

BAB II

KAJIAN LITERATUR

2.1. Sistem Informasi

Secara konseptual, sistem informasi didefinisikan sebagai sekumpulan komponen yang saling berinteraksi untuk mengumpulkan, memproses, menyimpan, serta mendistribusikan informasi guna mendukung proses pengambilan keputusan, koordinasi, dan pengendalian di dalam suatu organisasi [10]. Komponen pembentuknya mencakup perangkat keras, perangkat lunak, manusia, prosedur kerja, serta basis data yang bekerja secara selaras, mulai dari komponen masukan untuk menghimpun data mentah, komponen pengolahan untuk mentransformasikan data menjadi informasi bermakna, hingga komponen keluaran dan penyimpanan yang menjaga ketersediaan data untuk kebutuhan lanjutan [10]. Melalui pemanfaatan sistem informasi, berbagai proses administratif yang mulanya dikerjakan secara manual dapat diotomatisasi, sehingga menghemat tenaga, biaya, dan waktu secara signifikan bagi organisasi, termasuk pusat kebugaran [10]. Dalam penelitian ini, sistem informasi bertindak sebagai fondasi utama yang menopang seluruh proses pengelolaan data keanggotaan, rekapitulasi kehadiran, serta penerapan mekanisme gamifikasi, yang diwujudkan dalam bentuk aplikasi bergerak berbasis *mobile* sehingga pengelola dan anggota pusat kebugaran dapat mengakses seluruh data secara digital dan *real-time* [10].

2.1.1. Konsep Sistem Informasi

Konsep sistem informasi didasarkan pada gagasan bahwa data yang dikumpulkan dari berbagai sumber perlu diolah menjadi informasi yang berguna bagi penggunanya. Secara konseptual, sistem informasi dapat dipahami sebagai perpaduan antara teknologi, manusia, dan proses yang dirancang untuk menghasilkan informasi yang akurat, tepat waktu, dan relevan sesuai kebutuhan organisasi. Konsep ini menegaskan bahwa sistem informasi bukan semata-mata tentang teknologi komputer, melainkan juga tentang bagaimana teknologi tersebut dikelola dan dimanfaatkan oleh manusia dalam konteks organisasional tertentu [11]. Oleh karena itu, keberhasilan sebuah sistem informasi sangat bergantung pada kesesuaian antara solusi teknologi yang diterapkan dengan kebutuhan nyata penggunanya.

Dalam konteks pengembangan aplikasi *mobile* ini, konsep sistem informasi diwujudkan melalui integrasi berbagai komponen yang bekerja secara terkoordinasi untuk menghasilkan layanan yang berguna bagi pengguna. Komponen input berupa data latihan, kehadiran, dan keanggotaan yang dimasukkan pengguna ke dalam aplikasi diolah oleh lapisan logika bisnis melalui REST API, kemudian disimpan dalam basis data MySQL, dan disajikan kembali sebagai informasi yang bermakna kepada anggota maupun pengelola gym. Konsep sistem informasi dalam

penelitian ini juga mencakup aspek keamanan, di mana sistem menerapkan mekanisme autentikasi berbasis token untuk memastikan hanya pengguna yang berwenang yang dapat mengakses informasi dalam sistem.

2.1.2. Pengembangan Sistem Informasi

Pengembangan sistem informasi mencakup serangkaian proses terstruktur yang bertujuan untuk merancang, membangun, dan mengimplementasikan sistem berbasis teknologi guna memenuhi kebutuhan informasi pengguna secara efektif dan efisien. Proses pengembangan sistem informasi tidak dapat dilakukan secara sembarangan, melainkan harus mengikuti pendekatan sistematis yang memperhatikan kebutuhan pengguna, arsitektur sistem, kualitas perangkat lunak, serta keberlanjutan sistem setelah diimplementasikan [12]. Pemahaman terhadap konsep-konsep inti pengembangan sistem informasi menjadi landasan penting dalam penelitian ini karena aplikasi MissionFit dibangun melalui serangkaian proses pengembangan yang melibatkan perangkat lunak, rekayasa sistem, basis data, dan antarmuka pemrograman yang saling terintegrasi.

Terdapat beberapa konsep inti yang menjadi fondasi pengembangan sistem informasi dalam penelitian ini, yaitu sebagai berikut:

1. Perangkat Lunak (*Software*)

Perangkat lunak adalah sekumpulan instruksi atau program yang ditulis dalam bahasa pemrograman tertentu dan dieksekusi oleh perangkat keras untuk menjalankan tugas-tugas yang spesifik [10]. Perangkat lunak merupakan komponen kritis dalam sistem informasi karena menentukan fungsionalitas, kinerja, dan pengalaman pengguna secara keseluruhan. Dalam penelitian ini, perangkat lunak yang dikembangkan adalah aplikasi mobile berbasis Android yang menggunakan bahasa pemrograman Dart melalui framework Flutter untuk sisi antarmuka, serta REST API sebagai penghubung antara aplikasi dan basis data.

2. Rekayasa Perangkat Lunak (*Software Engineering*)

Rekayasa perangkat lunak adalah penerapan pendekatan sistematis, disiplin, dan terukur dalam proses pengembangan, pengoperasian, dan pemeliharaan perangkat lunak [10]. Bidang ini mencakup berbagai metodologi, alat bantu, dan praktik terbaik yang dirancang untuk menghasilkan perangkat lunak berkualitas tinggi secara efisien dan terstruktur. Dalam penelitian ini, prinsip rekayasa perangkat lunak diterapkan melalui penggunaan Waterfall Model sebagai metodologi pengembangan, UML sebagai alat pemodelan sistem, dan Black Box Testing sebagai metode validasi fungsionalitas.

3. Sistem Informasi Berbasis Mobile

Sistem informasi berbasis mobile adalah sistem yang dirancang dan diimplementasikan khusus untuk berjalan pada perangkat bergerak seperti smartphone dan tablet. Keunggulan utama sistem ini terletak pada kemampuannya untuk diakses kapan saja dan di mana saja tanpa dibatasi oleh waktu maupun lokasi fisik. Dalam penelitian ini, sistem informasi berbasis mobile diwujudkan dalam bentuk aplikasi mobile yang dikembangkan untuk platform Android menggunakan framework Flutter, sehingga anggota gym dapat mencatat aktivitas latihan, memantau perkembangan, dan mengakses fitur gamifikasi secara langsung dari perangkat smartphone [8]. Salah satu aspek penting dalam pengembangan sistem informasi berbasis mobile adalah proses pengumpulan data pengguna pada tahap awal penggunaan aplikasi. Pada aplikasi MissionFit, proses ini diwujudkan melalui halaman pengaturan profil awal yang wajib diselesaikan oleh anggota segera setelah proses pendaftaran akun berhasil dilakukan. Halaman ini dirancang dalam tiga langkah berurutan, yaitu pengisian data fisik dasar yang meliputi jenis kelamin, tanggal lahir, tinggi badan, dan berat badan, kemudian pemilihan tujuan latihan utama, dan terakhir penentuan tingkat aktivitas harian. Data yang dikumpulkan melalui tahap ini disimpan ke dalam basis data dan dimanfaatkan oleh sistem sebagai dasar personalisasi pengalaman latihan anggota sejak pertama kali menggunakan aplikasi.

