

BAB II

KAJIAN LITERATUR

2.1 Aplikasi *Mobile*

Aplikasi *mobile* merupakan perangkat lunak yang dirancang secara spesifik untuk beroperasi pada perangkat komputasi portabel seperti telepon pintar (*smartphone*) dan tablet. Secara umum, pengembangan perangkat lunak dikategorikan ke dalam tiga *platform* utama, di mana masing-masing memiliki karakteristik lingkungan operasional yang berbeda [10]. Berikut adalah penjabaran dari ketiga kategori tersebut:

1. Aplikasi *Desktop*: Aplikasi ini dirancang untuk dijalankan pada komputer pribadi (*Personal Computer* atau laptop) dengan sistem operasi seperti Windows, macOS, atau Linux. Karakteristik utamanya adalah mampu memanfaatkan sumber daya komputasi yang besar dan performa tinggi, namun bersifat statis atau terpaku pada lokasi fisik perangkat.
2. Aplikasi Web: Aplikasi ini diakses melalui peramban (*web browser*) dan tidak memerlukan instalasi pada perangkat pengguna. Keunggulannya terletak pada kompatibilitas lintas *platform*, namun kinerjanya sangat bergantung pada konektivitas internet dan memiliki akses yang terbatas terhadap fitur perangkat keras.
3. Aplikasi *Mobile*: Aplikasi ini dibangun untuk perangkat genggam yang memiliki sumber daya terbatas (seperti baterai dan layar kecil) namun menawarkan mobilitas tinggi. Aplikasi *mobile* saat ini menjadi teknologi yang paling banyak digunakan karena kemampuannya memanfaatkan fitur perangkat keras secara optimal, seperti kamera, GPS, dan sensor gerak [10].

Das et al. [11] menjelaskan bahwa *smartphone* modern kini dianggap sebagai gabungan antara komputer genggam dan ponsel yang didukung kemampuan komputasi tinggi serta fitur perangkat keras canggih. Integrasi berbagai fitur seperti GPS dan konektivitas nirkabel ini menciptakan lingkungan ideal bagi aplikasi *mobile* untuk menyediakan konten spesifik dan kontekstual bagi pengguna, layaknya kemampuan sebuah komputer pribadi (*PC*).

Kemajuan perangkat ini mendorong ledakan adopsi aplikasi *mobile*, terutama sejak peluncuran sistem operasi modern seperti Android dan iOS yang mempermudah akses melalui toko aplikasi. Fenomena ini memperluas fungsi aplikasi dari sekadar hiburan ke

berbagai sektor vital seperti pendidikan, bisnis, dan kesehatan, di mana miliaran pengguna kini mengandalkan aplikasi untuk memfasilitasi kebutuhan harian mereka [10], [11].

Aplikasi *mobile* Android didefinisikan sebagai perangkat lunak yang dirancang secara khusus untuk beroperasi pada perangkat bergerak seperti telepon pintar (*smartphone*) dan tablet dengan sistem operasi Android. Rifa'i et al. [12] menjelaskan bahwa aplikasi berbasis Android dikembangkan dengan tujuan utama untuk memfasilitasi interaksi antara pengguna dan sistem layanan secara efisien, memanfaatkan adopsi luas sistem operasi Android di masyarakat untuk menyediakan solusi yang praktis dan mudah diakses.

Selain sebagai media interaksi, aplikasi Android juga berperan vital dalam transformasi layanan operasional menjadi lebih efektif. Putra [13] memaparkan bahwa implementasi aplikasi berbasis *mobile* mampu menyederhanakan proses yang sebelumnya manual seperti pendaftaran atau transaksi menjadi sistem digital yang terintegrasi, yang secara signifikan mempercepat akses pengguna terhadap informasi dan layanan tersebut secara *real-time*.

2.2 Android Studio

Android Studio adalah *Integrated Development Environment* (IDE) resmi yang dikembangkan oleh Google untuk memfasilitasi pembuatan aplikasi pada sistem operasi Android. Menurut dokumentasi resmi Android Developers [14], perangkat lunak ini dibangun di atas *platform* IntelliJ IDEA dan dirancang khusus untuk mempercepat proses pengembangan melalui antarmuka kerja yang terpadu. Sebagai standar industri, Android Studio menyediakan lingkungan yang komprehensif bagi pengembang untuk menulis kode, merancang antarmuka, dan mengelola sumber daya aplikasi dalam satu wadah yang efisien [15].



Gambar 2.1 Android Studio

Dalam proses pembuatan program, Android Studio menawarkan kemudahan melalui dukungan alat yang lengkap untuk setiap tahapan pengembangan. Erni et al. [16]

menjelaskan bahwa IDE ini dilengkapi dengan fitur canggih untuk mengembangkan (*developing*), menelusuri kesalahan (*debugging*), hingga mengemas aplikasi (*packaging*) menjadi format yang siap didistribusikan. Fitur-fitur ini sangat krusial untuk memastikan bahwa aplikasi dapat berjalan dengan stabil dan bebas dari kesalahan teknis sebelum digunakan oleh pengguna akhir.

