

BAB II

KAJIAN LITERATUR

2.1 Sistem Informasi

Sistem informasi adalah seperangkat komponen yang saling terkait yang mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi, dan kontrol dalam suatu organisasi. Dengan menggunakan sistem informasi, sebuah perusahaan bahkan di beberapa bidang bisnis lainnya memerlukan sistem informasi dengan tujuan agar dapat menghemat biaya operasional, mengurangi risiko kesalahan, serta meningkatkan kerja harian [11]. Sistem informasi menggunakan *human* sebagai sumber, *hardware*, *software*, data dan jaringan untuk melakukan *input*, *output*, proses, penyimpanan, dan pengendalian data menjadi informasi. Data pertama yang diterima akan diubah menjadi format yang cocok untuk diproses (*input*). Data kemudian diolah dan dijadikan informasi, disimpan untuk digunakan nanti, atau disajikan kepada *user* [12]. Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi agar dapat menyediakan kepada pihak luar organisasi dengan laporan-laporan yang diperlukan [13]. Sistem informasi menyiratkan pengumpulan data yang terorganisasi beserta tata cara penggunaannya yang mencakup lebih jauh daripada hanya penyajian [14]. Sistem informasi telah banyak diterapkan dalam *platform website* dan juga aplikasi *mobile* untuk mendukung integrasi yang efisien antara data dan pengguna, sehingga memungkinkan akses informasi secara cepat dan meningkatkan pengalaman pengguna melalui antarmuka yang responsif. Dengan adanya platform ini, tim dapat bertukar dan mengelola data secara *real-time* yang mempercepat proses pengambilan keputusan dan mendorong inovasi dalam bisnis. Dengan integrasi teknologi ini, pengguna dapat mencari dan memperoleh data dari mana saja, sehingga meningkatkan daya tanggap dan efisiensi operasional.

2.2 Gym

Gym adalah kegiatan olahraga pembentukan otot-otot tubuh/fisik yang dilakukan secara rutin dan berkala yang bertujuan untuk menjaga kebugaran tubuh dan dilakukan menggunakan alat angkat beban yang diangkat sesuai bobot kekuatan otot seseorang. Olahraga memainkan peranan penting dalam hidup dan kehidupan bermasyarakat. Nilai-

nilai dalam olahraga sangat terkait dengan tradisi budaya masyarakat yang diwariskan secara turun-temurun dari satu generasi ke generasi lainnya. Karena itu, olahraga merefleksikan nilai-nilai sosial dalam masyarakat. Dengan adanya *gym centre*, manusia dapat melakukan aktivitas fisik dengan maksimal tanpa memerlukan lapangan atau tempat yang luas. Peralatan dan program latihan yang lengkap dapat diperoleh dengan bergabung di salah satu *gym centre* yang ada. Kini *gym centre* menjadi kian populer bahkan sudah menjadi gaya hidup dikalangan masyarakat yang telah menyadari pentingnya berolahraga [15].

Berikut ini adalah Data Pengguna *Program Membership Fitness Center* [16]:

Tabel 2.1 Data Pengguna *Program Membership Fitness Center* [16]

Tahun	Jumlah Pelanggan <i>Program Membership</i>	Jumlah Pelanggan <i>Program Membership</i>	Jumlah Pelanggan <i>Program Membership</i>
2020	45.005	511	44.494
2021	36.067	402	35.665
2022	25.183	285	24.898

Sumber : *Fitness Center* (2022)

Berdasarkan Tabel 2.1 menunjukkan bahwa selama 3 (tiga) tahun terakhir jumlah pengguna program *membership* mengalami penurunan, lebih dari 1000 orang yang tidak memperpanjangnya. Hal ini mengindikasikan fluktuasi pada loyalitas pelanggan *fitness centre*. Program *membership* memberikan pelayanan seperti potongan harga, penggunaan fasilitas yang terbaik, penawaran menarik, serta kelas-kelas senam yang dapat diikuti bagi *member*-nya.

Hasil pra riset kepada 30 *program membership* aktif menunjukkan loyalitas pelanggan pusat kebugaran rendah, dengan menyebarkan kuisioner kepada 30 responden pertanyaan “Apakah anda tertarik memperpanjang keanggotaan anda?”. Data yang diperoleh menunjukkan sebanyak 21 (dua puluh satu) responden tidak tertarik memperpanjang keanggotaan [16].

Dan juga dengan adanya *membership*, seseorang yang mendaftar keanggotaan di pusat kebugaran untuk menggunakan fasilitas olahraga dalam jangka waktu tertentu [17]. Proses pendaftaran keanggotaan umumnya melibatkan beberapa langkah sederhana. Pertama, calon anggota *gym* harus memilih paket keanggotaan yang sesuai dengan keinginan, biasanya tersedia dalam berbagai opsi seperti harian, bulanan, atau tahunan. Setelah itu, calon anggota *gym* akan diminta untuk mengisi formulir pendaftaran dan

menyelesaikan pembayaran, baik secara langsung di tempat atau melalui sistem *online* yang disediakan.

Keanggotaan *gym* memberikan berbagai manfaat bagi pengguna. Dengan menjadi anggota *gym*, pengguna memiliki akses ke fasilitas *gym* sesuai dengan paket yang dipilih. Selain itu, banyak pusat kebugaran menawarkan layanan tambahan seperti kelas olahraga, sesi pelatihan *personal*, atau diskon khusus untuk pembelian suplemen dan peralatan olahraga. *Membership* juga memotivasi anggota untuk berolahraga secara rutin karena dengan komitmen untuk memanfaatkan fasilitas yang telah tersedia dan membantu anggota *gym* mencapai tujuan kebugaran secara konsisten.

Personal trainer meningkatkan kesadaran masyarakat akan pentingnya olahraga turut meningkatkan juga kebutuhan akan pelatih pribadi (*personal trainer*). Pada penelitian [18] menyebutkan bahwa *personal trainer* adalah seorang pemimpin dari satu atau sekelompok orang yang ingin mencapai suatu tujuan dari kegiatan latihan fisik dari *fitness centre*. *Personal trainer* memiliki tanggung jawab atas tercapainya tujuan latihan fisik yang diminta *client*. *Personal trainer* memiliki tanggung jawab untuk membimbing dan memantau hasil *client* dalam berlatih. Seorang *personal trainer* harus dapat membantu menentukan latihan yang tepat bagi *client* dan harus terlatih serta memiliki sertifikat dari organisasi kebugaran ternama. *Personal trainer* bekerja untuk menilai tingkat kebugaran, mengatur *program* untuk membuat *client* tetap termotivasi dalam berlatih [18]

2.3 Course Fitness

Course adalah program terstruktur yang berfokus pada materi pembelajaran tertentu dan biasanya singkat, *course* biasanya adalah kelas atau rencana pembelajaran tentang subjek tertentu.