4. Basis Data (Database)

Basis data adalah kumpulan data yang terorganisir secara terstruktur dan disimpan secara elektronik dalam sistem komputer, dirancang untuk memudahkan penyimpanan, pengelolaan, dan pengambilan kembali data secara efisien [15]. Dalam pengembangan sistem informasi, basis data berperan sebagai pusat penyimpanan yang menjadi sumber data tunggal bagi seluruh komponen sistem. Penelitian ini menggunakan MySQL sebagai sistem manajemen basis data relasional yang berjalan di atas server lokal XAMPP, menyimpan seluruh data sistem mulai dari data pengguna, sesi latihan, absensi, hingga data gamifikasi.

5. Antarmuka Pemrograman Aplikasi (*Application Programming Interface*)

Application Programming Interface (API) adalah sekumpulan aturan, protokol, dan alat yang memungkinkan komunikasi serta pertukaran data antara dua perangkat lunak yang berbeda [16]. REST API merupakan gaya arsitektur API yang memanfaatkan protokol HTTP dan format data JSON untuk memfasilitasi komunikasi yang ringan, cepat, dan mudah diintegrasikan. Dalam penelitian ini, REST API berfungsi sebagai jembatan komunikasi antara aplikasi Flutter di sisi

mobile dengan basis data MySQL di sisi server, sehingga seluruh operasi data dapat dilakukan secara aman dan terstruktur.

2.2. Fitness (Pusat Kebugaran)

Pusat kebugaran atau gym merupakan lingkungan utama yang menjadi konteks pengembangan aplikasi dalam penelitian ini. Pemahaman mengenai konsep fitness, jenis latihan, dan kebutuhan anggota gym secara langsung mempengaruhi perancangan program latihan yang disediakan dalam aplikasi. Program latihan yang dirancang berdasarkan pemahaman ilmiah tentang kelompok otot dan prinsip pemulihan akan menghasilkan sesi latihan yang lebih terarah, terukur, dan aman bagi anggota gym.

Fitness atau kebugaran jasmani adalah kondisi fisik tubuh yang mencerminkan kemampuan seseorang untuk menjalankan aktivitas sehari-hari dengan penuh energi tanpa mengalami kelelahan berlebihan, sekaligus memiliki cadangan daya tahan yang cukup untuk menghadapi situasi tak terduga. Kondisi ini tidak terbatas pada aspek fisik semata, melainkan juga mencakup dimensi mental dan sosial yang membentuk kualitas hidup seseorang secara menyeluruh [1]. Pusat kebugaran atau gym adalah fasilitas yang menyediakan peralatan, ruang, dan program latihan terstruktur untuk mendukung pencapaian target kebugaran pengguna secara optimal. Seiring meningkatnya kesadaran masyarakat akan pentingnya gaya hidup aktif, gym telah bertransformasi dari sekadar tempat berolahraga menjadi ekosistem kebugaran yang komprehensif. Transformasi ini mendorong kebutuhan akan sistem pengelolaan yang lebih terdigitalisasi, karena pengelolaan konvensional dinilai tidak lagi memadai untuk mengakomodasi volume data keanggotaan, jadwal latihan, dan catatan kehadiran yang terus bertambah [3]. Di sinilah peran aplikasi fitness berbasis mobile menjadi sangat relevan, yaitu sebagai jembatan antara kebutuhan operasional pengelola gym dan kebutuhan pemantauan progres dari anggota gym.

Salah satu tantangan terbesar dalam ekosistem gym adalah mempertahankan motivasi dan konsistensi latihan anggota dalam jangka panjang. Anggota yang tidak memiliki sistem pencatatan progres yang terstruktur cenderung kehilangan motivasi lebih cepat karena tidak dapat melihat dan mengukur kemajuan yang telah dicapainya secara nyata [2]. Kondisi ini menjadi salah satu justifikasi utama dikembangkannya aplikasi mobile ini, yang tidak hanya berfungsi sebagai sistem manajemen gym tetapi juga sebagai alat pemantauan progres dan motivasi berbasis gamifikasi bagi setiap anggota, salah satunya melalui program latihan yang tersedia dan terdiri dari tiga sesi utama yang dirancang berdasarkan prinsip split training, yaitu metode pembagian sesi latihan berdasarkan kelompok otot yang dilatih agar setiap kelompok otot mendapat waktu pemulihan yang cukup sebelum dilatih kembali. Ketiga program tersebut adalah sebagai berikut.

1. Latihan Dorong

Latihan Dorong adalah sesi latihan yang berfokus pada kelompok otot yang bekerja saat melakukan gerakan mendorong, yaitu otot dada (*pectoral*), otot bahu (*deltoid*), dan otot lengan belakang (*triceps*). Latihan dalam sesi ini melibatkan beban untuk memberikan resistansi pada gerakan mendorong sehingga otot-otot tersebut berkembang dan menguat. Program ini mencakup gerakan-gerakan utama seperti *Bench Press*, *Incline Dumbbell Press*, *Shoulder Press*, *Lateral Raise*, *Cable Tricep Pushdown*, dan *Chest Fly*.

2. Latihan Tarik

Latihan Tarik adalah sesi latihan yang menargetkan kelompok otot yang bekerja saat melakukan gerakan menarik, yaitu otot punggung (*latissimus dorsi*, *rhomboid*, *trapezius*) dan otot lengan depan (*biceps*). Gerakan dalam sesi ini dirancang untuk membangun kekuatan tarikan, memperbaiki postur tubuh, serta membentuk lebar punggung yang merupakan salah satu indikator visual utama kebugaran seseorang. Program ini terdiri dari tujuh gerakan utama, yaitu *Pull Up*, *Barbell Row*, *Lat Pulldown*, *Seated Cable Row*, *Face Pull*, *Dumbbell Bicep Curl*, dan *Hammer Curl*.