Secara teknis, Android Studio mempermudah pengembang dengan memisahkan logika pemrograman dan desain tampilan. Rahmawanti et al. [17] memaparkan bahwa logika aplikasi umumnya dibangun menggunakan bahasa pemrograman seperti Java atau Kotlin, sedangkan tampilan antarmuka pengguna (*user interface*) disusun menggunakan *Extensible Markup Language* (XML). Integrasi yang kuat dengan sistem *build* otomatis berbasis Gradle juga memungkinkan pengembang untuk melakukan optimalisasi kinerja secara *native*, menjadikan proses pembuatan program Android lebih terstruktur dan efektif dibandingkan menggunakan kerangka kerja lain [15].

2.3 SQLite

SQLite merupakan sebuah sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*) yang bersifat *open-source* dan dirancang khusus untuk penyimpanan data lokal pada perangkat *mobile*. Abbas et al. [18] mendefinisikan SQLite sebagai mesin basis data yang "berdiri sendiri" (*self-contained*), tanpa *server* (*serverless*), dan tanpa konfigurasi (*zero-configuration*), yang berarti ia tidak memerlukan proses instalasi atau administrasi yang rumit seperti basis data tradisional pada umumnya. Karakteristik ini menjadikannya solusi yang sangat efisien dan ringan untuk ditanamkan langsung ke dalam aplikasi Android.



Gambar 2.2 SQLite

Dalam arsitektur pengembangan aplikasi *mobile*, SQLite berperan sebagai media penyimpanan data terstruktur yang andal dan persisten. Maulana et al. [19] menjelaskan bahwa SQLite memungkinkan aplikasi untuk menyimpan data penting seperti riwayat

transaksi, profil pengguna, atau data inventaris secara lokal di dalam memori perangkat, sehingga data tersebut tetap dapat diakses dengan cepat meskipun perangkat tidak terhubung ke jaringan internet (*offline*). Kemampuan ini sangat krusial untuk menjaga fungsionalitas aplikasi dan pengalaman pengguna yang responsif di berbagai kondisi jaringan.

Meskipun berukuran kecil, SQLite mendukung fitur-fitur basis data relasional standar yang lengkap. Kurnia [20] menyoroti bahwa SQLite mampu menangani operasi pengelolaan data yang kompleks seperti *Create, Read, Update, Delete* (CRUD) dan mendukung standar SQL, yang memudahkan pengembang dalam mengelola relasi antar data. Karena terintegrasi langsung dengan sistem operasi Android, SQLite menawarkan akses data yang lebih cepat dan penggunaan memori yang lebih rendah dibandingkan dengan solusi basis data eksternal lainnya, menjadikannya pilihan utama untuk pengembangan aplikasi yang memprioritaskan efisiensi sumber daya [18].

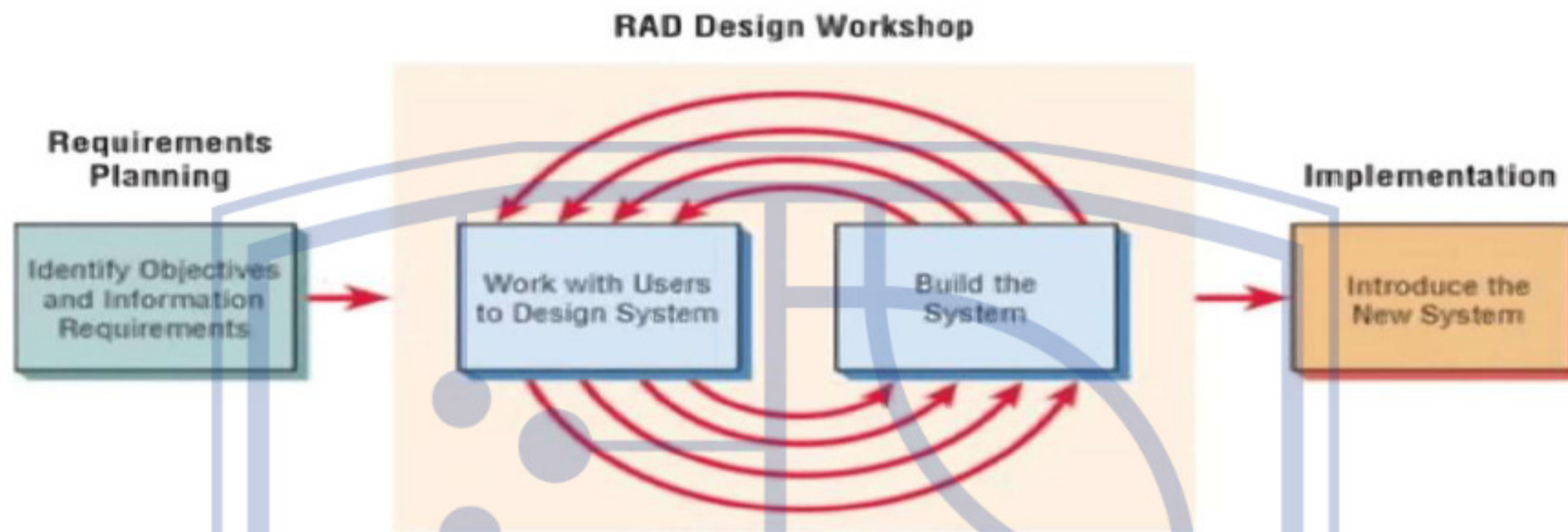
2.4 Metodologi *Rapid Application Development* (RAD)

Rapid Application Development (RAD) adalah metodologi pengembangan perangkat lunak yang menekankan pada siklus pengembangan yang singkat, cepat, dan berulang (*iterative*). Berbeda dengan metode tradisional yang kaku, RAD berfokus pada kecepatan penyampaian produk melalui penggunaan prototipe dan keterlibatan pengguna secara aktif selama proses perancangan. Pendekatan ini sangat cocok diterapkan untuk proyek yang membutuhkan waktu pengerjaan cepat dengan tetap memprioritaskan kualitas dan kepuasan pengguna melalui umpan balik yang intensif [21].