Course biasanya juga biasanya dapat dipelajari oleh masyarakat dan *course* juga dapat merujuk pada bidang olahraga seperti *fitness* dan beberapa contoh pelatihan yang ada di olahraga. Hipotesis diuji dengan *one sample t-test*. Adapun instrumen latihan WFH (*work from home*) adalah bentuk latihan beban untuk kekuatan otot dengan memanfaatkan beban sendiri dan peralatan yang ada di dalam rumah, sampel akan melakukan latihan empat kali dalam seminggu sampai 3 bulan sehingga akan terlihat perubahannya. Instrumen dalam WFH yang digunakan dalam proses penelitian sebagai berikut [19]:

Latihan 1

- Latihan dada: *Push Up Regular, Push Up Diamond, Incline Push Up, Decline Push Up, Push Up Wide*. [20].

- Latihan perut: *Sit Up, V Sit Up, Crunch, Russian Twist, Back Up, Side Bent Dumbel.*
- Latihan *biceps*: *Bicep Curl, Barbel Curl* [20].

Latihan 2

- Latihan paha: *Squat, Step Up Down, Front Lunges, Calf Raise, Riverse Raise, Wall Seat, Zigzag Lunges.*[19].
- Latihan *triceps*: *Tricep Exetention, Tricep Deep, Tricep Kick Back* [19].

Latihan 3

- Latihan bahu : *Barbel Srug, Single Dumbel Srug, Upright Row, Dumble Lateral, Dumble Front Raise, Lateral Raise, Shoulder Srug* [19].
- Latihan sayap : *Bent Over Row, Lat Pull Down, Single Arm Dumbbel* [19].

Tabel 2.2 Statistik Deskriptif [19]

Keterangan	N	Pretest				Posttest			
		Min	Max	Mean	Standard Deviation	Min	Max	Mean	Standard Deviation
Lingkar Dada	31	42	142	77.39	17.062	42	152	80.16	17.915
Lingkar Lengan	31	12	59	28.13	10.623	12	69	29.68	10.691
Lingkar Paha	31	28	77	50.39	10.484	28	87	52.33	11.071

Dari tabel 2.2 berupa hasil pengujian hipotesis terdapat perbedaan rata-rata lingkar dada yang signifikan sebelum dan sesudah penerapan WFH sebesar 77,39 dan 80,16. Untuk data yang lain terdapat perbedaan rata-rata lingkar lengan yang signifikan sebelum dan sesudah penerapan WFH yaitu sebesar 28,13 dan 29,68. Kemudian hasil untuk lingkar paha terdapat perbedaan rata-rata sebelum dan sesudah penerapan WFH sebesar 50,39 dan 52,33. Dari data di atas untuk lingkar dada, lingkar lengan, dan lingkar paha mengalami perubahan signifikan sebelum dan sesudah penerapan WFH dikarenakan dengan proses latihan yang rutin dan dengan menaikkan intensitas latihan (menaikkan repetisi, *set* dan beban) akan meningkatkan massa otot dada dan itu terlihat dari ukuran lingkaranya. Disimpulkan dari data di atas bahwa penerapan WFH dapat mempengaruhi hipertrofi otot secara signifikan [19].

2.3.1 Course Fitness Alat Beban

Course fitness dengan menggunakan alat beban juga harus terstruktur karena jika salah dalam melakukan gerakan akan membuat tubuh kita cedera dan harus dilakukan sesuai dengan arahan profesional dan juga dapat melakukan pembelajaran melalui *course*. Latihan beban merupakan sebuah aktivitas fisik yang sering digambarkan sebagai aktivitas yang hanya dapat dilakukan dengan hanya berkunjung ke tempat-tempat tertentu seperti pusat kebugaran, taman, dan lain sebagainya. Padahal, dengan memanfaatkan berbagai alat dan media yang ada, seseorang tetap mampu untuk melakukan aktivitas latihan beban dimana saja dan kapan saja, termasuk di rumah masing-masing [20]. Dengan pemanfaatan alat *dumbbell* dan media yang tersedia, sebab-sebab kendala seperti alasan kondisi keuangan yang tidak mencukupi sehingga tidak menjadi *member* pada tempat pusat kebugaran tertentu, kondisi lingkungan yang tidak strategis ataupun kondisi-kondisi darurat yang mengharuskan seseorang untuk tetap berada di rumah seperti maraknya penyebaran virus, dan lain sebagainya, tidak akan menjadi kendala yang cukup berarti. Dengan demikian, jika dapat pemanfaatan alat dan media yang ada sebenarnya dapat sangat membantu dan menjadi sangat diperlukan agar tujuan latihan beban dapat tercapai [20]. Selain melakukan penilaian terhadap kelayakan produk model latihan, para ahli/pakar juga memberikan masukan agar produk dapat di evaluasi dan diperbaiki kembali sehingga menjadi produk model latihan yang sesuai. Dalam produk model latihan berbasis *dumbbell* untuk mahasiswa ini. Validasi/uji kelayakan model dilakukan oleh 2 ahli/pakar angkat beban yang dapat dilihat pada tabel berikut.

Tabel 2.3 Nama Para Ahli Dalam Uji Validasi [20]

No.	Nama	Jabatan
1	Drs.Iwan Barata, M.Pd.	Dosen FIK UNJ/Dosen Ahli Angkat Beban
2	Dwi Indra Kurniawan, M.Pd.	Instruktur Fitness MAG UNJ

Adapun kesimpulan dari uji ahli yang dilakukan dapat dilihat pada tabel berikut.

Tabel 2.4 Data Hasil Penelitian [20]

No.	Nama	Nilai Rata-Rata
1	Drs.Iwan Barata, M.Pd.	86.15%
2	Dwi Indra Kurniawan, M.Pd.	86.90%
Hasil		87.50%

Berdasarkan hasil uji kelayakan di atas, dapat ditarik kesimpulan bahwa model latihan beban berbasis *dumbbell* untuk mahasiswa layak dan dapat digunakan sebagai produk model latihan dengan persentase rata-rata sebesar 87,5%. Latihan beban merupakan salah satu cara dalam meningkatkan kebugaran dan kesehatan, selain itu latihan beban juga dapat mencapai suatu tujuan tertentu. Kegiatan ini dapat dilakukan dalam waktu yang singkat untuk menghasilkan atau membuat perubahan yang dramatis terhadap tubuh. Latihan beban juga dapat memberikan manfaat seperti pengencangan otot, pembentukan tubuh, menjaga kepadatan tulang supaya tidak keropos [21]. Untuk mengetahui gerak teknik penggunaan alat latihan beban *member* Rembang *Body Fitness* antara lain seperti *bench press*, *lat pull down*, *bent arm fly*, *seated row*, *leg press*, dan *standing press* ditinjau dari segi kesesuaian. Keberhasilan suatu tujuan latihan dapat dilihat dari cara penggunaan alat beban yang baik dan benar, sehingga dapat menghindari terjadinya cedera.