3. Latihan Kaki dan Otot Perut

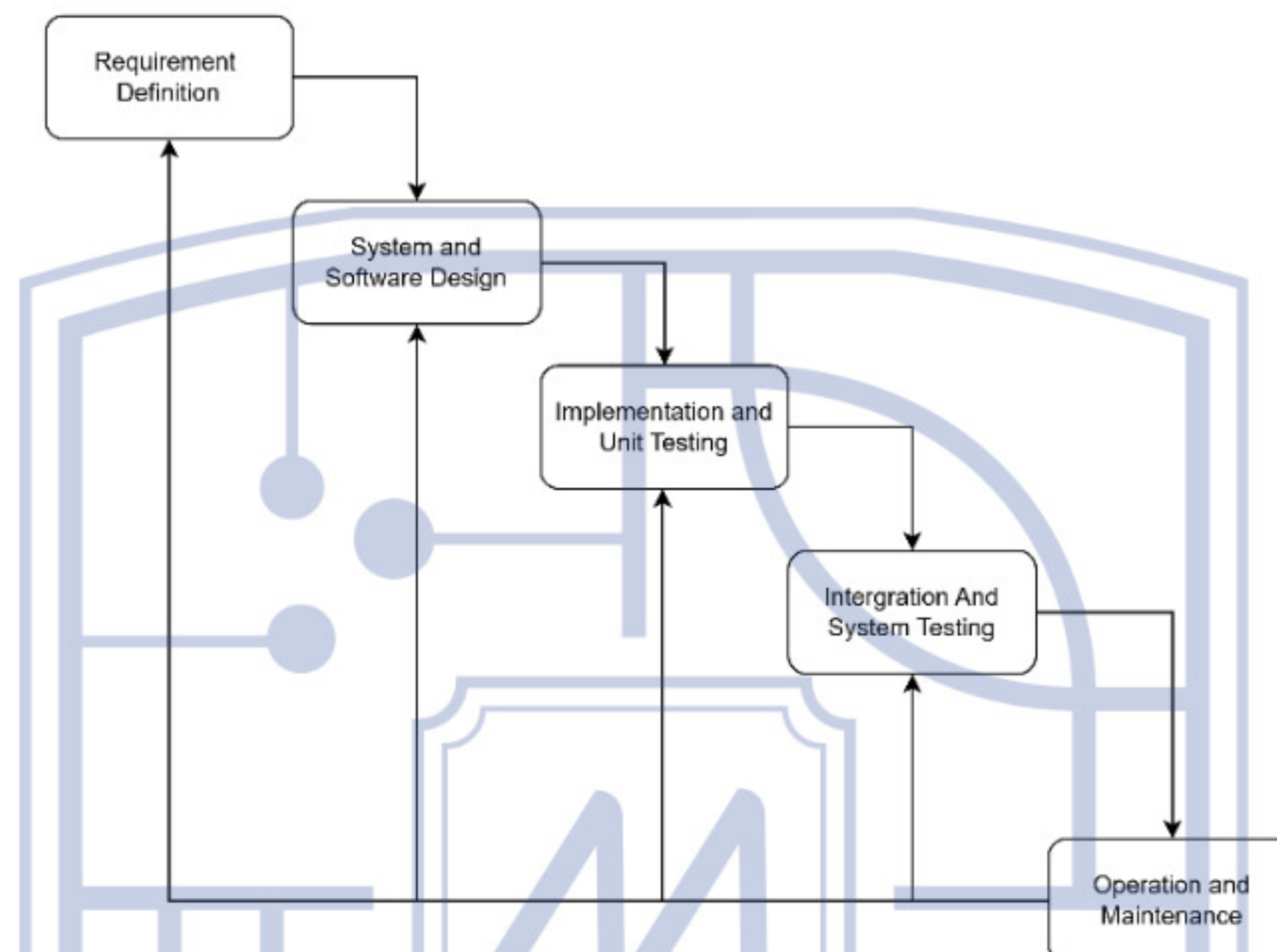
Latihan Kaki dan Otot Perut adalah sesi latihan yang ditujukan untuk melatih kelompok otot kaki (*quadriceps*, *hamstring*, *gluteus*, dan *calf*) sekaligus otot inti perut (*core*). Latihan kaki merupakan komponen yang sangat penting dalam program kebugaran menyeluruh karena kelompok otot kaki adalah kelompok otot terbesar dalam tubuh manusia, sehingga melatihnya secara rutin tidak hanya membentuk kekuatan kaki tetapi juga meningkatkan metabolisme tubuh secara signifikan. Program ini mencakup gerakan *Squat*, *Leg Press*, *Romanian Deadlift*, *Leg Curl*, *Calf Raise*, *Plank*, dan *Crunches*.

2.3. Metode Pengembangan Perangkat Lunak

Dalam proses pengembangan perangkat lunak, pemilihan metodologi yang tepat sangat menentukan keteraturan dan kualitas hasil akhir sistem yang dibangun. Penelitian ini menggunakan metode Waterfall sebagai pendekatan pengembangan karena seluruh kebutuhan sistem telah teridentifikasi secara lengkap sejak tahap awal, sehingga pengembangan dapat dilaksanakan secara bertahap dan terstruktur dari satu fase ke fase berikutnya. Waterfall merupakan model pengembangan tradisional yang bersifat linier dan berurutan, di mana setiap tahapan seperti analisis, desain, implementasi, dan pengujian dilakukan secara bertahap dan tidak dapat kembali ke tahap sebelumnya. Keunggulan Waterfall terletak pada dokumentasi yang jelas dan struktur proyek yang rapi, yang sangat sesuai untuk proyek dengan kebutuhan tetap dan pengawasan ketat.

Oleh karena itu, model ini banyak digunakan dalam proyek yang memerlukan pelacakan formal serta verifikasi terhadap setiap tahapan [13].

Metode Waterfall sering disebut sebagai pengembangan air terjun karena bergerak maju dari fase ke fase dengan cara yang sama seperti air terjun yang mengalir ke satu arah. Gambar 2.1 berikut ini menunjukkan tahapan pengembangan sistem dengan metode Waterfall Model



Gambar 2. 1 Tahapan Metode Waterfall[13]

1. *Requirement Definition*(Definisi Kebutuhan)

Tahap pertama adalah pendefinisian kebutuhan sistem secara menyeluruh. Pada tahap ini dilakukan identifikasi kebutuhan fungsional dan non-fungsional dari tiga aktor utama, yaitu Member Gym, Non-Member Gym, dan Admin (Pengelola Gym). Kebutuhan fungsional User meliputi registrasi akun, login, absensi berbasis QR Code, pemilihan program latihan, serta pemantauan pencapaian gamifikasi. Kebutuhan fungsional Admin meliputi pengelolaan data member, program latihan, konfigurasi gamifikasi, data absensi, dan laporan aktivitas.

2. *System and Software Design* (Perancangan Sistem dan Perangkat Lunak)

Pada tahap kedua, seluruh kebutuhan yang telah didefinisikan diterjemahkan ke dalam rancangan teknis yang menyeluruh. Kegiatan pada tahap ini mencakup perancangan *Use Case Diagram* menggunakan UML untuk mendefinisikan interaksi fungsional sistem, perancangan *Entity Relationship Diagram* (ERD) sebagai cetak biru struktur basis data MySQL, perancangan arsitektur REST API beserta daftar *endpoint* yang dibutuhkan, serta perancangan antarmuka pengguna (*user interface*) seluruh halaman aplikasi. Seluruh rancangan ini menjadi acuan teknis yang mengikat pada tahap implementasi selanjutnya.

3. *Implementation and Unit Testing* (Implementasi dan Pengujian Unit)

Tahap ketiga adalah perwujudan seluruh rancangan menjadi kode program yang dapat dieksekusi. Pengembangan dilakukan menggunakan bahasa Dart melalui *framework* Flutter untuk sisi antarmuka aplikasi Android, serta REST API yang berjalan di atas server lokal XAMPP sebagai lapisan *backend* yang menghubungkan aplikasi dengan basis data MySQL. Implementasi mencakup pengembangan seluruh halaman antarmuka, pembangunan *endpoint* REST API, implementasi basis data MySQL, integrasi QR Code menggunakan *package mobile_scanner* dan *qr_flutter*, serta implementasi mekanisme gamifikasi otomatis pada sisi *backend*.

