air Dingin No. 24, samping Perumahan Melati Mas, dengan nomor kontak 0813-7153-1345.



Gambar 2. 8 Promo setiap hari (1)

Gambar tersebut merupakan spanduk promosi mingguan dari Ultra Laundry yang menawarkan berbagai diskon dan promo khusus berbeda setiap harinya untuk menarik pelanggan. Promo dimulai pada hari Senin dengan penawaran cuci setrika minimal 7 kg gratis 1 kg serta diskon 15% untuk jas, dilanjutkan pada hari Selasa dan Rabu dengan diskon 10% untuk perlengkapan tidur seperti *bed cover*, selimut, dan spre. Pada hari Kamis, pelanggan mendapatkan diskon 10% untuk jaket, helm, dan sepatu, sementara hari Jumat dikhususkan bagi perlengkapan ibadah seperti sajadah dan mukenah dengan diskon sebesar 20%. Rangkaian promo ditutup pada hari Sabtu dengan layanan cuci lipat minimal 5 kg gratis 1 kg dan hari Minggu yang menawarkan diskon 10% untuk gorden dan spre.



Gambar 2. 9 Layanan Baby Care (1)

Gambar tersebut menampilkan materi promosi untuk layanan khusus *Baby Care* di Ultra Laundry yang menekankan pada penggunaan bahan kimia khusus merk *Sleek Baby* untuk memastikan pakaian bayi tetap steril dan nyaman. Poster ini menyertakan daftar harga yang bervariasi tergantung pada jenis layanan dan durasi pengerjaan, mulai dari cuci setrika pakaian

bayi Reguler hingga layanan *Ekspres*. Selain pakaian, layanan ini juga mencakup pencucian perlengkapan bayi lainnya seperti selimut, bantal, gendongan hingga barang besar seperti kasur bayi, *stroller* ukuran sedang, yang semuanya ditujukan untuk Memberikan solusi kebersihan yang aman bagi kebutuhan perlengkapan bayi Anda.



PENUKARAN POIN	
✓ Disc Layanan 12 %	: 170 Poin
✓ Disc Layanan 25 %	: 320 Poin
✓ Voucher Belanja Voiho Boba	: 140 Poin
✓ Manggo Thai Voiho Boba	: 140 Poin
✓ Teh Telur / Skotang	: 170 Poin
✓ Teh Telur Tapai/Skotang Telur	: 220 Poin
✓ Gelas Cantik Ultra Laundry	: 520 Poin
✓ Vacuum Flash Set	: 720 Poin
✓ Boneka	: 820 Poin
✓ Tas	: 950 Poin

Gambar 2. 10 Sistim Penukaran Poin (1)

Gambar tersebut menampilkan daftar program loyalitas berupa penukaran poin bagi pelanggan setia Ultra Laundry, di mana poin yang telah dikumpulkan dapat ditukarkan dengan berbagai kategori hadiah menarik. Pelanggan dapat menukarkan poin mereka mulai dari jumlah terkecil untuk mendapatkan diskon layanan sebesar 12% (170 poin) atau 25% (320 poin), serta berbagai *voucher* makanan dan minuman seperti *Manggo Thai* atau Teh Telur dengan kisaran 140 hingga 220 poin. Selain diskon dan konsumsi, tersedia juga pilihan hadiah berupa barang fisik yang memerlukan poin lebih tinggi, seperti gelas cantik (520 poin), *Vacuum Flask Set* (720 poin), hingga hadiah utama berupa boneka dan tas dengan nilai penukaran mencapai 950 poin, semuanya berlaku selama stok masih tersedia sesuai syarat dan ketentuan yang ditetapkan.