Tabel 2.5 Analisis Kesesuaian Gerak Teknik Keseluruhan Alat Latihan Beban [21]

No.	Analisis Gerak Pada Alat Beban	Mean	Kriteria
1	<i>Bench Press</i>	85	Baik
2	<i>Bent arm fly</i>	84	Baik
3	<i>Lat Pull Down</i>	86	Baik
4	<i>Seated Row</i>	74	Baik
5	<i>Leg Press</i>	87	Baik
6	<i>Standing Press</i>	87	Baik

Analisis di atas menunjukkan bahwa kesesuaian gerak teknik alat latihan beban *bench press* memiliki nilai rata-rata 85 dengan kriteria “baik”, *bent arm fly* memiliki nilai rata-rata 84 dengan kriteria “baik”, *lat pull down* memiliki nilai rata-rata 86 dengan kriteria “baik”, *seated row* memiliki nilai rata-rata 74 dengan kriteria “baik”, *leg press* memiliki nilai rata-rata 87 dengan kriteria “baik”, *standing press* memiliki nilai rata-rata 87 dengan kriteria “baik”.

2.4 Extreme Programming (XP)

Extreme Programming (XP) adalah metode pengembangan perangkat lunak yang menekankan komunikasi yang intensif antar anggota tim dan kolaborasi dengan pengguna dan memberikan sebuah fleksibilitas kepada pengguna. Metode XP didesain agar dapat

mengakomodasi perubahan kebutuhan yang sering terjadi dalam proses pengembangan. Metode ini berfokus pada iterasi pendek dan pengujian terus-menerus untuk menghasilkan produk yang berkualitas tinggi. Metode XP sangat cocok diterapkan pada proyek yang memiliki kebutuhan yang tidak stabil dan berubah-ubah. Metode XP adalah salah satu metode tangkas yang paling banyak digunakan dan menjadi pendekatan yang sangat terkenal. Tujuan metode XP adalah untuk meminimalkan kebutuhan pembentukan tim yang besar, sehingga cocok untuk digunakan oleh tim yang kecil atau menengah. Hal ini dimaksudkan untuk mengatasi persyaratan yang tidak jelas dan perubahan persyaratan dengan sangat cepat [22]. *Extreme Programming (XP)* merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai dengan skala besar, serta metode ini juga sesuai jika tim dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan-perubahan *requirement* yang sangat cepat [23]. Metodologi pengembangan perangkat lunak yang pengembangannya berfokus pada pengkodean merupakan aktivitas inti dalam semua fase siklus pengembangan perangkat lunak [24].

Extreme programming (XP) memiliki kerangka pelaksanaan pembangunan produk dalam bentuk perangkat lunak. Fase pengembangan perangkat lunak dengan *Extreme programming* (XP) meliputi:

Gambar 2.1 Tahapan Pada *Extreme Programming* (XP) [25]

Dari tahapan pengembangan metode XP diatas, berikut penjelasan dari setiap tahapan:

1. *Planning* (Perencanaan)

Tahap perencanaan dimulai dengan memahami konteks bisnis dan aplikasi, mendefinisikan keluaran, karakteristik aplikasi, fungsi aplikasi yang dibuat, dan alur pengembangan aplikasi. Disimpulkan bahwa pada tahapan ini menentukan fungsionalitas keseluruhan yang akan dikembangkan dalam sistem [25].

2. *Design* (Perancangan)

Pada tahap perancangan, dilakukan pembuatan model sistem berdasarkan hasil analisis kebutuhan. Pada tahap ini peneliti membuat tampilan *use case diagram*, *activity diagram*, *sequence diagram* tentang aplikasi *mobile* berbasis *android*, mulai dari menu, penambahan alternatif, dan lainnya. Lalu peneliti membuat algoritma program tersebut. Pada perancangan sistemnya menggunakan UML (*Unified Modelling Language*) [26].

3. *Coding* (Pengkodean)

Pada penelitian Melinda et al., menjelaskan *coding* atau pengkodean adalah penerjemahan dari perancangan dalam bahasa pemrograman yang dikenali oleh komputer [27]. Tahapan pengkodean adalah tahapan yang menyiapkan kode pada perangkat lunak yang dapat digunakan dalam pengembangan aplikasi sehingga dapat menjadi pemecahan masalah [28].

4. *Testing* (Pengujian)

Tahapan pengujian merupakan tahapan terakhir untuk menguji layanan atau fitur dan fungsionalitas yang terdapat pada aplikasi yang dibangun [29]. Sehingga dapat mengambil kesimpulan dari pengujian yang dilakukan [30]. Metode *extreme programming* memiliki keunggulan daya tarik yang menjadikan metode ini layak dipakai karena keunggulannya yakni fleksibilitas dan kemampuan beradaptasi yang tinggi, meningkatkan kualitas, dan keterlibatan pengguna dalam melakukan pengembangan terhadap produk yang dipastikan dapat memenuhi kebutuhan pengguna. Berikut merupakan analisa nilai terhadap metode *extreme programming* yang telah diimplementasikan selama proses pengembangan sistem berlangsung dapat dilihat pada Tabel 2.6 [29].

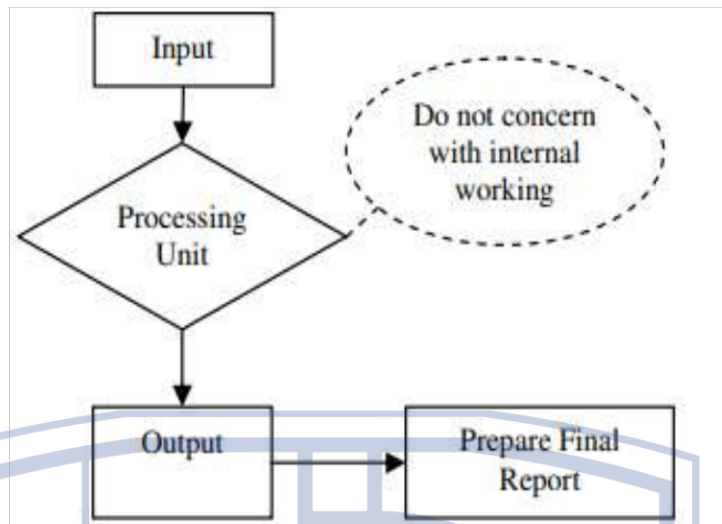
Tabel 2.6 Analisa Nilai Metode *Extreme Programming*

Variabel	Analisa
<i>Communication</i>	Komunikasi dengan klien memudahkan pengembang untuk mengetahui sistem yang diinginkan klien (<i>user stories</i>).