4. *Integration and System Testing* (Integrasi dan Pengujian Sistem)

Pada tahap keempat, seluruh komponen yang telah diimplementasikan secara terpisah diintegrasikan menjadi satu kesatuan sistem yang utuh, kemudian diuji secara menyeluruh menggunakan metode *Blackbox Testing*. Pengujian dilakukan dengan menyusun skenario uji untuk setiap fitur *User* dan *Admin*, mengeksekusi setiap skenario, lalu memverifikasi kesesuaian antara hasil aktual dengan hasil yang diharapkan. Fitur yang tidak lulus pengujian diperbaiki dan diuji ulang hingga seluruh skenario pengujian mencapai status valid.

5. *Operation and Maintenance* (Operasi dan Pemeliharaan)

Tahap terakhir secara umum mencakup pengoperasian sistem pada lingkungan nyata beserta kegiatan pemeliharaan berkelanjutan, seperti pemantauan performa aplikasi dan basis data, perbaikan *bug* pascaoperasional, serta pembaruan sistem sesuai kebutuhan pengguna yang berkembang dari waktu ke waktu.

2.4. Teknologi yang Digunakan dalam Pengembangan Sistem

Pengembangan aplikasi ini memanfaatkan beberapa teknologi yang dipilih berdasarkan kesesuaian dengan kebutuhan sistem, kemudahan integrasi, dan kematangan ekosistem pengembangannya.

Teknologi-teknologi tersebut saling terintegrasi dalam satu arsitektur sistem yang utuh, mulai dari sisi antarmuka pengguna pada perangkat mobile, lapisan komunikasi data, hingga penyimpanan data pada sisi server. Berikut ini adalah penjelasan mengenai setiap teknologi yang digunakan dalam penelitian ini.

1. Flutter

Flutter adalah framework pengembangan aplikasi mobile lintas platform yang dibuat oleh Google dan pertama kali dirilis dalam versi stabil pada tahun 2018. Framework ini menggunakan bahasa pemrograman Dart dan menerapkan paradigma berbasis widget, di mana setiap elemen antarmuka pengguna dibangun dari komponen widget yang modular dan dapat dikustomisasi secara bebas, dengan pendekatan rendering langsung ke kanvas grafis tanpa bergantung pada komponen bawaan sistem operasi sehingga menghasilkan tampilan yang konsisten dan performa yang tinggi di berbagai perangkat Android [14]. Penelitian ini memilih Flutter sebagai framework pengembangan antarmuka aplikasi Android karena kemampuannya menghasilkan aplikasi dengan performa mendekati native dari satu basis kode tunggal, sehingga efisiensi pengembangan meningkat secara signifikan, ditambah fitur hot reload yang memungkinkan pengembang melihat dampak perubahan kode secara instan tanpa harus menghentikan proses build sehingga mempercepat siklus iterasi pengembangan secara nyata [14]. Dalam penelitian ini, Flutter digunakan untuk membangun seluruh antarmuka pengguna (user interface) aplikasi yang berjalan pada platform Android, memanfaatkan kemampuan lintas platform dan performa native yang menjadi karakteristik utama framework tersebut [14].

2. Dart

Dart adalah bahasa pemrograman yang dikembangkan oleh Google dan dirancang khusus untuk mendukung pengembangan aplikasi klien modern pada berbagai platform, termasuk mobile, web, dan desktop. Dart bersifat strongly-typed dengan sintaks yang familier bagi pengembang yang berpengalaman dengan bahasa seperti Java atau C#, sehingga kurva belajarnya relatif tidak terlalu curam, dan mendukung pemrograman berorientasi objek secara penuh serta pemrograman asinkron melalui mekanisme async/await yang penting dalam pengembangan aplikasi mobile [14]. Dalam konteks Flutter, kode Dart dikompilasi menjadi kode native ARM untuk platform mobile sehingga menghasilkan performa eksekusi yang tinggi di perangkat Android [14].

3. MySQL

MySQL adalah sistem manajemen basis data relasional yang bersifat open-source dan dikelola oleh Oracle Corporation. MySQL menggunakan Structured Query Language (SQL) sebagai antarmuka standar untuk seluruh operasi terhadap data, mulai dari pembuatan struktur tabel, penyisipan, pembaruan, hingga penghapusan dan pengambilan data dengan berbagai kondisi yang kompleks, dan keandalan, kecepatan pemrosesan query, serta kemudahan administrasinya menjadikan MySQL sebagai pilihan basis data yang sangat populer dalam pengembangan sistem informasi berbasis mobile [15]. Dalam arsitektur penelitian ini, MySQL yang berjalan di atas

server lokal XAMPP berperan sebagai repositori terpusat yang menyimpan seluruh data operasional sistem, mencakup informasi pengguna dan keanggotaan, program latihan, catatan absensi berbasis QR Code, serta data gamifikasi seperti poin, level, progress bar, streak, dan achievement setiap anggota.

4. XAMPP

XAMPP adalah paket perangkat lunak server lokal yang dikembangkan oleh Apache Friends, mencakup komponen-komponen utama yang dibutuhkan untuk menjalankan aplikasi web secara lokal, yaitu Apache sebagai web server dan MySQL sebagai sistem manajemen basis data. XAMPP merupakan solusi yang lazim digunakan pengembang untuk membangun dan menguji aplikasi di lingkungan lokal sebelum dideploy ke server produksi. Dalam penelitian ini, XAMPP digunakan sebagai server lokal yang menjalankan REST API dan basis data MySQL selama proses pengembangan dan pengujian aplikasi, serta menyediakan lingkungan pengembangan yang terisolasi dan mudah dikonfigurasi, sehingga seluruh komponen backend dapat berjalan secara terintegrasi di mesin lokal pengembang dan memudahkan proses debugging.

5. QR Code

QR Code atau Quick Response Code adalah kode matriks dua dimensi yang dikembangkan oleh Denso Wave di Jepang pada tahun 1994, dapat dibaca dengan cepat melalui kamera smartphone, serta memiliki kapasitas penyimpanan data yang lebih besar dibandingkan barcode konvensional [17]. Penelitian ini memilih QR Code sebagai solusi teknologi absensi karena pemindaian dapat dilakukan langsung melalui kamera smartphone tanpa perangkat keras tambahan, menggunakan satu kode QR statis yang ditempatkan permanen di area gym untuk seluruh anggota. Mekanismenya: setelah login, anggota memindai QR Code sebagai tanda masuk sehingga waktu masuk tersimpan otomatis ke basis data, dan saat selesai berolahraga, anggota menekan tombol pencatatan keluar pada kartu kehadiran di halaman Dashboard untuk mencatat waktu keluar. Sistem ini terbukti mampu menggantikan pencatatan kehadiran manual yang rentan kesalahan menjadi proses yang lebih akurat dan real-time [8].