PAKET MEMBER	
DIAMOND	PLATINUM
Bayar Hanya Rp.900.000	Bayar Hanya Rp.435.000
+ Bonus Rp.100.000	+ Bonus Rp.65.000
Dapat Saldo Rp. 1.000.000	Dapat Saldo Rp. 500.000
GOLD	SILVER
Bayar Hanya Rp.250.000	Bayar Hanya Rp.125.000
+ Bonus Rp.50.000	+ Bonus Rp.25.000
Dapat Saldo Rp. 300.000	Dapat Saldo Rp. 150.000

Transaksi Lebih Mudah Dengan E-Payment

Make Everything Is Better

laundry.ultra

Ultra Laundry

Gambar 2. 11 Paket Member (1)

Gambar tersebut menjelaskan program paket *member* dan kemudahan pembayaran elektronik yang ditawarkan oleh Ultra Laundry untuk meningkatkan kenyamanan pelanggan. Terdapat empat tingkatan keanggotaan dengan sistem bonus saldo, yaitu

1. Paket *Diamond* (bayar Rp900.000 dapat saldo Rp1.000.000),
2. *Platinum* (bayar Rp435.000 dapat saldo Rp500.000),
3. *Gold* (bayar Rp250.000 dapat saldo Rp300.000), dan
4. *Silver* (bayar Rp125.000 dapat saldo Rp150.000).

Selain keuntungan saldo tambahan tersebut, Ultra Laundry juga mendukung transaksi nontunai yang *modern* melalui berbagai platform *E-Payment* seperti QRIS, GoPay, DANA, ShopeePay, dan OVO, yang semuanya dihadirkan dengan identitas visual berwarna kuning khas brand tersebut lengkap dengan informasi media sosial resminya.



Gambar 2. 12 Diskon 20% Bagi Mahasiswa (1)

Gambar tersebut menampilkan poster promosi dari Ultra Laundry *Express* yang ditujukan secara eksklusif bagi kalangan mahasiswa melalui penawaran diskon 20%. Promo ini dirancang untuk memberikan keringanan biaya jasa cuci bagi pelajar perguruan tinggi dengan syarat yang sangat mudah, yaitu hanya dengan menunjukkan Kartu Tanda Mahasiswa (KTM) yang berlaku saat melakukan transaksi.

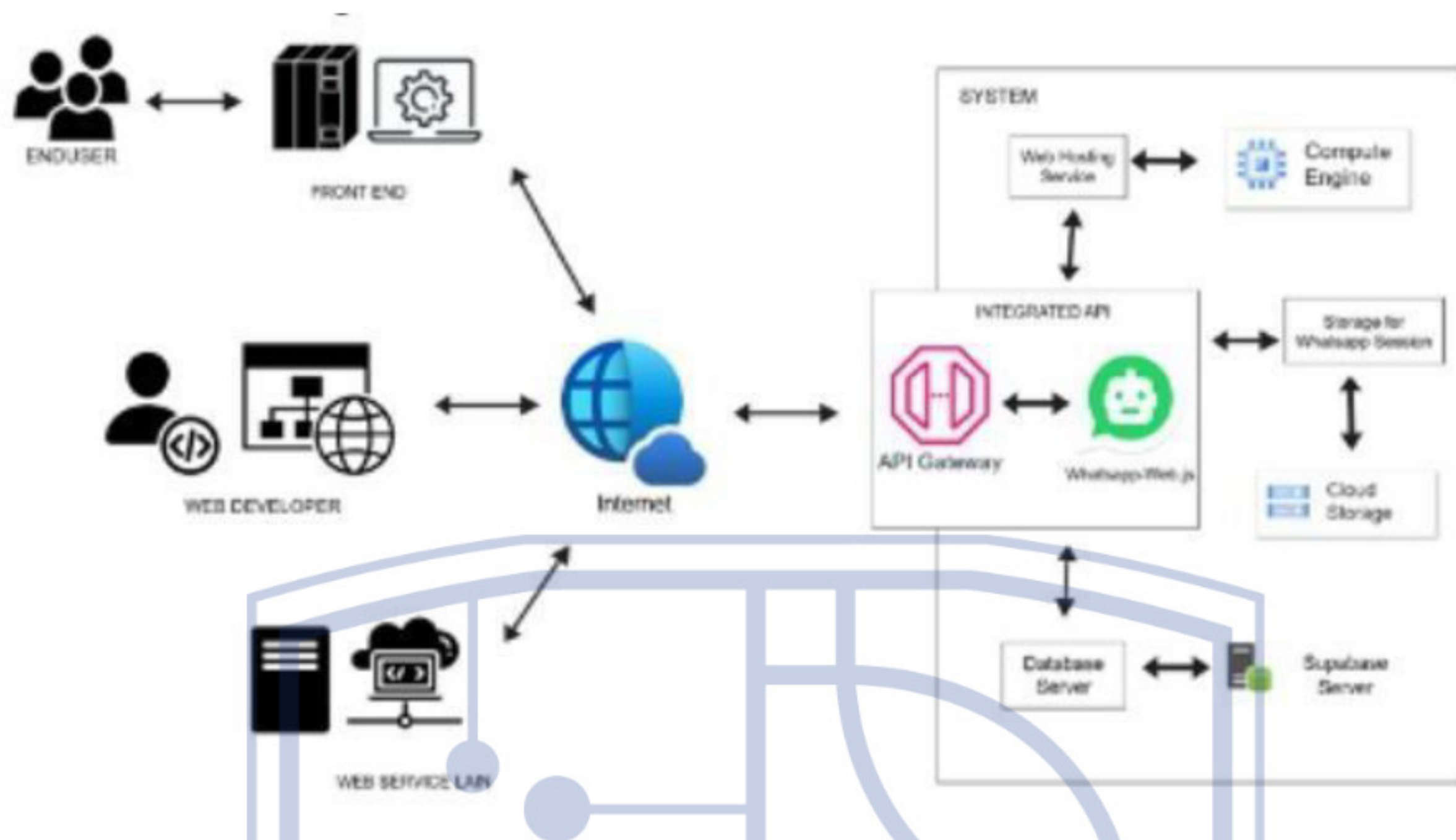
Ultra Laundry mewakili banyak UMKM jasa di Indonesia yang unggul dalam kualitas layanan, tapi terhambat strategi pemasaran tradisional. Potensinya besar berkat layanan lengkap dan program loyalitas yang melebihi kompetitor. Kehadiran *web* profesional sebagai media informasi promosi memungkinkan perluasan pangsa pasar signifikan, khususnya di kalangan mahasiswa dan keluarga muda pengguna internet paling aktif di Kota Padang.