Metode ini memungkinkan pengembang untuk membuat perubahan fungsionalitas di tengah proses pengembangan tanpa harus mengulang dari awal, menjadikannya solusi ideal untuk pengembangan aplikasi *mobile* yang dinamis. Menurut Febiani et al. [22], terdapat tiga tahapan utama dalam siklus RAD, yaitu:

1. *Requirements Planning*: Tahap ini melibatkan metode pengumpulan data seperti observasi untuk mengetahui kebutuhan aplikasi yang diperlukan, kemudian dilanjutkan dengan penentuan kebutuhan fungsional dan kebutuhan non-fungsional.
2. *Design Workshop*: Pada tahap ini, dilakukan perancangan arsitektur sistem dan pembuatan prototipe antarmuka aplikasi. Pada tahap ini juga berlangsung proses *wireframe design* untuk antarmuka aplikasi dan perancangan struktur basis data aplikasi. Proses ini dapat diulang jika ditemukan ketidaksesuaian dengan *user design* yang sudah ditentukan.

3. *Implementation*: Setelah tahap *design workshop* diselesaikan, tahap selanjutnya adalah menerjemahkan rancangan tersebut ke dalam bentuk produk akhir. Tahap ini mencakup proses pengkodean (*coding*), serta pengujian sistem (*testing*) untuk memastikan tidak ada kesalahan (*bug*), hingga aplikasi siap untuk dioperasikan.



Gambar 2.3 Tahapan - Tahapan pada RAD

2.5 Black-box Testing

Black-box testing didefinisikan sebagai metode pengujian perangkat lunak yang berfokus sepenuhnya pada fungsionalitas eksternal aplikasi tanpa perlu mengetahui struktur kode internal atau logika pemrograman yang membangunnya. Hardika et al. [23] menjelaskan bahwa pendekatan ini memperlakukan perangkat lunak sebagai “kotak hitam”, di mana penguji hanya mengamati kesesuaian antara masukan (*input*) yang diberikan dengan keluaran (*output*) yang dihasilkan berdasarkan spesifikasi kebutuhan yang telah ditetapkan sebelumnya. Tujuan utamanya adalah untuk memastikan bahwa setiap fitur dalam aplikasi berjalan dengan benar dan sesuai dengan harapan pengguna [23].

Dalam konteks pengembangan aplikasi *mobile* Android, pengujian ini memegang peranan krusial untuk menjamin kualitas aplikasi sebelum dirilis. Maulana et al. [24] menekankan bahwa penerapan *black-box testing* membantu pengembang dalam mendeteksi dan meminimalkan kesalahan (*bug*) fungsional, kesalahan antarmuka, serta kesalahan kinerja sistem. Dengan berfokus pada perspektif pengguna akhir, pengujian ini memvalidasi alur kerja aplikasi mulai dari proses *login*, input data, hingga hasil akhir yang ditampilkan di layar perangkat *mobile* [24].

Salah satu teknik yang sering diterapkan dalam metode ini adalah *Equivalence Partitioning*. Frayudha et al. [25] memaparkan bahwa teknik ini dirancang untuk meningkatkan efisiensi pengujian dengan cara mengelompokkan data uji ke dalam kelas-kelas atau partisi yang setara. Dengan metode ini, pengujian tidak perlu dilakukan pada

setiap kemungkinan nilai input secara satu per satu, melainkan cukup mengambil perwakilan dari setiap partisi (data valid dan tidak valid) untuk memverifikasi apakah sistem meresponsnya dengan benar, sehingga proses pengujian menjadi lebih cepat namun tetap efektif [25].

Tabel 2.1 Contoh Pengujian Form Register Dengan Equivalence Partitioning

Nomor	Deskripsi Uji	Hasil yang Diharapkan	Hasil Uji	Kesimpulan
1	Input nama lengkap menggunakan karakter huruf (A-Z)	Sistem menerima data dan tidak muncul peringatan	Nama berhasil diinput dan tidak muncul peringatan	Sesuai
2	Input email tanpa menggunakan simbol '@' atau domain	Sistem menolak dan menampilkan pesan "Format email salah"	Muncul notifikasi bahwa email harus menggunakan '@'	Sesuai
3	Input kata sandi sesuai rentang (misal: 8-16 karakter)	Sistem menerima kata sandi dan melanjutkan proses	Kata sandi diterima dan indikator kekuatan sandi muncul	Sesuai
4	Input kata sandi di bawah batas minimum (kurang dari 8 karakter)	Sistem menampilkan pesan error "Kata sandi terlalu pendek"	Sistem menolak dan menampilkan pesan peringatan merah	Sesuai
5	Input nomor hp menggunakan karakter non-angka (huruf/simbol)	Sistem menampilkan pesan error "Hanya masukkan angka"	Muncul pesan "Input tidak valid" pada kolom nomor HP	Sesuai