	Komunikasi terjadi setiap kali sebuah <i>user stories</i> atau fitur selesai dikerjakan.
<i>Courage</i>	Keberanian diterapkan pada saat adanya permintaan penambahan fitur oleh klien. Hambatan yang terjadi ketika penambahan fitur yaitu pengembang harus menerima setiap perubahan, sehingga ketika terdapat fitur yang diminta dirasa cukup sulit untuk dikerjakan, maka akan mengakibatkan waktu pengerjaan yang dibutuhkan akan menjadi lebih lama.
<i>Simplicity</i>	Fase ini merupakan kesederhanaan yang diterapkan pada saat proses <i>design</i> . Bagi pengembang dilakukan perancangan yang sederhana bermaksud untuk mengurangi waktu proses pengerjaan. Dan kesederhanaan juga diterapkan pada saat <i>coding</i> , agar baris <i>code</i> yang dibuat mudah dipahami.
<i>Feedback</i>	Timbal balik termasuk dalam komunikasi, karena pada <i>feedback</i> dari klien inilah pengembang mengetahui ada atau tidaknya masukan yang akan diberikan terkait dengan sistem diinginkan.

2.5 Black-Box Testing

Black-Box Testing merupakan salah satu metode pengujian perangkat lunak yang menekankan pada validasi fungsionalitas tanpa memperhatikan struktur internal sistem [31]. Pengujian ini dilakukan berdasarkan spesifikasi dan persyaratan yang telah ditentukan, dimana penguji hanya menguji *input* dan *output* tanpa mengetahui proses internal yang terjadi di dalam sistem. Dalam metode ini, penguji fokus pada bagaimana sistem merespons berbagai *input* yang diberikan dan membandingkan hasil *output* yang dihasilkan dengan *output* yang diharapkan sesuai dengan spesifikasi [32].



Gambar 2.2 Proses Kerja *Black-Box Testing* [33]

Karakteristik utama dari *Black-Box Testing* adalah pendekatannya yang berbasis spesifikasi. Pengujian dilakukan tanpa memerlukan pengetahuan tentang kode atau implementasi internal, sehingga penguji hanya perlu memahami fungsionalitas yang diharapkan dari perangkat lunak. Oleh karena itu, metode ini sering digunakan oleh penguji yang tidak terlibat langsung dalam pengembangan kode, seperti penguji sistem atau penguji pengguna akhir, yang hanya melihat hasil akhir dari perangkat lunak berdasarkan kebutuhan fungsional yang telah ditentukan.

Beberapa teknik yang sering digunakan dalam *Black-Box Testing* antara lain [33].

- *Equivalence Partitioning* : Teknik ini membagi *input* ke dalam beberapa partisi atau kelas ekivalensi yang diasumsikan memiliki perilaku sistem yang sama, sehingga pengujian dapat dilakukan lebih efisien.
- *Boundary Value Analysis* : Teknik ini menekankan pengujian pada nilai-nilai batas dari *input* untuk memastikan sistem berfungsi dengan baik pada titik-titik batas *input* yang *valid* maupun yang tidak *valid*.
- *Decision Table Testing* : Teknik ini menggunakan tabel keputusan untuk menguji semua kemungkinan kombinasi kondisi dan tindakan dalam sistem, sehingga semua skenario pengujian dapat tercakup.
- *State Transition Testing* : Teknik ini digunakan untuk memvalidasi perubahan kondisi atau *state* dalam sistem berdasarkan *input* tertentu, memastikan bahwa transisi antar *state* berjalan sesuai dengan spesifikasi.

- *Error Guessing* : Teknik ini mengandalkan intuisi dan pengalaman penguji dalam menebak area-area yang mungkin berisiko atau rentan terhadap kesalahan dalam sistem.

Keuntungan dari *Black-Box Testing* meliputi:

- Tidak memerlukan pengetahuan teknis tentang implementasi kode, sehingga dapat dilakukan oleh penguji tanpa latar belakang teknis mendalam.
- Efektif dalam memvalidasi apakah fungsionalitas perangkat lunak sudah sesuai dengan spesifikasi atau persyaratan yang diberikan.
- Dapat diterapkan pada berbagai tingkat pengujian, mulai dari *unit testing* hingga *acceptance testing*.

Namun, *Black-Box Testing* juga memiliki beberapa kelemahan, di antaranya:

- Tidak mampu mengevaluasi kualitas dan efisiensi dari kode internal, sehingga pengujian performa atau keamanan kode tidak bisa dicakup secara mendalam.
- Beberapa jalur logika yang kompleks dalam kode mungkin tidak diuji dengan cukup *detail*, karena pengujian hanya dilakukan berdasarkan *input* dan *output* yang terlihat dari luar.
- Pengujian ini rentan melewatkan *bug* yang terkait dengan implementasi internal atau optimasi performa, karena tidak mengeksplorasi struktur internal sistem.

Dengan mengaplikasikan uji coba *blackbox*, diharapkan dapat menghasilkan sekumpulan kasus uji yang memenuhi kriteria berikut [34]:

- Kasus uji yang berkurang, jika jumlahnya lebih dari 1, maka jumlah dari uji kasus tambahan harus didesain untuk mencapai uji coba yang cukup beralasan.
- Kasus uji yang memberitahukan sesuatu tentang keberadaan atau tidaknya suatu jenis kesalahan, daripada kesalahan yang terhubung hanya dengan suatu uji coba yang spesifik *Equivalence Partitioning Equivalence* merupakan metode uji coba *blackbox* yang membagi domain *input* dari program menjadi beberapa kelas data dari kasus uji coba yang dihasilkan. Kasus uji penanganan *single* yang ideal menemukan sejumlah kesalahan (misalnya: kesalahan pemrosesan dari seluruh data karakter) yang merupakan syarat lain dari suatu kasus yang dieksekusi sebelum kesalahan umum diamati.