2.5. Gamifikasi

Mempertahankan motivasi pengguna agar tetap konsisten dalam menggunakan sebuah aplikasi merupakan tantangan yang umum dihadapi dalam pengembangan sistem digital, termasuk aplikasi di bidang kebugaran [4]. Tanpa adanya dorongan yang berkelanjutan, pengguna cenderung kehilangan minat setelah periode penggunaan awal, sehingga manfaat jangka panjang dari aplikasi tersebut tidak dapat tercapai secara optimal. Permasalahan serupa juga banyak ditemukan pada

anggota gym, di mana motivasi untuk berlatih secara rutin cenderung menurun seiring berjalannya waktu apabila tidak disertai bentuk penghargaan atau umpan balik yang jelas atas usaha yang telah dilakukan [5]. Ketidakkonsistenan ini pada akhirnya berdampak pada rendahnya keberhasilan pencapaian target kebugaran serta meningkatnya kemungkinan anggota berhenti menggunakan layanan gym secara keseluruhan. Salah satu pendekatan yang banyak digunakan untuk mengatasi tantangan tersebut adalah gamifikasi, yaitu penerapan elemen-elemen permainan ke dalam konteks aktivitas yang bukan permainan [4]. Gamifikasi dibahas dalam penelitian ini sebagai pendekatan yang digunakan untuk menjawab tantangan tersebut, dengan menerapkan mekanisme penghargaan berbasis elemen permainan ke dalam aktivitas latihan anggota gym agar motivasi dan konsistensi latihan dapat dipertahankan dalam jangka panjang.

Gamifikasi adalah penerapan elemen-elemen desain yang lazim ditemukan dalam dunia permainan ke dalam konteks aktivitas yang bukan permainan, dengan tujuan utama meningkatkan motivasi intrinsik, keterlibatan aktif, dan konsistensi perilaku pengguna [4]. Penerapan gamifikasi bukan berarti mengubah sebuah aplikasi menjadi permainan secara utuh, melainkan mengambil konsep penghargaan dan umpan balik yang biasa ditemukan dalam permainan, seperti poin, level, dan pencapaian, untuk kemudian diterapkan ke dalam konteks non-permainan [4]. Melalui penerapan elemen-elemen tersebut, pengguna terdorong untuk tetap aktif secara rutin karena setiap tindakan yang dilakukan memperoleh respons dan penghargaan yang dapat dirasakan secara langsung [18]. Elemen gamifikasi yang umum diterapkan pada berbagai jenis aplikasi antara lain poin sebagai satuan penghargaan atas suatu tindakan, level sebagai penanda tingkat perkembangan pengguna, *progress bar* sebagai indikator kemajuan visual, *streak* sebagai pencatat konsistensi penggunaan, dan *achievement* sebagai penghargaan atas pencapaian target tertentu [4]. Kelima elemen tersebut saling berkaitan satu sama lain dan bekerja secara bersamaan untuk membentuk sebuah sistem penghargaan yang berkesinambungan bagi pengguna aplikasi. Dalam pengembangan aplikasi MissionFit, kelima elemen tersebut diadaptasi dan diterapkan pada aktivitas latihan anggota gym, sehingga setiap kehadiran dan penyelesaian sesi latihan yang dilakukan oleh anggota dapat memperoleh penghargaan yang terukur secara sistematis.

Dalam ranah aplikasi kebugaran, gamifikasi terbukti efektif untuk mengatasi masalah penurunan motivasi dan ketidakkonsistenan latihan yang menjadi tantangan paling umum di kalangan anggota gym [5]. Penguatan positif berupa penghargaan yang diberikan segera setelah pengguna menyelesaikan suatu aktivitas mendorong terbentuknya kebiasaan latihan yang berkelanjutan dalam jangka panjang [6]. Selain pada aplikasi kebugaran secara umum, penerapan gamifikasi pada aplikasi olahraga berbasis *mobile* juga terbukti mampu membuat penggunaannya lebih konsisten dalam menjalankan aktivitas fisik [5]. Sistem gamifikasi dalam penelitian ini

dirancang bersifat personal dan non-kompetitif, sehingga tidak terdapat *leaderboard* yang membandingkan pencapaian antara satu anggota dengan anggota lainnya [18]. Pendekatan personal ini dipilih karena setiap anggota gym memiliki kondisi fisik, target latihan, dan ritme perkembangan yang berbeda-beda, sehingga perbandingan langsung antarpengguna berpotensi menimbulkan tekanan psikologis alih-alih motivasi [18]. Dengan demikian, setiap anggota hanya berfokus pada perkembangan dirinya sendiri melalui poin, level, *progress bar*, *streak*, dan *achievement* yang diperolehnya, tanpa merasa perlu bersaing dengan anggota lain untuk tetap termotivasi berlatih. Efektivitas gamifikasi dalam penelitian ini juga bertumpu pada Self-Determination Theory, yang menjelaskan bahwa motivasi intrinsik seseorang berkembang melalui pemenuhan tiga kebutuhan psikologis dasar, yaitu kompetensi, otonomi, dan keterhubungan [18]. Kompetensi merujuk pada kebutuhan seseorang untuk merasakan kemajuan dan penguasaan atas suatu aktivitas yang dilakukannya [18]. Otonomi merujuk pada kebutuhan untuk merasa memiliki kendali atas pilihan dan tindakan yang diambil, bukan sekadar mengikuti instruksi yang ditetapkan pihak lain [18]. Keterhubungan merujuk pada kebutuhan untuk merasa menjadi bagian dari sesuatu yang bermakna, baik melalui interaksi sosial maupun elemen naratif dalam suatu sistem [18]. Studi eksperimental oleh Sailer, Hense, Mayr, dan Mandl (2017) menunjukkan bahwa kebutuhan kompetensi secara konsisten dipenuhi oleh elemen seperti badge atau achievement, leaderboard, dan performance graph atau progress bar, sedangkan kebutuhan keterhubungan lebih dipengaruhi oleh elemen sosial seperti avatar, narasi, dan interaksi antaranggota tim [23]. Berdasarkan temuan tersebut, kebutuhan kompetensi dalam penelitian ini dipenuhi melalui elemen poin, level, progress bar, dan achievement, sedangkan kebutuhan otonomi dipenuhi melalui kebebasan anggota dalam memilih program latihan sesuai tujuan pribadinya. Mengingat aplikasi MissionFit dirancang bersifat personal dan tidak menerapkan elemen sosial seperti leaderboard, avatar, maupun interaksi antaranggota, kebutuhan keterhubungan tidak menjadi fokus utama dalam perancangan gamifikasi pada penelitian ini. Ketika kebutuhan kompetensi dan otonomi tersebut terpenuhi, motivasi pengguna untuk terus menggunakan aplikasi cenderung bertahan lebih lama dibandingkan motivasi yang murni bergantung pada dorongan eksternal semata [18].