2.6 Confusion Matrix

Dalam bidang pembelajaran mesin (*machine learning*) dan kecerdasan buatan, evaluasi kinerja sebuah model klasifikasi merupakan tahapan yang sangat krusial. *Confusion Matrix* (Matriks Kebingungan) adalah salah satu alat evaluasi analitis fundamental yang paling umum digunakan untuk mengukur dan memvisualisasikan kinerja algoritma tersebut[26]. Matriks ini divisualisasikan dalam bentuk tabel kontingensi dua dimensi yang secara langsung membandingkan kelas hasil prediksi dari model komputasi dengan kelas data aktual atau nilai kebenaran dasarnya (*ground truth*)[27]. Dengan memetakan jumlah prediksi yang benar maupun yang keliru ke dalam baris dan kolom, *confusion matrix* memungkinkan para peneliti untuk tidak hanya mengetahui seberapa banyak tingkat kebenaran secara umum, tetapi juga mengidentifikasi secara spesifik jenis dan pola kesalahan klasifikasi yang dihasilkan oleh sistem tersebut.

Pada penerapan klasifikasi biner, seperti halnya pada sistem yang bertugas mendeteksi apakah eksekusi gerakan latihan pengguna sudah benar atau salah, matriks ini memecah hasil prediksi ke dalam empat komponen utama [28], yaitu:

1. *True Positive* (TP): Jumlah data yang secara aktual termasuk dalam kelas positif dan model berhasil memprediksinya sebagai kelas positif dengan benar.
2. *True Negative* (TN): Jumlah data yang secara aktual termasuk dalam kelas negatif dan model berhasil memprediksinya sebagai kelas negatif dengan benar.
3. *False Positive* (FP): Jumlah data yang secara aktual termasuk dalam kelas negatif, namun model secara keliru memprediksinya sebagai kelas positif.
4. *False Negative* (FN): Jumlah data yang secara aktual termasuk dalam kelas positif, namun model secara keliru memprediksinya sebagai kelas negatif.

Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Gambar 2.4 Confusion Matrix

Berdasarkan keempat komponen fundamental tersebut, dapat diturunkan berbagai metrik evaluasi kinerja matematis untuk menilai keandalan model dari berbagai sudut pandang[28]. Metrik-metrik utama tersebut meliputi:

1. *Accuracy* (Akurasi)

Akurasi merupakan metrik komprehensif yang mengukur proporsi prediksi yang benar (baik positif maupun negatif) terhadap keseluruhan jumlah data yang dievaluasi.

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

2. *Precision* (Presisi)

Presisi merupakan ukuran yang menunjukkan tingkat keakuratan antara data yang diminta dengan hasil prediksi yang diberikan oleh model. Metrik ini mengukur rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif.

$$Precision = \frac{TP}{TP + FP}$$

3. *Recall*

Recall atau sensitivitas mengukur kemampuan model dalam mengidentifikasi kembali seluruh anggota dari kelas positif. Metrik ini merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan populasi data yang secara aktual bernilai positif.

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

4. *F1-Score*

F1-Score merupakan rata-rata harmonik yang menggabungkan metrik *Precision* dan *Recall* ke dalam satu nilai tunggal. Metrik ini digunakan untuk mencari keseimbangan performa algoritma, terutama apabila terdapat ketidakseimbangan proporsi jumlah data antar kelas.

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

2.7 Figma

Figma merupakan aplikasi desain antarmuka berbasis web yang memungkinkan pengguna untuk merancang *User Interface* (UI) dan *User Experience* (UX) secara interaktif tanpa memerlukan instalasi perangkat lunak yang berat. Karena sifatnya yang berbasis *cloud*, alat ini menjadi solusi praktis bagi desainer dan pengembang dalam berbagai proyek sistem informasi karena dapat diakses dengan mudah melalui peramban web tanpa spesifikasi perangkat yang tinggi [29]. Selain itu, Karimah et al. [30] menambahkan bahwa Figma menyediakan berbagai fitur desain yang lengkap namun tetap mudah digunakan (*user-friendly*), sehingga sangat efektif untuk memfasilitasi pembuatan tata letak aplikasi yang menarik dan responsif.



Gambar 2.5 Figma

Salah satu keunggulan utama yang ditawarkan oleh platform ini adalah fitur kolaborasi tim secara waktu nyata (*real-time*). Fitur ini memungkinkan tim desain untuk bekerja bersama dalam satu ruang kerja digital, berbagi ide, dan melakukan revisi secara langsung, yang sangat krusial untuk mempercepat alur kerja pengembangan [29]. Dalam konteks perancangan sistem, desainer dapat memanfaatkan Figma untuk mensimulasikan interaksi antarmuka dan alur pengguna (*user flow*) sebelum masuk ke tahap pengodean, memastikan bahwa desain yang dihasilkan benar-benar sesuai dengan kebutuhan pengguna [31].