2.10 WhatsApp

WhatsApp merupakan sebuah teknologi antarmuka pemrograman yang memungkinkan integrasi fungsi pengiriman pesan otomatis antara sebuah sistem informasi atau *website* dengan

aplikasi WhatsApp pelanggan. Penggunaan teknologi ini dalam dunia bisnis bertujuan untuk menciptakan jalur komunikasi yang lebih cepat, personal, dan efisien, mengingat WhatsApp merupakan platform pesan instan dengan tingkat penggunaan tertinggi. Dalam sistem informasi, WhatsApp sering kali diimplementasikan sebagai media notifikasi *real-time* yang dapat mengirimkan pesan secara otomatis berdasarkan pemicu tertentu dalam basis data, seperti perubahan status layanan atau pengingat transaksi, tanpa memerlukan intervensi manual dari operator secara terus-menerus.

Implementasi WhatsApp dalam layanan jasa *Laundry* memberikan keunggulan kompetitif, terutama dalam hal transparansi informasi dan peningkatan kualitas layanan pelanggan. Merujuk pada penelitian [33], integrasi notifikasi WhatsApp ke dalam aplikasi *Laundry* terbukti efektif dalam mempermudah pelacakan transaksi dan komunikasi status cucian kepada pelanggan, mulai dari status "sedang dicuci" hingga "siap diambil". Penggunaan fitur ini pada *website* Ultra Laundry dapat mengatasi kendala manajemen manual dan meminimalisir risiko penumpukan cucian yang tidak diambil oleh pelanggan. Dengan adanya pemberitahuan otomatis langsung ke ponsel pengguna, koordinasi operasional menjadi lebih teratur dan memberikan kesan profesionalisme serta kepedulian yang lebih tinggi terhadap kebutuhan pelanggan. Teori WhatsApp dalam penelitian ini berfungsi sebagai acuan teknis dalam merancang fitur komunikasi otomatis pada *website* Ultra Laundry untuk meningkatkan efektivitas penyampaian informasi. Selain itu, teori ini menjadi landasan dalam membangun sistem manajemen pelanggan yang lebih interaktif, sehingga mampu memperkuat hubungan antara pihak Ultra Laundry dengan pelanggannya melalui notifikasi yang transparan dan tepat waktu.