Dibawah ini merupakan rumus menghitung persentase pengujian yang berhasil. Rumus ini digunakan dalam pengujian Black Box untuk mengetahui cakupan fungsional terhadap kebutuhan pengguna [35].

$$\text{Coverage (\%)} = \frac{\text{Total Test Case}}{\text{Jumlah Test Case Yang Berhasil}} \times 100\%$$

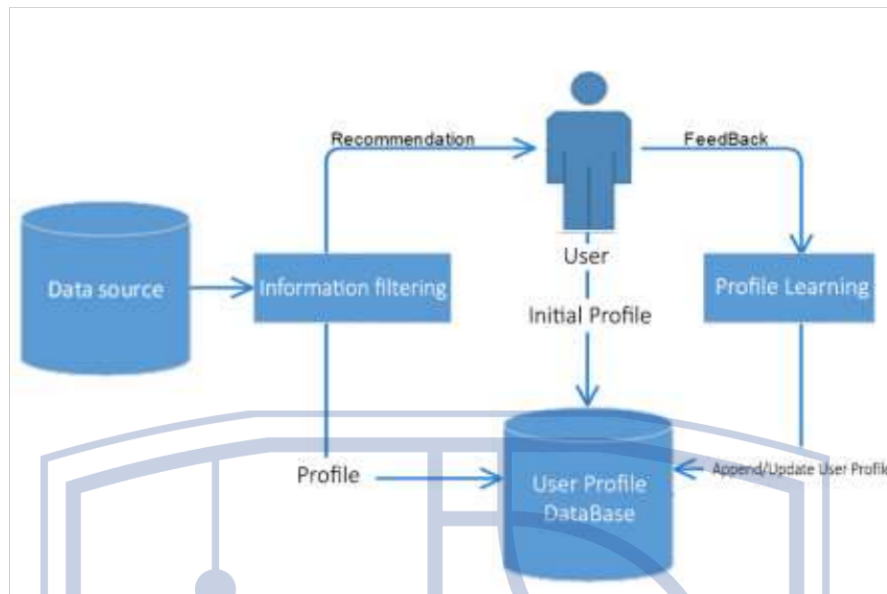
2.6 Payment Gateway

Payment gateway adalah sistem pembayaran yang memungkinkan pembeli dan penjual melakukan transaksi secara *online* melalui berbagai metode pembayaran pada aplikasi *website* dan *mobile* [36]. *Payment gateway* telah banyak digunakan dalam dunia bisnis karena memudahkan pelanggan untuk menyelesaikan pembelian dengan mudah dan cepat. *Payment gateway* tidak hanya berfokus pada kenyamanan pelanggan dalam memproses transaksi, tetapi juga memastikan keamanan data dan integritas transaksi antara pihak-pihak yang terlibat, dengan mengenkripsi informasi sensitif seperti nomor kartu dan kode keamanan untuk mengurangi risiko pencurian data. Di Indonesia, terdapat berbagai penyedia *payment gateway* yang memungkinkan pemilik aplikasi untuk mengintegrasikan layanan ini melalui *API* yang disediakan oleh penyedia tersebut, salah satu penyedia yang banyak digunakan adalah *Midtrans*.

Midtrans adalah salah satu *payment gateway* yang menyediakan fasilitas berbagai cara pembayaran [37]. *Midtrans* menawarkan berbagai metode pembayaran mulai dari kartu kredit, kartu debit, *e-wallet* seperti GoPay, OVO, dan Dana hingga *transfer bank* dan gerai *retail* seperti Alfamart dan Indomaret. *Midtrans* dapat diintegrasikan ke dalam aplikasi bisnis, baik itu *website* maupun aplikasi *mobile* melalui *Application Programming Interface* (*API*) yang telah disediakan. Proses integrasi ini memungkinkan aplikasi untuk berkomunikasi dengan sistem *Midtrans* guna memproses pembayaran dari pengguna.

2.7 Sistem Rekomendasi

Sistem rekomendasi adalah suatu sistem atau aplikasi yang dirancang untuk dapat memberikan rekomendasi mengenai suatu hal untuk mengambil keputusan yang diinginkan oleh pengguna sistem [38]. Sistem rekomendasi ini dapat membantu para pengguna yang kebingungan dalam memilih produk atau konten dengan memberikan rekomendasi yang dihasilkan dari pengolahan data pengguna itu sendiri, sehingga memiliki kemungkinan yang tinggi untuk mendapatkan rekomendasi yang sesuai dengan preferensi pengguna. Tujuan dari penerapan sistem rekomendasi adalah untuk meningkatkan penjualan, menjual lebih banyak barang yang beragam, meningkatkan kepuasan pengguna, meningkatkan kesetiaan pengguna, dan lebih memahami keinginan pengguna [39]. Terdapat beberapa metode yang dapat digunakan untuk sistem rekomendasi yaitu:



Gambar 2.3 Arsitektur Sistem Rekomendasi [40]

1. *KNN*: Berfungsi untuk menemukan kelompok *item* atau pengguna yang paling mirip dengan suatu *target* dengan mengidentifikasi “tetangga” atau K pengguna/*item* yang paling dekat berdasarkan metrik kesamaan.
2. *Collaborative Filtering*: Merekomendasikan *item* berdasarkan kesamaan preferensi antar pengguna dengan mengidentifikasi pola selera yang mirip, sehingga *item* yang disukai oleh sekelompok pengguna dapat direkomendasikan kepada pengguna lain dalam kelompok tersebut.
3. *Content-Based Filtering*: Memilih *item* dengan konten yang mirip berdasarkan apa yang dicari oleh pengguna, tanpa memperhatikan penilaian dari pengguna lain.
4. *Popularity-Based*: Memberikan rekomendasi *item* berdasarkan popularitas atau tingkat penggunaan oleh seluruh pengguna, terlepas dari preferensi individu.

2.8 Content-Based Filtering

Content-Based Filtering pada sistem rekomendasi adalah metode yang mempertimbangkan perilaku dari pengguna di masa lalu yang kemudian diidentifikasi pola perilakunya untuk merekomendasikan barang yang sesuai dengan pola perilaku tersebut [41]. Metode *content-based filtering* menganalisis preferensi perilaku pengguna sebelumnya untuk membuat pola. Algoritma tersebut akan disesuaikan dengan sekumpulan atribut karakteristik produk yang akan direkomendasikan. *Item* dengan persentase kecocokan tertinggi akan menjadi rekomendasi bagi pengguna. Tahapan dari *Content-Based Filtering* adalah [42]:

1. Pengguna melakukan pencarian dari suatu *item* dan memberikan like [43]
2. Profil pengguna dibentuk berdasarkan fitur dari *item-item* yang telah di-like. Pembuatan profil pengguna dapat menggunakan algoritma TF-IDF (*term frequency-inverse document frequency*) [44]. TF merupakan jumlah *term* dalam sebuah dokumen dan dapat dihitung dengan persamaan:

$$idf_i = \log \left(\frac{n}{df_i} \right) \dots \dots \dots (1)$$

$$W_{td} = tf \times IDF$$

$$W_{td} = tf \times \log \frac{N}{df}$$

Dimana: d : dokumen ke-d

t : kata ke-t dari kata kunci

W : bobot dokumen ke-d terhadap kata ke-t

tf : banyak kata yang dicari pada sebuah dokumen

IDF : *Inversed Document Frequency*

N : jumlah total dokumen

3. Setelah profil pengguna terbentuk, sistem mulai mencari *item-item* lain yang mirip dengan profil pengguna. Pengukuran kesamaan antara profil pengguna dan *item* baru dilakukan dengan menggunakan metode yang umum digunakan untuk menghitung kesamaan adalah *cosine similarity*, yang menghitung vektor yang terkait antara nilai 0 dan 1. Nilai 0 mewakili *user profile* dan *item* sangat tidak relevan sedangkan nilai 1 adalah sebaliknya. Untuk mencari *cosine similarity* antar dokumen dapat dinyatakan pada persamaan berikut:

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^n A_i \cdot B_i}{\sqrt{(\sum_{i=1}^n A_i)^2} \times \sqrt{(\sum_{i=1}^n B_i)^2}} \dots \dots \dots (2)$$

Di mana :

A = Vektor A, yang akan dibandingkan kemiripannya

B = Vektor B, yang akan dibandingkan kemiripannya

A.B = dot product antara vektor A dan vektor B

|A| = panjang vektor A

|B| = panjang vektor B

2.9 Evaluasi Model

Evaluasi model adalah langkah krusial dalam proses pengembangan *ML* (*Machine Learning*). Ini memberi kita gambaran tentang seberapa baik model bekerja pada data yang tidak pernah dilihatnya sebelumnya, yang penting untuk memastikan bahwa model dapat menggeneralisasi dengan baik. Ada berbagai metrik evaluasi yang digunakan tergantung pada jenis masalah. Misalnya, dalam tugas klasifikasi, akurasi, presisi, *recall*, dan *F1-Score* adalah metrik umum. Untuk tugas regresi, *Mean Squared Error* (*MSE*) dan *Root Mean Squared Error* (*RMSE*) sering digunakan [45].

Masalah umum yang dihadapi dalam *ML* adalah *overfitting* dan *underfitting*. *Overfitting* terjadi ketika model belajar pola dalam data pelatihan dengan sangat detail hingga kehilangan kemampuan untuk menggeneralisasi pada data baru. *Underfitting* adalah kebalikannya, di mana model tidak belajar cukup dari data pelatihan dan karenanya berkinerja buruk bahkan pada data pelatihan. Tuning model dan menggunakan teknik seperti validasi silang dapat membantu mengatasi masalah ini [45].

Confusion Matrix, juga dikenal sebagai tabel kontingensi, adalah sebuah metode yang digunakan untuk mengevaluasi performa model klasifikasi. Ini adalah sebuah tabel yang memperlihatkan jumlah prediksi yang benar dan salah yang dilakukan oleh model pada set data pengujian yang diketahui label kelasnya. Dengan menggunakan *Confusion Matrix*, kita dapat mendapatkan wawasan yang lebih rinci tentang performa model klasifikasi, termasuk kekuatan dan kelemahan model dalam mengklasifikasikan instans pada set data pengujian [46].

Confusion Matrix adalah sebuah metode evaluasi kinerja model *machine learning* yang digunakan untuk menghitung jumlah prediksi yang benar dan yang salah. *Confusion matrix* terdiri dari empat elemen yaitu *True Positive* (*TP*), *True Negative* (*TN*), *False Positive* (*FP*), dan *False Negative* (*FN*). *True Positive* adalah jumlah data yang benar diprediksi sebagai positif, *True Negative* adalah jumlah data yang benar diprediksi sebagai negatif, *False Positive* adalah jumlah data yang salah diprediksi sebagai positif, dan *False Negative* adalah jumlah data yang salah diprediksi sebagai negatif. *Confusion matrix* dapat digunakan untuk menghitung berbagai metrik evaluasi model seperti akurasi, presisi, *recall*, dan *F1-score*. Akurasi menghitung persentase prediksi yang benar dari seluruh data, presisi menghitung persentase data positif yang benar [46].

Untuk mengukur kemampuan model memprediksi yang benar untuk data positif dari total prediksi positif yang tersedia, digunakan metrik evaluasi Presisi atau *Precision* dengan model sebagai berikut [47]:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Untuk mengukur kemampuan model dalam mengidentifikasi semua data positif dengan benar, digunakan metrik evaluasi sensitivitas (*Recall*) dengan model sebagai berikut [47]:

$$Recall = \frac{TP}{TP+FN}$$

Untuk mengukur keseimbangan atau kombinasi antara Presisi (*Precision*) dan Sensitivitas (*Recall*), digunakan metrik evaluasi *F1-score* dengan model sebagai berikut [47]:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

2.10 Normalisasi Data

Teknik yang umum digunakan mencakup penggunaan rumus *Simple Additive Weighting* (SAW) dan metode *Min-Max Normalization*. Teknik SAW mengonversi data menjadi skala yang dapat diperbandingkan, mempermudah proses pengambilan keputusan. Sementara itu, *Min-Max Normalization* menyesuaikan nilai data dalam rentang tertentu, biasanya antara 0 dan 1, untuk menghilangkan perbedaan skala dan memastikan konsistensi data dalam analisis lebih lanjut.

2.10.1 Simple Additive Weighting (SAW)

Metode SAW sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari *rating* kinerja pada setiap *alternative* pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua *rating alternative* yang ada [48]. Metode *Simple Additive Weighting* (SAW) mengenal adanya 2 (dua) atribut yaitu kriteria keuntungan (*benefit*) dan kriteria biaya (*cost*). Perbedaan mendasar dari kedua kriteria ini adalah dalam pemilihan kriteria ketika mengambil keputusan [49].

Metode SAW ini mengharuskan pembuat keputusan menentukan bobot bagi setiap atribut. Skor total untuk alternatif diperoleh dengan menjumlahkan seluruh hasil perkalian antara *rating* (yang dapat dibandingkan lintas atribut) dan bobot tiap atribut. *Rating* tiap atribut haruslah bebas dimensi dalam arti telah melewati proses normalisasi matriks sebelumnya [48].

$$r_{ij} = \begin{cases} \frac{X_{ij}}{\max X_{ij}} & \text{jika } j \text{ atribut keuntungan (benefit)} \\ \frac{X_{ij}}{\min X_{ij}} & \text{jika } j \text{ atribut biaya (Cost)} \end{cases} \quad (4)$$

Keterangan:

r_{ij} = Nilai rating kinerja ternormalisasi

x_{ij} = Nilai atribut yang dimiliki dari setiap kriteria

X_{ij} = Nilai terbesar dari setiap kriteria

$\min X_{ij}$ = Nilai terkecil dari setiap kriteria

Benefit = Jika nilai terbesar adalah terbaik Cost

Cost = Jika nilai terkecil adalah terbaik

Dimana r_{ij} adalah rating kinerja ternormalisasi dari *alternative* A_i pada atribut C_j ; $i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$. Nilai preferensi untuk setiap *alternative* (V_i) diberikan sebagai:

$$V_i = \sum_{j=1}^n w_j r_{ij} \quad (3)$$

Keterangan :

V_i = ranking untuk setiap *alternative*

W_j = nilai bobot dari setiap kriteria

r_{ij} = nilai rating kinerja ternormalisasi nilai-nilai V_i yang lebih besar mengindikasikan bahwa *alternative* A_i lebih terpilih.