Penggunaan Figma juga berperan penting dalam memberikan gambaran visual yang jelas mengenai sistem yang akan dibangun melalui pembuatan prototipe mendetail. Putri et al. [31] menjelaskan bahwa dengan visualisasi yang akurat, para pemangku kepentingan dapat memberikan umpan balik lebih awal terhadap rancangan sistem. Hal ini pada akhirnya berkontribusi pada penciptaan produk akhir yang lebih efektif, intuitif, dan sesuai dengan standar industri desain modern [30].

2.8 Artificial Intelligence (AI)

Artificial Intelligence (AI) atau Kecerdasan Buatan merupakan cabang keilmuan komputer komprehensif yang berfokus pada perancangan sistem atau mesin agar memiliki kemampuan untuk mensimulasikan kecerdasan kognitif manusia. Berbeda dengan program komputer tradisional yang beroperasi berdasarkan rangkaian instruksi statis (*rule-based systems*), sistem AI dibekali kemampuan untuk mengekstraksi dan memproses kumpulan data (*datasets*) berskala masif, mengidentifikasi pola-pola yang rumit, serta mengambil keputusan secara otomatis untuk memecahkan masalah di lingkungan yang dinamis [32].

Dalam perkembangannya, kecerdasan buatan memiliki beberapa sub-disiplin utama yang menjadi motor penggerak inovasi, di mana yang paling menonjol adalah *Machine Learning* (ML) dan *Deep Learning* (DL). *Machine Learning* merupakan pendekatan yang memungkinkan sistem untuk mengidentifikasi pola aktivitas manusia secara matematis dari himpunan data tanpa memerlukan instruksi pemrograman eksplisit. Sebagai evolusi lebih lanjut, *Deep Learning* menggunakan arsitektur jaringan saraf tiruan berlapis untuk menangani data yang jauh lebih masif dan tidak terstruktur. Keunggulan utama dari teknologi ini adalah kemampuannya dalam mempelajari dan mengekstraksi fitur secara otomatis langsung dari data mentah seperti deretan bingkai video (*frames*) tanpa memerlukan rekayasa fitur secara manual, sehingga menghasilkan tingkat akurasi yang sangat tinggi [33].

Implementasi yang paling relevan dari AI pada bidang analisis gerakan adalah teknologi *Computer Vision*, yaitu ilmu yang memberikan kemampuan pada mesin untuk melihat dan menginterpretasikan informasi visual secara seketika. Pada ranah biomekanika olahraga dan kebugaran, penerapan AI berbasis *Computer Vision* mampu melakukan penangkapan gerak tanpa penanda (*markerless motion capture*) untuk mengevaluasi teknik eksekusi gerakan pengguna [34]. Algoritma ini berperan krusial dalam memberikan umpan balik koreksi postur yang objektif. Evaluasi teknis yang cerdas ini mendorong pergeseran paradigma dari penanganan cedera reaktif menjadi pencegahan cedera proaktif, serta membebaskan pengguna dari keharusan memakai alat sensor fisik tambahan di tubuh (*wearable devices*) saat berlatih [34].



Gambar 2.6 Komponen Utama AI

2.8.1 Machine Learning (Pembelajaran Mesin)

Machine Learning (Pembelajaran Mesin) merupakan sub-bidang utama dari *Artificial Intelligence* (AI) yang berfokus pada perancangan algoritma dan model statistik agar sistem komputer mampu belajar secara mandiri. Berbeda dengan pendekatan pemrograman konvensional yang mengandalkan aturan statis atau logika kondisional yang disusun secara manual, model *machine learning* bekerja dengan mengekstraksi dan mengidentifikasi pola-pola tersembunyi (*hidden patterns*) dari himpunan data historis [35]. Proses komputasi adaptif ini memberdayakan mesin untuk membangun generalisasi pengetahuan, sehingga sistem dapat membuat keputusan, prediksi, atau mengklasifikasikan data baru yang belum pernah diobservasi sebelumnya tanpa harus diprogram secara eksplisit untuk setiap kemungkinan skenario [36].

Dalam arsitektur pembelajarannya, *machine learning* umumnya dikategorikan ke dalam beberapa paradigma utama, di mana *Supervised Learning* (Pembelajaran Terarah)

menjadi kerangka kerja yang paling esensial untuk tugas klasifikasi. Pada paradigma ini, model dilatih menggunakan kumpulan data yang telah dilengkapi dengan label target yang jelas atau nilai kebenaran dasar (*ground truth*) [35]. Selama proses pelatihan, algoritma secara iteratif menyesuaikan parameter atau bobot internalnya untuk meminimalkan fungsi kerugian yaitu selisih tingkat kesalahan antara hasil prediksinya dengan label aktual. Kemampuan optimasi matematis ini membuat algoritma *supervised learning* sangat andal dalam membedakan dan memetakan berbagai kelas data dengan tingkat akurasi yang amat tinggi [37].