Gambar 2. 13 Rancangan awal arsitektur API [34]

Alur kerja arsitektur ini diawali oleh klien API, seperti sisi *front-end*, pengembang eksternal, atau layanan lain yang bertindak sebagai konsumen. Klien melakukan permintaan melalui *endpoint* yang tersedia di dokumentasi dengan memanfaatkan protokol HTTP (GET, POST, PUT, DELETE) berbasis jaringan internet. Permintaan tersebut kemudian diterima dan diolah oleh *API Gateway* yang telah terintegrasi dengan pustaka *WhatsApp-Web.js*. Dalam eksekusinya, *API Gateway* didukung oleh tiga infrastruktur utama: *Google Compute Engine* untuk manajemen *hosting*, *Google Cloud Storage* untuk penyimpanan sesi pengguna, serta *Supabase* sebagai pengelola basis data. Proses diakhiri dengan pengiriman respons balik dari *API Gateway* ke klien dalam format JSON untuk kemudian diolah lebih lanjut sesuai kebutuhan aplikasi [34]

Integrasi WhatsApp pada *web Ultra Laundry* merupakan keputusan strategis cerdas, mengingat dominasi WhatsApp di komunikasi digital Indonesia, termasuk Kota Padang. Tombol WhatsApp terintegrasi mengurangi hambatan dalam *customer journey from* pencarian informasi hingga konversi sehingga potensi pengunjung situs menjadi pelanggan aktif meningkat signifikan dan terukur.

2.11 Blackbox Testing

Blackbox Testing merupakan metode pengujian perangkat lunak yang berfokus pada fungsionalitas sistem tanpa harus mengetahui struktur internal kode atau logika program di dalamnya. Pengujian ini dilakukan dengan cara memberikan *input* tertentu pada sistem dan

mengamati *output* yang dihasilkan untuk memastikan apakah hasilnya sudah sesuai dengan spesifikasi kebutuhan yang telah didefinisikan sebelumnya. Fokus utama dari metode ini adalah untuk menemukan kesalahan dalam kategori fungsi yang tidak benar atau hilang, kesalahan pada antarmuka (*interface*), kesalahan dalam struktur data atau akses basis data eksternal, serta kesalahan performa atau inisialisasi. Dengan menggunakan pendekatan ini, pengembang dapat memastikan bahwa aplikasi dapat beroperasi dengan baik dari sudut pandang pengguna akhir (*end-user*).

Penerapan *Blackbox Testing* sangat krusial dalam menjamin kualitas sebuah sistem informasi berbasis *web* sebelum diimplementasikan secara luas. Berdasarkan penelitian [35], pengujian ini sangat efektif untuk memvalidasi elemen-elemen penting dalam sebuah *website*, seperti halaman *Login*, pendaftaran pelanggan, hingga proses konfirmasi transaksi. Melalui skenario pengujian yang komprehensif, setiap fitur pada *website* Ultra Laundry dapat dipastikan berjalan sesuai dengan harapan dan tujuan pengembangannya. Hasil dari pengujian ini memberikan indikator yang jelas mengenai tingkat kesiapan sistem, sehingga potensi kegagalan operasional saat digunakan oleh pelanggan dapat diminimalisir dan kepuasan pengguna terhadap layanan digital dapat tercapai secara maksimal. Teori *Blackbox Testing* ini berfungsi sebagai metodologi validasi fungsional untuk menguji tingkat kelayakan *website* Ultra Laundry sebelum diluncurkan. Dalam penelitian ini, teori tersebut digunakan sebagai dasar untuk menyusun skenario uji yang menjamin bahwa seluruh fitur utama, mulai dari penyajian informasi layanan hingga fitur promosi, telah bebas dari kesalahan teknis dan siap mendukung operasional bisnis secara profesional.