Kelebihan dari model *Simple Additive Weighting* (SAW) dibandingkan dengan model pengambilan keputusan yang lain terletak pada kemampuannya untuk melakukan penilaian secara lebih tepat karena didasarkan pada nilai kriteria dan bobot preferensi yang sudah ditentukan, selain itu SAW juga dapat menyeleksi *alternative* terbaik dari sejumlah *alternative* yang ada karena adanya proses perangkingan setelah menentukan nilai bobot untuk setiap atribut [48].

2.10.2 Min Max Normalization

Penghitungan untuk prediksi *rating* dengan menggunakan *dot product* pada hasil kemiripan *item* tersebut. Hasil prediksi kemudian dilakukan normalisasi dengan nilai antara *rating* terendah dan tertinggi pada *film* dengan menggunakan *MinMaxScaler*. *MinMaxScaler* merupakan metode normalisasi yang membuat data ada pada rentang nilai minimum dan maksimum pada *dataset*. Setiap nilai pada hasil prediksi dikurangi dengan nilai minimum *rating* pada *dataset rating*, kemudian dibagi dengan rentang nilai atau nilai maksimum

dikurangi nilai minimum dari *dataset rating*. Rumus yang digunakan perhitungan *MinMaxScaler* sesuai dengan persamaan (5) [16].

$$x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}} \dots\dots\dots (5)$$

x_{new} = hasil nilai normalisasi *MinMaxScaler*

x_{old} = nilai awal

x_{max} = nilai maksimum

x_{min} = nilai minimum

Lalu pada data *training* diurutkan berdasarkan hasil prediksi dengan nilai tertinggi pada tiap pengguna sebagai *film* yang direkomendasikan. Sedangkan pada data *testing* dihitung nilai akurasi pada hasil prediksi yang menggunakan algoritma *cosine similarity* dan *euclidean distance*. Perhitungan akurasi menggunakan nilai *mean absolute error* serta nilai *root mean square error*.

2.11 Basis Data

Kumpulan data yang terorganisir dan disimpan secara elektronik disebut basis data. Sistem ini dibuat untuk mengatur, menyimpan, dan mengambil informasi dengan efisien. Basis data biasanya terdiri dari beberapa tabel yang saling terhubung, dimana setiap tabel berisi catatan atau entri dengan informasi khusus. Penggunaan basis data mempermudah pengguna untuk memasukkan, memperbarui, menghapus, dan mengambil data. Ini sangat penting dalam berbagai aplikasi, seperti sistem manajemen inventaris dan *platform* media sosial. Basis data membantu menyusun data yang rumit, memungkinkan beberapa orang mengakses informasi bersamaan, serta melindungi keaslian dan keamanan data. Dengan kemampuan untuk menangani volume data yang besar dan melakukan operasi pencarian yang cepat, basis data menjadi komponen kunci dalam sistem informasi *modern* [50].

Jenis-jenis Basis Data :

- Basis Data relasional : Menyusun data ke dalam tabel dengan baris dan kolom untuk menyimpan dan mengambil informasi dengan mudah. Contoh: *MySQL*, *PostgreSQL*, *Oracle Database*, *Microsoft SQL Server*, dan *SQLite*.
- Basis Data berorientasi objek : Basis Data berorientasi objek adalah jenis database yang menyimpan informasi dalam bentuk objek, seperti data dan metode untuk menyimpan data tersebut. Contoh: Basis data ini cocok untuk merepresentasikan objek dunia nyata dan relasinya.

- Basis Data NoSQL : Basis Data NoSQL tidak mengikuti model relasional tradisional. Data disimpan dalam bentuk tidak terstruktur, seperti dokumen, grafik, atau nilai kunci. Contoh: *MongoDB*, *Apache Cassandra*, *Apache HBase*, *CouchDB*.
- Basis Data Graf : basis data yang menggunakan struktur grafik untuk menyimpan dan mengelola data. Contoh : Sistem rekomendasi film.

2.12 Perancangan Sistem

2.12.1 Use Case Diagram

Use Case Diagram merupakan suatu bentuk diagram yang menggambarkan fungsi-fungsi yang diharapkan dari sebuah sistem yang dikembangkan [51]. Diagram ini digunakan untuk menjelaskan bagaimana pengguna berinteraksi dengan sistem melalui serangkaian "use case" atau skenario yang mendeskripsikan setiap fungsi utama yang ditawarkan oleh sistem. Dengan menggambarkan kebutuhan fungsional dari perspektif pengguna, *use case diagram* menjadi alat yang efektif untuk menggambarkan proses-proses apa saja yang dapat dilakukan oleh sistem, siapa saja yang terlibat, dan hubungan antara pengguna (aktor) dan fungsi-fungsi yang ada.

Pada *use case diagram*, terdapat sejumlah simbol yang digunakan untuk menggambarkan elemen-elemen tersebut. Simbol-simbol ini memiliki makna tertentu dan penting untuk dipahami agar interpretasi diagram menjadi lebih jelas dan akurat. Berikut merupakan simbol-simbol pada *use case diagram*:

Tabel 2.7 Simbol *Use Case Diagram* [51] [52]

NO.	Simbol	Nama	Keterangan
1.		<i>Actor</i>	Menspesifikasi himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri (<i>independent</i>).

3.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada diatas objek induk (<i>ancestor</i>).
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> adalah sumber secara eksplisit.
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case target</i> memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6.		<i>Associaton</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

2.12.2 Activity Diagram

Activity Diagram adalah *diagram* yang secara visual menggambarkan aliran kerja atau rangkaian aktivitas dalam sebuah sistem atau proses bisnis, mulai dari aktivitas pertama hingga hasil akhir [52]. Diagram ini digunakan untuk menunjukkan urutan dan logika dari setiap langkah yang dilakukan dalam suatu proses, baik itu proses sistem maupun proses bisnis yang melibatkan banyak pihak. Dalam diagram ini, kita bisa melihat bagaimana setiap aktivitas dimulai, bagaimana satu langkah mengalir ke langkah berikutnya, serta bagaimana keputusan atau kondisi tertentu dapat mempengaruhi alur yang terjadi. *Activity diagram* juga memungkinkan kita memahami jalur alternatif, proses paralel, serta perulangan yang mungkin terjadi dalam suatu proses, memberikan gambaran yang lebih lengkap dan terperinci.