2.8.2 Deep Learning (Pembelajaran Mendalam)

Deep Learning (Pembelajaran Mendalam) merupakan sub-disiplin komputasi tingkat lanjut dari *machine learning* yang digerakkan oleh arsitektur Jaringan Saraf Tiruan berlapis (*multilayered neural networks*). Perbedaan mendasar antara *deep learning* dan pembelajaran mesin tradisional terletak pada kedalaman (*depth*) arsitekturnya yang tersusun atas berbagai lapisan tersembunyi (*hidden layers*) [38]. Keberadaan lapisan-lapisan pemrosesan internal ini memungkinkan sistem untuk mentransformasikan dan memetakan fitur input dari data mentah menjadi representasi bobot jaringan (*network weights*) secara hierarkis. Melalui metode pelatihan per lapisan, model *deep learning* mampu mengekstraksi dan menguraikan pola-pola matematis yang sangat rumit dan abstrak dari sebuah dataset, menjadikannya fondasi kemajuan utama dalam kecerdasan buatan modern [38].

Keunggulan paling revolusioner dari *deep learning* dibandingkan metode konvensional adalah kemampuannya dalam melakukan ekstraksi representasi fitur secara otomatis (*automated feature extraction*). Pada alur kerja sistem komputasi visual tradisional, performa model sangat bergantung pada kualitas rekayasa fitur manual (*manual feature engineering*) yang membutuhkan campur tangan dan pengetahuan domain dari pakar manusia [38]. Namun, dalam ekosistem *deep learning*, tahapan rekayasa fitur tersebut diotomatisasi secara penuh. Model mampu mempelajari struktur visual, menyaring kebisingan data (*noise*), dan mengekstraksi informasi spasial maupun temporal yang krusial secara mandiri langsung dari data mentah, sehingga secara signifikan meningkatkan efisiensi dan kemampuan generalisasi model [38].

Penerapan *deep learning* telah membawa kemajuan yang luar biasa dalam berbagai aplikasi visi komputer (*computer vision*), khususnya pada tugas Pengenalan Aktivitas Manusia (*Human Activity Recognition* atau HAR). Arsitektur *deep learning* tingkat lanjut, terutama *Convolutional Neural Networks* (CNN), memiliki kemampuan komputasional

yang sangat tangguh untuk mengekstraksi dan menganalisis fitur langsung dari deretan bingkai video secara otomatis [39]. Kemampuan analitis ini menjadikannya solusi mutakhir dalam sistem pemantauan gerakan yang kompleks, pelacakan koordinat sendi untuk estimasi postur, serta analisis aktivitas yang membutuhkan akurasi tinggi tanpa bergantung pada perangkat tambahan di tubuh pengguna [39].

2.8.3 Computer Vision (Visi Komputer)

Computer Vision (Visi Komputer) merupakan cabang interdisipliner dari kecerdasan buatan yang berfokus pada pengembangan teknik dan teori agar sistem komputer dapat mengekstraksi, menganalisis, dan menginterpretasikan informasi yang bermakna dari data visual. Tujuan fundamental dari visi komputer adalah untuk mereplikasi kemampuan sistem visual manusia melalui algoritma perangkat keras dan perangkat lunak, sehingga memungkinkan mesin untuk melihat, menguraikan piksel mentah, dan merespons lingkungan fisik di sekitarnya [40]. Proses ini melibatkan alur kerja yang kompleks, di mana gambar tunggal maupun rangkaian video ditransformasikan menjadi deskripsi tingkat tinggi atau keputusan kategorikal yang dapat diproses lebih lanjut oleh mesin secara berkecepatan tinggi [40].

Dalam implementasi modernnya, keandalan sistem visi komputer sangat didorong oleh integrasi model pembelajaran mendalam (*deep learning*) dan arsitektur pemrosesan di perangkat ujung (*edge computing*). Pendekatan ini memungkinkan sistem untuk memproses representasi fitur spasio-temporal secara lokal langsung pada perangkat tanpa harus selalu bergantung pada *cloud server* [41]. Melalui normalisasi postur dan ekstraksi fitur yang cerdas, algoritma visi komputer kontemporer dirancang untuk memiliki ketahanan (*robustness*) yang tinggi terhadap berbagai variabilitas visual di dunia nyata, termasuk perubahan pencahayaan, perbedaan sudut pandang kamera, maupun oklusi atau penutupan objek [41].

Dalam lingkup Pengenalan Aktivitas Manusia (*Human Activity Recognition* atau HAR), visi komputer berfungsi sebagai instrumen utama untuk mengidentifikasi pola perilaku melalui teknik pengamatan optik. Implementasi ini mencakup penggunaan algoritma estimasi postur untuk mengekstraksi koordinat titik persendian (*joint keypoints*) yang kemudian dipetakan menjadi representasi kerangka secara digital [42]. Secara teoritis, pendekatan ini mengutamakan sifat pengamatan yang non-invasif dan tanpa kontak fisik. Dengan mentransformasikan informasi visual menjadi fitur-fitur gerakan yang terukur, visi komputer mampu menyediakan data evaluasi yang objektif tanpa memerlukan

ketergantungan pada perangkat sensor fisik yang melekat pada tubuh, sehingga mengoptimalkan fleksibilitas sistem dalam melakukan analisis aktivitas di berbagai lingkungan secara mandiri [42].

2.9 Human Pose Estimation

Human Pose Estimation (HPE) didefinisikan sebagai tugas dalam visi komputer (*computer vision*) yang bertujuan untuk memprediksi lokasi sendi-sendi artikulasi tubuh manusia (*articulated joint locations*) dari sebuah gambar atau video digital [43]. Teknologi ini bekerja dengan mendeteksi dan melacak titik-titik anatomis penting (*keypoints*) pada tubuh, seperti kepala, bahu, siku, pergelangan tangan, pinggul, lutut, dan pergelangan kaki, tanpa memerlukan penanda fisik (*marker*) atau peralatan laboratorium khusus [44], [45].