Fungsi utama dari penerapan teori *black box testing* dalam pengembangan sistem adalah untuk membantu memvalidasi fungsionalitas sistem secara keseluruhan dan mendeteksi adanya persyaratan yang tidak lengkap atau sulit diprediksi sejak dini. Selain itu, teknik ini berfungsi untuk menangani berbagai jenis *input*, baik yang valid maupun tidak valid, untuk memastikan ketahanan sistem terhadap kesalahan pengguna. Karena penguji tidak memerlukan pengetahuan mendalam tentang bahasa pemrograman tertentu, metode ini memungkinkan adanya independensi antara tim pengembang dan tim penguji, sehingga proses penemuan kesalahan atau ambiguitas dalam perangkat lunak menjadi lebih objektif dan efektif [36]



Gambar 2. 14 Representasi *Black Box Testing* [37]

Gambar tersebut menunjukkan alur program eksekusi: ada *requirement* yang menjadi dasar pembuatan program, lalu program menerima *input* dan menghasilkan *output*. inti gambar ini adalah hubungan antara kebutuhan, proses program, dan hasil yang keluar dari program [37]:

1. *Requirement*: kebutuhan atau spesifikasi yang harus dipenuhi program.
2. *Input*: data atau masukan yang diberikan ke program.
3. *Executable Program*: program yang sudah jadi dan bisa dijalankan.
4. *Output*: hasil yang dihasilkan setelah program memproses *input*.

Berikut adalah teknik pengujian dari *black box testing* yang umum digunakan [38]:

1. *Equivalence Partitions*

Equivalence partitions merupakan salah satu metode dalam *black box testing* yang dilakukan dengan membagi domain *input* sebuah program ke dalam beberapa kelas data untuk menghasilkan *test case*. Teknik ini mengelompokkan seluruh nilai masukan menjadi beberapa partisi, baik partisi valid maupun tidak valid. Setiap partisi kemudian dijadikan dasar penyusunan kasus uji untuk menemukan kemungkinan kesalahan pada program. Kondisi *input* umumnya berupa nilai *numerik*, rentang nilai, serta kumpulan nilai yang berkaitan dengan kondisi *boolean*.

2. *Boundary Value Analysis*

Boundary value analysis adalah teknik yang melengkapi *equivalence partitions*. Metode ini memanfaatkan nilai-nilai batas yang diperoleh dari batas kelas *equivalence* untuk menyusun kasus uji. Nilai yang diuji mencakup nilai minimum, maksimum, nilai yang berada tepat di dalam atau di luar batas, nilai *error*, serta nilai yang lazim digunakan.

3. *Cause-Effect Graphing*

Cause-effect graphing merupakan teknik desain pengujian perangkat lunak yang berfokus pada perilaku eksternal sistem. Teknik ini digunakan untuk menggambarkan hubungan logis antara kondisi *input* dan *output* dengan bantuan operator *boolean*. Nilai *input* berperan sebagai *cause* atau sebab, sedangkan nilai *output* berfungsi sebagai *effect* atau akibat. Hubungan antara sebab dan akibat ini membantu dalam penyusunan kasus uji.

4. *Decision Based*

Decision based adalah teknik yang sesuai digunakan ketika terdapat banyak *input* dan *output* yang saling berhubungan. Teknik ini menyajikan tabel yang memuat seluruh kemungkinan nilai dari berbagai kondisi. *Decision based* diterapkan saat program memerlukan pengambilan keputusan dan aturan tertentu untuk menghasilkan logika yang tepat.

Empat area utama dalam teknik ini meliputi:

1. Kondisi Awal

Kondisi awal merupakan keadaan yang terjadi sebelum proses pengambilan keputusan dilakukan. Bagian ini menggambarkan status sistem, data, atau situasi tertentu yang menjadi dasar dalam menentukan langkah berikutnya. Misalnya, apakah pengguna telah masuk ke sistem, apakah stok barang tersedia, atau apakah transaksi sudah memenuhi syarat.

2. Kondisi Masukan

Kondisi masukan adalah seluruh data atau *input* yang diterima oleh sistem untuk diproses. *Input* tersebut digunakan untuk menentukan apakah suatu aturan atau keputusan perlu dijalankan. Contohnya meliputi nilai pembayaran, jenis pelanggan, jumlah barang, atau pilihan menu tertentu.