Untuk menggambarkan aliran aktivitas ini, *activity diagram* menggunakan sejumlah simbol khusus. Setiap simbol memiliki peran tertentu dalam merepresentasikan elemen-

elemen aktivitas, sehingga pemahaman mengenai simbol-simbol ini penting agar diagram dapat dibaca dan dipahami dengan benar. Berikut merupakan simbol-simbol pada *activity diagram* :

Tabel 2.8 Simbol-Simbol *Activity Diagram* [52]

No.	Simbol	Nama	Keterangan
1.		<i>Initial</i>	Titik awal, untuk memulai suatu aktivitas.
2.		<i>Dependency</i>	Titik akhir, untuk mengakhiri aktivitas.
3.		<i>Activity</i>	Menandakan sebuah aktivitas.
4.		<i>Decision</i>	Pilihan untuk mengambil keputusan.
5.		<i>Fork / Join</i>	Menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.

2.12.3 Entity Relational Diagram

Entity Relationship Diagram (ERD) adalah suatu representasi visual yang menggambarkan struktur data dan hubungan antar entitas di dalam suatu sistem basis data [53]. ERD sangat bermanfaat dalam tahap awal pengembangan perangkat lunak karena membantu para desainer basis data, pengembang perangkat lunak, atau pemangku kepentingan lainnya dalam memahami bagaimana data saling terkait dan terstruktur. ERD juga digunakan untuk memastikan bahwa semua kebutuhan data dalam sistem dapat teridentifikasi dengan baik dan tidak ada informasi penting yang terlewatkan.

ERD memiliki beberapa simbol yang digunakan untuk merepresentasikan berbagai komponen utama dalam model data, yaitu [54] [55]:

1. Entitas: menggunakan simbol persegi panjang untuk merepresentasikan objek yang dapat diidentifikasi secara unik dalam sistem.
2. Atribut: menggunakan simbol *oval* untuk merepresentasikan karakteristik atau informasi penting.
3. Hubungan: menggunakan simbol belah ketupat untuk menggambarkan koneksi antara dua atau lebih entitas untuk melihat bagaimana entitas berinteraksi satu sama lain di dalam sistem. Terdapat beberapa jenis kardinalitas untuk merepresentasikan hubungan antar entitas yaitu:

Tabel 2.9 Kardinalitas Hubungan ERD [56]

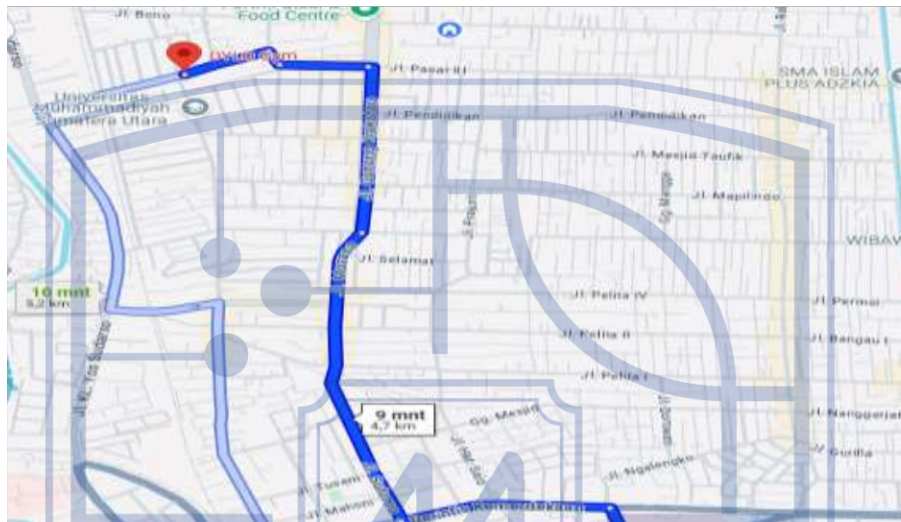
NO.	Notasi	Simbol	Keterangan
1.		<i>one-one</i>	Hubungan di mana satu entitas berhubungan dengan satu entitas lainnya
2.		<i>one-to-many</i>	Hubungan di mana satu entitas dapat berhubungan dengan banyak entitas lain. Contoh: Dosen - Mahasiswa.
3.		<i>many-to-one</i>	Hubungan di mana banyak entitas berhubungan dengan satu entitas.
4.		<i>many-to-many</i>	Hubungan di mana banyak entitas berhubungan dengan banyak entitas lainnya

2.13 Semesta GYM

Semesta GYM merupakan pusat kebugaran jasmani yang bergerak dalam bidang jasa, yaitu tempat untuk melakukan aktivitas fisik dan olahraga. Semesta GYM beroperasi pada hari Senin sampai dengan hari Sabtu di 07.00 WIB sampai jam 22.00 WIB dan hari Minggu Jam 09.00 WIB sampai jam 22.00 WIB, dan jumlah anggota *gym* pada jangka waktu tersebut mencapai 25-30 orang per hari. Anggota *gym* yang datang di Semesta GYM dapat

melakukan aktivitas *gym* sepuasnya dari berbagai fasilitas yang ada di Semesta GYM berupa alat beban, mesin olahraga dan loker.

Dengan dedikasi yang berkelanjutan, Semesta GYM juga memiliki kesan yang bagus karena Semesta GYM memiliki anggota *gym* yang berupa binaragawan dan TNI yang latihan di pusat kebugaran yang disediakan oleh Semesta GYM. Semesta GYM berlokasi di jalan Alfalah No.27, Glugur Darat I, Kec. Medan Tim., Kota Medan, Sumatera Utara 20238.



Gambar 2.4 Lokasi Semesta GYM

Semesta GYM dikelola langsung oleh pemiliknya dan beberapa saudara yang ikut membantu pemilik Semesta GYM dalam pengelolaan GYM, keamanan di Semesta GYM, mengecek proses pencatatan anggota *gym* serta menciptakan lingkungan yang ramah dan penuh kepercayaan bagi anggota *gym* yang datang. Anggota *gym* yang mengunjungi Semesta GYM dari sumber yang beragam seperti rekomendasi dari beberapa binaragawan dan ada juga dari mahasiswa mahasiswi yang ingin melatih otot dan postur badan yang sempurna.



Gambar 2.5 Semesta GYM

Dari Gambar 2.5 merupakan pemandangan di depan Semesta GYM dan menjelaskan bahwa Semesta GYM ialah pusat kebugaran yang bergerak dibidang jasa dan menyediakan lapangan latihan kepada orang-orang umum dan tidak ada larangan dalam menggunakan alat dan larangan dalam berpakaian.