Gambar 2.7 Human Pose Estimation

Secara fundamental, sistem HPE memproses data visual masukan untuk menghasilkan representasi digital dari postur tubuh manusia. Keluaran (*output*) dari proses ini umumnya berupa sekumpulan koordinat (x, y) untuk 2D atau (x, y, z) untuk 3D yang merepresentasikan lokasi sendi, yang kemudian dihubungkan untuk membentuk kerangka (*skeleton*) digital pengguna [46], [47]. Representasi kerangka ini memberikan informasi geometris mengenai konfigurasi tubuh subjek dalam ruang, yang menjadi dasar untuk berbagai analisis gerak selanjutnya [6].

Perkembangan teknologi HPE saat ini didorong oleh kemajuan pesat dalam metode *Deep Learning* dan *Convolutional Neural Networks* (CNNs), yang menggantikan metode tradisional berbasis fitur manual [43]. Algoritma modern dirancang untuk mampu mengenali pola struktur tubuh manusia secara otomatis dari data visual RGB standar [45]. Pendekatan ini memungkinkan ekstraksi fitur tubuh dilakukan secara *end-to-end*, di mana sistem langsung memetakan piksel gambar menjadi koordinat sendi dan parameter bentuk tubuh (*mesh vertices*) [46].

Dalam konteks implementasi pada perangkat dengan sumber daya terbatas, telah tersedia berbagai model estimasi pose yang ringan dan efisien. Model-model seperti MediaPipe, BlazePose, dan MoveNet dirancang khusus untuk berjalan secara *real-time* pada perangkat *mobile* atau sistem *embedded* [6], [47]. Model-model ini menawarkan keseimbangan antara kecepatan inferensi dan akurasi deteksi, memungkinkan aplikasi pada *smartphone* untuk melakukan pelacakan gerak tubuh manusia hanya dengan menggunakan *Central Processing Unit* (CPU) atau *Graphics Processing Unit* (GPU) bawaan perangkat [47].

2.10 Google ML Kit

Google ML Kit adalah sebuah *Software Development Kit* (SDK) lintas platform yang dikembangkan secara resmi oleh Google untuk menghadirkan keandalan teknologi *machine learning* (pembelajaran mesin) ke dalam ekosistem perangkat seluler Android maupun iOS. Secara arsitektural, antarmuka pemrograman aplikasi (API) pada ML Kit mengabstraksikan kompleksitas model *deep learning* tingkat lanjut menjadi bentuk integrasi fungsional yang siap pakai [48]. Kerangka kerja ini memungkinkan sistem untuk mengimplementasikan solusi kecerdasan buatan komprehensif tanpa mengharuskan pengembangan model algoritma matematis secara manual dari nol, sehingga mempercepat proses transisi dari ekstraksi data mentah menuju pengambilan keputusan komputasional pada perangkat seluler [48], [49].



Gambar 2.8 Ilustrasi Google ML Kit

Keunggulan teoretis utama dari ekosistem ML Kit terletak pada kapabilitas komputasinya yang berpusat pada paradigma pemrosesan di perangkat (*on-device machine learning*). Berbeda dengan arsitektur komputasi awan yang bergantung pada integrasi peladen eksternal (seperti *Firebase*) dan rentan terhadap latensi jaringan, sebagian besar eksekusi algoritma ML Kit dirancang untuk berjalan sepenuhnya secara lokal dan mandiri

di dalam unit pemrosesan perangkat seluler [50]. Di dalam pipa pemrosesan lokal inilah ekstraksi koordinat landmarks tubuh dikonversi menjadi data sudut sendi. Secara biomekanika komputasional, penentuan sudut sendi (θ) yang dibentuk oleh tiga titik koordinat berurutan misalnya titik atas $A(x_A, y_A)$, titik sendi pusat $B(x_B, y_B)$, dan titik bawah $C(x_C, y_C)$ dihitung melalui pendekatan representasi vektor menggunakan fungsi arkus tangen dua bilangan (atan2) [51]. Formulasi matematis untuk mengekstraksi nilai sudut dari titik-titik koordinat spasial tersebut secara presisi didefinisikan sebagai berikut [52]:

$$\theta = |\text{atan2}(y_C - y_B, x_C - x_B) - \text{atan2}(y_A - y_B, x_A - x_B)| \times \frac{180}{\pi}$$

Keterangan Rumus:

- θ (Theta): Hasil besaran sudut antar sendi (dalam satuan derajat).
- atan2 : Fungsi trigonometri arc-tangent 2 dimensi yang digunakan untuk mencari sudut kemiringan vektor garis berdasarkan koordinat kartesius.
- x_1, y_1 : Titik koordinat posisi vertikal dan horizontal pada layar dari sendi ujung pertama (contoh: titik sendi bahu).
- x_2, y_2 : Titik koordinat dari sendi poros atau pusat sudut (contoh: titik sendi siku).
- x_3, y_3 : Titik koordinat dari sendi ujung kedua (contoh: titik sendi pergelangan tangan).
- $| \dots |$: Tanda nilai mutlak (absolute value) yang berfungsi memastikan hasil perhitungan sudut selalu bernilai positif.
- $(180 / \pi)$: Konstanta konversi untuk mengubah hasil perhitungan dari satuan Radian menjadi satuan Derajat (0 derajat hingga 180 derajat).