3. Kondisi Akhir

Kondisi akhir adalah keadaan sistem setelah keputusan diambil dan seluruh proses selesai dilakukan. Bagian ini menunjukkan hasil atau status terakhir dari suatu aturan yang dijalankan. Contohnya, transaksi berhasil, stok barang berkurang, atau data tersimpan ke dalam basis data.

4. Tindakan yang dihasilkan.

Tindakan yang dihasilkan adalah aksi atau *respons* yang dilakukan sistem berdasarkan kombinasi kondisi yang telah ditentukan. Ini merupakan *output* yang muncul setelah aturan tertentu terpenuhi. Contohnya, sistem menampilkan pesan berhasil, mencetak struk, menolak transaksi, atau menyimpan data.

5. *Error Guessing*

Error guessing merupakan teknik yang didasarkan pada asumsi dan tebakan. Keberhasilan metode ini sangat bergantung pada keahlian penguji. Penguji yang berpengalaman biasanya dapat memperkirakan lokasi dan jenis kesalahan yang paling mungkin muncul.

6. *Orthogonal Array Testing Strategy (OATS)*

Orthogonal array testing strategy (OATS) adalah pendekatan pengujian yang sistematis dan berbasis statistik untuk menguji interaksi pasangan dengan membentuk kumpulan kasus uji yang efisien dari berbagai skenario. Teknik ini membantu mengurangi jumlah kombinasi pengujian, namun tetap memberikan hasil yang optimal dengan kasus uji yang lebih sedikit. OATS dilakukan dengan mengombinasikan beberapa faktor secara berpasangan agar diperoleh skenario pengujian yang relevan dan efektif.

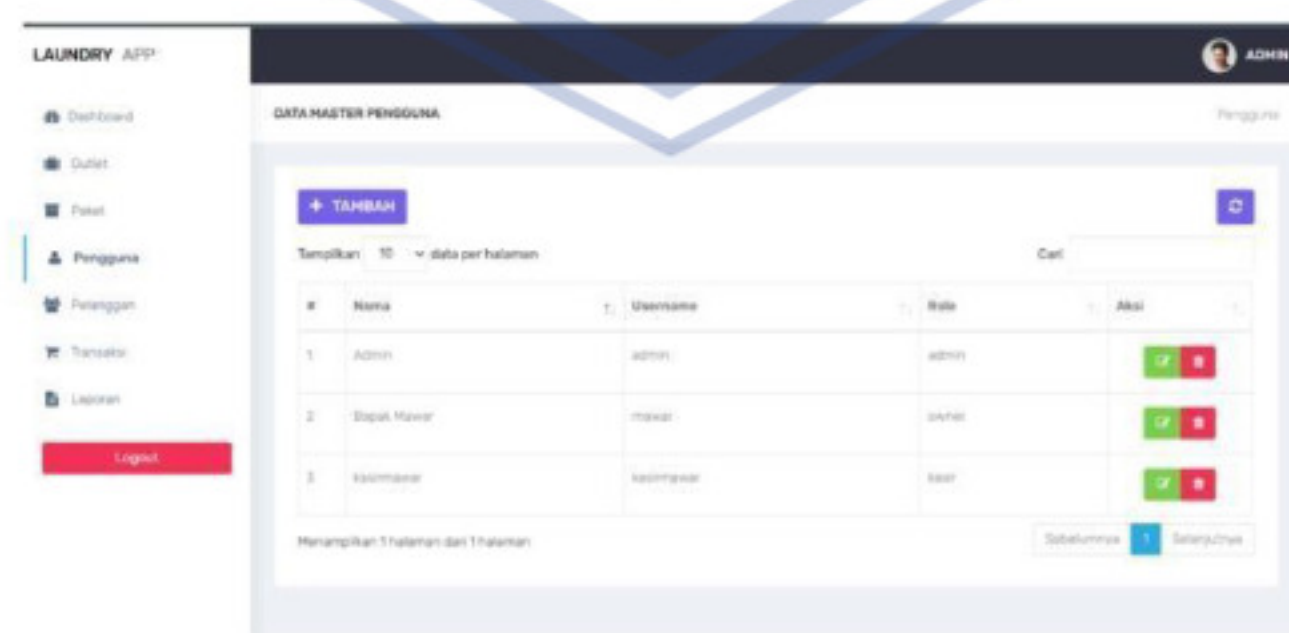
7. *Fuzzing*

Fuzz testing atau *fuzzing* adalah metode pengujian yang digunakan untuk menemukan kesalahan implementasi dengan cara memasukkan data yang salah atau tidak lengkap secara otomatis maupun semi-otomatis. Hasil *fuzzing* dapat menunjukkan bagian program yang perlu mendapat perhatian lebih lanjut, misalnya pada pemeriksaan kode, analisis statis, atau penulisan ulang sebagian program.