Penerapan gamifikasi pada aplikasi olahraga berbasis *mobile* secara khusus juga terbukti dapat membuat pengguna lebih konsisten dalam berolahraga [5]. Konsistensi tersebut terbentuk karena pengguna memperoleh umpan balik yang jelas atas setiap aktivitas yang telah diselesaikan, alih-alih hanya mencatat data latihan tanpa adanya bentuk apresiasi apa pun [5]. Dalam penelitian ini, sistem gamifikasi dirancang bersifat personal dan tidak kompetitif, di mana setiap anggota gym hanya dapat melihat perkembangan dirinya sendiri melalui elemen poin, level, *progress bar*, *streak*, dan *achievement*, tanpa adanya perbandingan dengan pengguna lain. Pendekatan ini dipilih

agar setiap anggota dapat berfokus pada perkembangan dirinya sendiri dan termotivasi untuk terus hadir serta menyelesaikan sesi latihan secara konsisten. Dibandingkan dengan pendekatan kompetitif berbasis *leaderboard*, pendekatan personal dinilai lebih sesuai diterapkan pada konteks gym yang anggotanya memiliki latar belakang kebugaran dan tujuan latihan yang beragam [18]. Selain berlandaskan *Self-Determination Theory*, mekanisme gamifikasi juga sejalan dengan prinsip penguatan positif dalam teori perilaku, yaitu pemberian respons yang menyenangkan segera setelah suatu tindakan dilakukan untuk mendorong pengulangan tindakan tersebut di masa mendatang [4]. Penguatan positif berupa penghargaan yang diberikan segera setelah pengguna menyelesaikan suatu aktivitas terbukti mendorong terbentuknya kebiasaan yang berkelanjutan dalam konteks aplikasi berbasis gamifikasi [6]. Dalam aplikasi MissionFit, prinsip ini diterapkan melalui pemberian poin secara otomatis setiap kali anggota melakukan absensi maupun menyelesaikan sesi latihan, sehingga anggota memperoleh umpan balik instan atas usaha yang telah dilakukan. Kombinasi antara pemenuhan kebutuhan psikologis dasar dan penguatan positif inilah yang menjadi dasar teoretis bagi perancangan kelima elemen gamifikasi berikut ini. Kelima elemen tersebut dirancang agar saling melengkapi, sehingga anggota gym dapat memperoleh penghargaan dalam berbagai bentuk dan skala waktu, mulai dari penghargaan instan setelah satu sesi latihan hingga penghargaan jangka panjang setelah mencapai target tertentu. Dengan demikian, motivasi anggota untuk tetap berlatih secara rutin diharapkan dapat terjaga secara berkelanjutan, tidak hanya bergantung pada motivasi awal ketika pertama kali bergabung menjadi anggota gym.

Berikut ini adalah penjelasan mengenai setiap elemen gamifikasi yang diterapkan pada aplikasi MissionFit.

1. Level

Secara umum, level pada suatu sistem gamifikasi dihitung berdasarkan akumulasi poin yang telah dikumpulkan pengguna, di mana kenaikan level terjadi secara otomatis setiap kali pengguna mencapai ambang batas poin tertentu yang telah ditetapkan, dimulai dari level awal bagi pengguna yang baru bergabung. Kenaikan level ini memberikan pengguna rasa kompetensi (*competence*) yang terus bertambah seiring berjalannya waktu, sejalan dengan kebutuhan psikologis dasar dalam *Self-Determination Theory* [23].

2. Streak

Streak adalah elemen gamifikasi yang mencatat jumlah hari berturut-turut anggota melakukan absensi ke gym tanpa terputus dan menyelesaikan sesi latihan pada hari tersebut [19]. Kekuatan psikologis *streak* bersumber pada prinsip *loss aversion*, yaitu kecenderungan manusia merasakan

kehilangan lebih kuat dibandingkan perasaan mendapatkan sesuatu dengan nilai setara [19]. Semakin panjang *streak* yang dimiliki, semakin kuat dorongan untuk mempertahankannya, sehingga secara organik membentuk kebiasaan latihan yang konsisten [19].

3. Progress Bar

Progress Bar adalah elemen gamifikasi yang menampilkan kemajuan pengguna secara visual dalam bentuk indikator yang menunjukkan seberapa jauh pengguna telah berkembang menuju level berikutnya [4]. Elemen ini bekerja dengan cara menghitung akumulasi poin yang telah dikumpulkan dan menampilkannya dalam bentuk visual yang mudah dipahami, sehingga membantu pengguna memahami kemajuannya tanpa perlu melakukan perhitungan tambahan [4]. Indikator kemajuan visual semacam ini (*performance graph*) terbukti secara empiris memperkuat rasa kompetensi pengguna, sejalan dengan kebutuhan psikologis dasar dalam Self-Determination Theory [23].

4. Poin

Poin adalah unit penghargaan numerik yang diberikan kepada pengguna secara otomatis atas suatu tindakan yang dilakukan [4]. Sebagai elemen paling fundamental dalam sistem gamifikasi, poin berfungsi sebagai umpan balik instan yang mengonfirmasi bahwa tindakan yang dilakukan pengguna bernilai positif dalam sistem, sejalan dengan prinsip penguatan positif [6]. Poin juga berfungsi sebagai satuan dasar yang menghubungkan seluruh elemen gamifikasi lain dalam suatu aplikasi, karena elemen seperti level maupun progress bar umumnya dihitung berdasarkan akumulasi poin yang telah diperoleh pengguna.

5. Achievement

Achievement merupakan elemen gamifikasi yang berfungsi sebagai bentuk penghargaan atas pencapaian tertentu yang berhasil diraih oleh pengguna dalam sistem [4]. Berbeda dengan poin yang diberikan secara rutin atas setiap tindakan kecil, achievement diberikan secara khusus ketika pengguna mencapai target atau milestone tertentu berdasarkan aktivitas yang telah dilakukannya, sehingga pencapaiannya dirasakan sebagai pencapaian besar yang layak dirayakan [18]. Elemen *badge* atau achievement semacam ini terbukti secara empiris memperkuat rasa kompetensi (*competence*) pengguna, sejalan dengan kebutuhan psikologis dasar dalam Self-Determination Theory [23].

2.6. Unified Modeling Language (UML)

Sebelum proses implementasi dilaksanakan, sistem yang akan dibangun perlu dimodelkan terlebih dahulu agar struktur dan alur kerjanya dapat dipahami secara menyeluruh. UML digunakan dalam penelitian ini sebagai standar pemodelan sistem untuk mendokumentasikan kebutuhan fungsional antara aktor dengan sistem melalui Use Case Diagram, serta alur proses sistem melalui Activity Diagram.

Pada bagian ini akan dibahas mengenai Unified Modeling Language (UML) yang merupakan salah satu alat bantu utama dalam proses perancangan sistem berbasis Object-Oriented. UML digunakan untuk memvisualisasikan, merancang, serta mendokumentasikan struktur dan perilaku sistem secara lebih terstruktur dan mudah dipahami.

2.6.1 Pengertian Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan bahasa pemodelan berstandar grafis yang digunakan untuk memvisualisasikan, menspesifikasikan, merancang, dan mendokumentasikan artefak dari sistem perangkat lunak berbasis orientasi objek (*Object-Oriented*) [20]. Notasi UML menggambarkan struktur statis maupun perilaku dinamis sistem sehingga mempermudah pemahaman arsitektur perangkat lunak secara menyeluruh [20].

UML menyediakan berbagai jenis diagram untuk memodelkan spesifikasi sistem, seperti *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *Sequence Diagram*, hingga *Deployment Diagram* [20]. Pada penelitian ini, jenis diagram UML yang dimanfaatkan dibatasi pada *Use Case Diagram* untuk memodelkan fungsionalitas sistem dan *Activity Diagram* untuk menggambarkan alur proses kerja sistem [20].