8. *All-Pair Testing*

All-pair testing adalah teknik *black box testing* yang menyusun kasus uji untuk mencakup seluruh kemungkinan kombinasi diskrit dari setiap pasangan parameter *input*. Karena tidak ada metode pengujian yang mampu menemukan semua *bug*, *all-pair testing* biasanya digunakan bersama teknik lain, seperti *fuzzing*, *unit testing*, dan *code review*.

Dalam pengujian *Blackbox Testing* ini halaman atau *interface* yang akan diuji yaitu halaman pengguna seperti Gambar 2.15 dibawah ini.



Gambar 2. 15 Interface Pengguna [39]

Form halaman pengguna, *form* yang dapat berfungsi sebagai menambahkan pelanggan secara manual yang dimana seorang admin menginputkan data pengguna, edit data pengguna, dan hapus data pengguna di aplikasi pengelolaan *laundry*, tampilan pada Gambar 2.15

Tabel 2. 5 Pengujian Halaman Pengguna [39]

Rencana Pengujian	Hasil yang Diharapkan
Menekan tombol “tambah”	Sistem menampilkan form <i>penginputan</i> data baru.
Menekan tombol tambah, mengisi semua <i>input</i> secara benar, lalu menekan tombol simpan	Data berhasil tersimpan ke dalam basis data.
Menekan tombol tambah, hanya mengisi satu data <i>input</i> , lalu menekan tombol simpan	Sistem menampilkan peringatan "Masukkan semua data".
Menekan tombol tambah, mengisi salah satu data dengan format yang tidak sesuai, lalu menekan tombol simpan	Sistem menampilkan pesan "Masukkan data yang benar".
Menekan tombol “edit”	Sistem menampilkan form edit data sesuai dengan baris yang dipilih.
Menekan tombol edit, mengubah salah satu data dengan benar, lalu menekan tombol simpan	Perubahan data berhasil diperbarui dan disimpan.
Menekan tombol edit, mengkosongkan salah satu data <i>input</i> , lalu menekan tombol simpan	Sistem menampilkan <i>pop-up</i> pesan "Masukkan semua data".
Menekan tombol “hapus”	Data yang dipilih akan terhapus dari sistem/basis data.

Alur pendaftaran akun Ultra Laundry dimulai saat *Customer* memilih menu daftar, direspons sistem dengan tampilan formulir pendaftaran. Setelah *input* data, Sistem memvalidasi jika tidak valid, pengguna kembali ke *form* jika valid, data otomatis tersimpan ke *database*. Proses selesai sukses setelah penyimpanan, menandakan akun berhasil dibuat.

Blackbox Testing paling cocok untuk penelitian *web* Ultra Laundry, karena memvalidasi sistem dari sudut pandang pengguna akhir calon pelanggan dan admin. Metode ini memastikan fitur intuitif bagi non-teknis, dengan *output* sesuai harapan. Pengujian sistematis via *Blackbox* menjamin *web* bebas kesalahan fungsional saat peluncuran, sehingga

mempertahankan kepercayaan pelanggan terhadap profesionalisme layanan digital Ultra Laundry.