Salah satu tahapan penting dalam SDLC adalah tahap desain sistem yang bertujuan agar perangkat lunak yang dikembangkan mampu memenuhi kebutuhan pengguna dan memiliki kinerja yang handal. Dalam tahap ini, kebutuhan pengguna baik secara fungsional maupun non-fungsional ditransformasikan ke dalam model yang menggambarkan bagaimana sistem akan dibangun dan dijalankan [20].

2.6.2 Jenis-Jenis Diagram Unified Modeling Language (UML)

Terdapat beberapa jenis-jenis diagram UML beserta penjelasannya [20]:

1. Use Case Diagram yaitu diagram yang digunakan untuk menggambarkan interaksi antara sistem dengan pengguna (user) atau sistem lain. Diagram ini sangat berguna untuk memetakan kebutuhan (requirement) dari suatu sistem.

2. Activity Diagram yaitu diagram yang digunakan untuk menggambarkan alur aktivitas atau alur kerja (workflow) secara sekuensial maupun paralel dalam sistem, termasuk percabangan keputusan yang terjadi dalam suatu proses.
3. Class Diagram yaitu diagram yang digunakan untuk menggambarkan kelas (class), tipe, interface, serta hubungan antar ketiganya. Diagram ini merupakan inti dari pemodelan berorientasi objek.
4. Object Diagram yaitu diagram yang digunakan untuk menggambarkan instance dari objek-objek yang telah didefinisikan pada class diagram serta memperlihatkan hubungan antar objek tersebut.
5. Sequence Diagram yaitu diagram yang digunakan untuk menggambarkan interaksi antar objek di dalam sistem, dengan menekankan pada urutan pesan atau interaksi yang terjadi dari waktu ke waktu.
6. Communication Diagram yaitu diagram yang digunakan untuk menggambarkan cara objek berinteraksi satu sama lain serta koneksi yang dibutuhkan untuk melakukan interaksi tersebut.
7. Timing Diagram yaitu diagram yang digunakan untuk menggambarkan interaksi antar objek dengan menekankan aspek waktu sebagai faktor yang penting dalam proses komunikasi.
8. Component Diagram yaitu diagram yang digunakan untuk menggambarkan komponen-komponen penting dalam sistem serta interface yang digunakan untuk saling berinteraksi antar komponen.
9. Package Diagram yaitu diagram yang digunakan untuk menggambarkan struktur hierarki dan pengelompokan dari sekumpulan class atau component dalam sistem.
10. State Machine Diagram yaitu diagram yang digunakan untuk menggambarkan berbagai keadaan (state) dari suatu objek selama masa aktifnya serta peristiwa yang dapat menyebabkan perubahan keadaan tersebut.
11. Deployment Diagram yaitu diagram yang digunakan untuk menggambarkan bagaimana sistem diimplementasikan di lingkungan nyata, termasuk perangkat keras dan hubungan antar komponennya.

Pada penelitian ini, jenis diagram UML yang digunakan adalah Use Case Diagram dan Activity Diagram. Use Case Diagram digunakan untuk memodelkan fungsionalitas sistem dari sudut pandang tiga aktor utama yaitu Member Gym, Non-Member Gym dan Admin (Pengelola Gym), sedangkan Activity Diagram digunakan untuk menggambarkan alur proses dari setiap fungsionalitas yang dikembangkan pada Aplikasi Mobile ini.

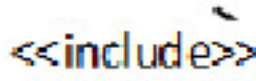
2.6.3 Use Case Diagram

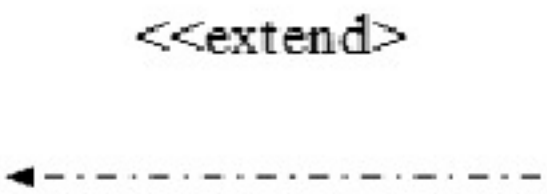
Use Case Diagram adalah jenis diagram UML yang menggambarkan kebutuhan fungsional sistem dari sudut pandang pengguna akhir. Diagram ini terdiri dari tiga elemen utama: aktor yang merepresentasikan pihak eksternal yang berinteraksi dengan sistem, use case yang merepresentasikan fungsi atau layanan yang disediakan sistem, serta garis relasi yang menggambarkan hubungan antara aktor dan use case maupun antar use case [20].

Use Case Diagram merupakan titik awal yang krusial dalam pengembangan sistem karena mampu menyajikan gambaran fungsionalitas yang intuitif dan mudah dipahami tanpa harus memiliki latar belakang teknis. Manfaat penggunaan Use Case Diagram dalam penelitian ini antara lain memastikan seluruh kebutuhan fungsional sistem telah teridentifikasi sebelum perancangan teknis dimulai, memberikan gambaran visual yang jelas mengenai hak akses setiap aktor, menjadi acuan penyusunan skenario uji Blackbox Testing, serta mempertegas batasan sistem secara eksplisit [20]. Dalam penelitian ini, Use Case Diagram digunakan untuk memodelkan seluruh interaksi antara Member Gym, Non-Member Gym dan Pengelola Gym (Admin) dengan fungsionalitas.

Simbol-simbol yang digunakan dalam Use Case Diagram beserta deskripsinya ditunjukkan pada Tabel 2.1 [20]

Tabel 2. 1 Simbol Use Case Diagram [20]

Nama Simbol	Simbol	Deskripsi
Use Case		Menggambarkan fungsi atau layanan yang disediakan oleh sistem kepada aktor. Use case biasanya ditulis dengan kata kerja aktif di awal frase
Aktor		Merepresentasikan pengguna atau sistem eksternal yang berinteraksi dengan sistem. Dalam aplikasi ini, aktor terdiri dari Member Gym, Non-Member Gym dan Admin Pengelola Gym
Asosiasi		Garis penghubung yang menggambarkan hubungan komunikasi langsung antara aktor dengan use case
Include		Menunjukkan bahwa sebuah use case secara wajib memanggil use case lain sebagai bagian dari proses utamanya. Use case yang dipanggil selalu dieksekusi setiap kali use case pemanggil dijalankan

<i>Extend</i>		Menunjukkan bahwa sebuah use case dapat diperluas oleh use case lain secara opsional. Use case perluasan hanya dijalankan apabila kondisi tertentu terpenuhi, sehingga tidak selalu dieksekusi setiap kali use case utama dijalankan
Sistem		Kotak pembatas yang menggambarkan batas sistem. Seluruh use case yang berada di dalam kotak merupakan bagian dari fungsionalitas sistem yang dikembangkan

2.6.4 Activity Diagram

Activity Diagram merupakan salah satu jenis diagram dalam UML yang digunakan untuk menggambarkan alur aktivitas atau alur kerja (*workflow*) dari suatu proses dalam sistem secara sekuensial maupun paralel. Diagram ini memvisualisasikan urutan langkah-langkah yang dilakukan dari titik awal (*initial node*) hingga titik akhir (*activity final node*), termasuk percabangan keputusan (*decision node*) yang menentukan jalur alur berdasarkan suatu kondisi, serta aktivitas yang dapat berlangsung secara bersamaan [20]. Dengan representasi visualnya yang terstruktur, Activity Diagram mempermudah pengembang dan pemangku kepentingan dalam memahami logika alur proses yang terdapat di dalam sistem.

Dalam penelitian ini, Activity Diagram digunakan untuk menggambarkan alur proses dari seluruh fungsionalitas yang terdapat dalam aplikasi MissionFit, mencakup proses yang dilakukan oleh User (Anggota Gym) maupun Admin (Pengelola Gym). Penggunaan Activity Diagram dipilih karena mampu menggambarkan interaksi antara dua aktor sekaligus melalui notasi swimlane, sehingga tanggung jawab setiap aktor dalam suatu proses dapat terlihat dengan jelas. Simbol-simbol yang digunakan dalam Activity Diagram beserta deskripsinya ditunjukkan pada Tabel 2.2.[20]

Tabel 2. 2 Simbol Activity Diagram[20]

Nama Simbol	Simbol	Deskripsi
Status Awal		Menandai titik awal dari suatu aktivitas atau alur kerja.
Aktivitas		Menggambarkan pekerjaan atau tugas yang dilakukan dalam alur proses.

Keputusan/ Decision		Titik percabangan yang menentukan jalur alur selanjutnya berdasarkan suatu kondisi tertentu.
Swimlane		Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
Penggabungan/Join		Digunakan untuk menggabungkan kembali beberapa aliran paralel menjadi satu aliran tunggal
Status Akhir		Menandai titik akhir dari seluruh proses aktivitas dalam sistem.

2.7. Pengujian Perangkat Lunak (Black Box Testing)

Proses pengujian merupakan tahapan yang tidak dapat dilewati dalam pengembangan perangkat lunak karena menentukan apakah sistem yang dibangun telah memenuhi seluruh kebutuhan yang ditetapkan. Penelitian ini menggunakan metode Black Box Testing sebagai pendekatan pengujian yang berfokus pada validasi fungsionalitas dari sudut pandang pengguna akhir tanpa memperhatikan struktur internal kode program.

Pengujian perangkat lunak merupakan proses evaluasi sistematis untuk memastikan sistem bebas dari cacat fungsional sebelum dioperasikan oleh pengguna [21]. Penelitian ini menerapkan teknik *Black Box Testing*, yaitu metode pengujian yang berfokus pada validasi masukan (*input*) dan keluaran (*output*) antarmuka tanpa perlu menginspeksi struktur internal kode program [21]. Keunggulan pengujian ini terletak pada objektivitas evaluasi karena dilakukan murni dari sudut pandang pengguna akhir [21].

Dalam penelitian ini, Black Box Testing diterapkan untuk memvalidasi seluruh fungsionalitas Aplikasi Mobile ini, mencakup fitur autentikasi (login email/password), mekanisme gamifikasi (poin, level, Progress Bar, achievement, streak), absensi berbasis QR Code, dan seluruh fitur manajemen pada panel Admin. Pengujian dilakukan dengan menyusun skenario uji untuk setiap fitur yang mencakup data masukan, hasil yang diharapkan berdasarkan spesifikasi, dan verifikasi hasil aktual. Pengujian dinyatakan berhasil apabila seluruh skenario menghasilkan status valid. Hasil pengujian selengkapnya akan disajikan pada Bab IV Penelitian Ini

2.8. System Usability Scale (SUS)

Usability menggambarkan sejauh mana suatu aplikasi dapat dioperasikan pengguna secara efektif, efisien, dan memuaskan [22]. Untuk mengukur tingkat kepuasan pengguna secara kuantitatif, penelitian ini menggunakan instrumen baku *System Usability Scale* (SUS) yang dikembangkan oleh John Brooke [22]. Instrumen ini terdiri atas 10 butir pernyataan berskala Likert 5 poin yang kemudian dikonversi menjadi skor akhir berskala 0–100 untuk menentukan kategori kelayakan sistem (*Adjective Rating* dan *Grade Scale*) [22].

Tingkat usability suatu sistem dapat dinilai berdasarkan lima komponen kualitas, yaitu Learnability, Efficiency, Memorability, Error, dan Satisfaction. Learnability mengacu pada tingkat kemudahan pengguna baru dalam memahami dan mempelajari fungsi dasar sistem pada saat pertama kali berinteraksi dengannya. Efficiency mengacu pada kecepatan pengguna dalam menyelesaikan suatu tugas setelah proses pembelajaran awal terhadap sistem selesai dilakukan. Memorability berkaitan dengan kemampuan pengguna untuk mengingat kembali cara pengoperasian sistem meskipun telah tidak menggunakannya dalam kurun waktu tertentu. Error mengacu pada frekuensi kesalahan yang dilakukan pengguna selama berinteraksi dengan sistem, tingkat keseriusan kesalahan tersebut, serta kemudahan pengguna dalam memulihkan diri dari kesalahan yang terjadi. Adapun Satisfaction berkaitan dengan tingkat kenyamanan subjektif yang dirasakan pengguna terhadap sistem secara keseluruhan.

Salah satu instrumen yang umum digunakan untuk mengukur tingkat usability secara kuantitatif adalah System Usability Scale (SUS), yang dikembangkan oleh John Brooke pada tahun 1996. SUS merupakan instrumen usability testing yang bersifat baku (standardized) dan telah banyak diaplikasikan secara luas dalam berbagai penelitian, baik pada evaluasi usability aplikasi berbasis mobile maupun berbasis web, termasuk di antaranya dalam pengukuran tingkat kepuasan pengguna pada aplikasi e-commerce Shopee [22]. Instrumen ini terdiri atas sepuluh butir pernyataan yang direspons oleh pengguna menggunakan skala Likert lima poin, dengan pilihan jawaban mulai dari Sangat Tidak Setuju hingga Sangat Setuju, yang masing-masing dikonversi ke dalam skor numerik 1 sampai dengan 5.

Perhitungan skor SUS dilakukan dengan terlebih dahulu mengonversi nilai jawaban tiap responden ke dalam skor kontribusi pada rentang 0 sampai 4. Pada pernyataan bernomor ganjil, skor kontribusi diperoleh melalui pengurangan nilai jawaban dengan satu ($n-1$), sedangkan pada pernyataan bernomor genap, skor kontribusi diperoleh dari selisih antara lima dengan nilai jawaban ($5-n$). Ketentuan ini diterapkan mengingat pernyataan pada SUS disusun secara berselang-seling antara pernyataan bermuatan positif dan negatif, sehingga proses konversi tersebut diperlukan untuk menyamakan arah interpretasi seluruh butir pernyataan. Seluruh skor kontribusi dari sepuluh

butir pernyataan kemudian dijumlahkan dan dikalikan dengan konstanta 2,5, sehingga diperoleh skor akhir SUS untuk masing-masing responden dalam rentang 0 hingga 100. Skor SUS pada tingkat populasi selanjutnya diperoleh melalui perhitungan nilai rata-rata dari keseluruhan skor responden.

Skor akhir SUS yang telah diperoleh kemudian diinterpretasikan dengan mengacu pada tiga parameter penilaian, yaitu Acceptability Ranges, Grade Scale, dan Adjective Ratings, sebagaimana disajikan pada tabel berikut.[22]

Tabel 2. 3 Interpretasi Skor System Usability Scale (SUS)

Skor SUS	Grade	Kategori (Adjective Rating)
> 80,3	A	Excellent
68 – 80,3	B	Good
51 – 68	C	OK / Acceptable
< 51	D – F	Poor