2.12 *System Usability Scale (SUS)*

Metode *System Usability Scale (SUS)* merupakan instrumen pengukuran *usability* yang pertama kali diperkenalkan oleh John Brooke pada tahun 1986 dan telah menjadi standar dalam menilai tingkat kebergunaan sebuah sistem. Instrumen ini tersusun atas sepuluh pernyataan dengan menggunakan skala Likert lima poin, yang bertujuan untuk mengukur bagaimana pengguna mempersepsikan kemudahan, kenyamanan, serta efisiensi dalam menggunakan suatu sistem. Masing-masing pernyataan memiliki bobot penilaian yang dihitung melalui rumus tertentu, sehingga menghasilkan skor akhir dengan rentang nilai 0 hingga 100. Suatu sistem dapat dikategorikan memiliki *usability* yang "acceptable" apabila memperoleh skor di atas 68, sedangkan kategori "good usability" berada pada rentang skor 70 hingga 80. SUS tetap mampu menghasilkan pengukuran yang andal meskipun diterapkan pada jumlah responden yang relatif sedikit. Mengingat metode ini mampu menggambarkan secara menyeluruh tingkat kemudahan penggunaan serta kepuasan pengguna terhadap sistem tersebut. Berdasarkan uraian tersebut, metode SUS dipandang tepat digunakan untuk mengevaluasi persepsi pengguna terhadap berbagai aplikasi digital, termasuk aplikasi berbasis layanan seperti Laundry Keeper [40]

1. Skala Penilaian

Sebelum melakukan pengisian kuesioner, responden terlebih dahulu diminta untuk mencoba menggunakan aplikasi Laundry Keeper dalam kurun waktu beberapa menit, dengan mengeksplorasi berbagai fitur yang tersedia seperti fitur pemesanan, layanan, serta riwayat penggunaan. Setelah proses uji coba tersebut selesai dilakukan, responden kemudian diminta untuk memberikan penilaian berdasarkan pengalaman pribadi yang telah dirasakan selama menggunakan aplikasi, dengan mengacu pada skala Likert sebagai berikut:

- a. Apabila responden memilih jawaban "Sangat Tidak Setuju (STS)", maka diberikan nilai 1.
- b. Apabila responden memilih jawaban "Tidak Setuju (TS)", maka diberikan nilai 2.
- c. Apabila responden memilih jawaban "Netral (N)", maka diberikan nilai 3.
- d. Apabila responden memilih jawaban "Setuju (S)", maka diberikan nilai 4.
- e. Apabila responden memilih jawaban "Sangat Setuju (SS)", maka diberikan nilai 5.

2. Perhitungan Skor *System Usability Scale (SUS)*

Cara hitung skor SUS Adapun cara menghitung hasil skor SUS yaitu :

- Setiap pertanyaan pada urutan ganjil nilainya dikurangi dengan nilai 1.
- Sedangkan pernyataan pada urutan genap maka nilai 5 (lima) dikurangi nilai dari jawaban pertanyaan pada urutan genap.
- Skor SUS didapat dari hasil penjumlahan skor pernyataan pada urutan ganjil dan skor pertanyaan pada urutan genap yang kemudian dikali dengan 2,5.

Aturan penghitungan skor SUS hanya berlaku untuk 1 (satu) responden. Perhitungan selanjutnya akan dicari skor SUS tiap responden dengan menjumlahkan semua skor dan membaginya dengan jumlah responden. Berikut rumus untuk menghitung skor SUS [41] :

$$\bar{x} = \frac{\sum x}{n} \dots \dots \dots (2.1)$$

Dimana :

$\bar{x}$: Skor Rata – rata SUS

$\sum x$: Jumlah Skor SUS

n : Jumlah Responden

Tabel 2. 6 Kriteria Interpretasi Skor[42]

Interval Persentase	Kriteria
0% - 20%	Sangat Tidak Layak
21% - 40%	Tidak Layak
41% - 60%	Cukup Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

Tabel 2.6 menyajikan kriteria interpretasi skor yang digunakan untuk menilai kelayakan suatu produk atau instrumen berdasarkan persentase hasil penilaian. Kriteria interpretasi ini terbagi ke dalam lima kategori, yaitu sangat tidak layak untuk *interval* 0%-20%, tidak layak untuk *interval* 21%-40%, cukup layak untuk *interval* 41%-60%, layak untuk *interval* 61%-80%, dan sangat layak untuk *interval* 81%-100%. Semakin tinggi persentase skor yang diperoleh, semakin tinggi pula tingkat kelayakan yang dihasilkan, sehingga tabel ini berfungsi

sebagai acuan dalam menentukan kategori kelayakan berdasarkan hasil perhitungan skor yang telah dilakukan.