Secara teoritis komputasional, ML Kit beroperasi sebagai kerangka kerja abstraksi (*framework abstraction*) yang membungkus kompleksitas model dasar agar dapat terintegrasi secara mulus dengan arsitektur perangkat seluler yang memiliki keterbatasan sumber daya. Pipa pemrosesan datanya (*data pipelines*) dikonfigurasi untuk secara otomatis menangani prapemrosesan input visual seperti ekstraksi titik koordinat spasial secara seketika tanpa mengharuskan pengembang memiliki keahlian rekayasa kecerdasan buatan yang mendalam [49], [50]. Dengan memanfaatkan teknik optimasi algoritma tingkat lanjut, ML Kit mampu menjaga keseimbangan antara penggunaan daya komputasi yang rendah dengan tingkat presisi prediksi yang tinggi. Kemampuan pengolahan yang efisien ini menjadikannya standar teknologi yang tangguh untuk menjalankan berbagai tugas analisis

kompleks, mulai dari pengenalan karakter hingga pelacakan biomekanika secara *real-time* di lingkungan dunia nyata [49].

2.11 Latihan Beban Tubuh

Latihan beban tubuh (*bodyweight training*) didefinisikan sebagai metode latihan fisik yang memanfaatkan berat badan individu itu sendiri sebagai beban utama untuk memberikan tahanan, tanpa bergantung pada peralatan tambahan atau mesin kebugaran. Metode ini menawarkan kepraktisan yang tinggi karena dapat dilakukan di mana saja dan kapan saja, menjadikannya solusi yang sangat efisien dari segi waktu dan biaya bagi individu yang memiliki kesibukan padat atau akses terbatas ke fasilitas olahraga [3], [53].

Gerakan-gerakan dalam latihan beban tubuh umumnya bersifat alamiah dan melibatkan berbagai kelompok otot secara bersamaan (*multi-joint*). Contoh gerakan yang sering diterapkan meliputi *push-up*, *sit-up*, *squat*, *lunges*, dan *dips* yang dirancang untuk melatih kekuatan tubuh bagian atas, bawah, serta otot inti [54]. Karena tidak terikat pada jalur gerak mesin yang kaku, latihan ini memungkinkan tubuh bergerak lebih bebas sesuai dengan fungsinya, yang juga dapat membantu meminimalkan risiko cedera saat berolahraga.

Dari segi manfaat kesehatan, latihan beban tubuh terbukti efektif sebagai sarana untuk mengelola berat badan dan meningkatkan kebugaran fisik secara keseluruhan. Intervensi latihan ini secara rutin diketahui dapat membantu menurunkan berat badan secara signifikan pada individu dengan masalah obesitas [54], [55]. Selain itu, penelitian terbaru menunjukkan bahwa latihan beban tubuh juga berperan penting dalam meningkatkan kapasitas fungsional, seperti daya tahan otot dan fleksibilitas, serta dapat memperkuat sistem imun tubuh [53].

2.11.1 *Pushup*

Pushup merupakan salah satu bentuk latihan resistensi beban tubuh (*bodyweight resistance training*) fundamental yang mengeksekusi gerakan dorongan dari posisi telungkup menggunakan kekuatan ekstremitas atas. Dalam disiplin ilmu biomekanika, latihan ini diklasifikasikan sebagai gerakan rantai kinetik tertutup (*closed kinetic chain exercise*), di mana bagian distal tubuh (telapak tangan dan ujung kaki) tetap diam atau terfiksasi pada permukaan penyangga, sementara bagian proksimal tubuh bergerak melintasi ruang [56]. Sifat gerakan ini menuntut koordinasi antar-segmen tubuh yang tinggi, menjadikannya standar fungsional yang sangat efektif untuk melatih kekuatan dan daya tahan otot tubuh bagian atas tanpa bergantung pada ketersediaan peralatan beban eksternal [56].

Secara kinesiology, eksekusi *pushup* melibatkan aktivasi sinergis dari berbagai kelompok otot utama. Otot *pectoralis major* dan *triceps brachii* bertindak sebagai penggerak utama, didukung oleh kelompok otot *deltoid* serta otot stabilisator inti tubuh. Secara teoritis, modifikasi pada jarak atau posisi penempatan tangan akan memberikan dampak langsung terhadap persentase rekrutmen serat otot [57]. Sebagai contoh, jarak tangan yang lebih sempit (*diamond push-up*) akan secara signifikan meningkatkan beban kerja pada *triceps brachii*, sementara posisi yang lebih lebar akan menggeser tegangan mekanis secara dominan menuju *pectoralis major* [56], [57]. Pemahaman mendalam terhadap distribusi aktivasi neuromuskular ini menjadi esensial untuk memaksimalkan efektivitas gerakan.

Pemenuhan standar postur yang presisi tidak hanya penting untuk optimasi otot, melainkan juga bertindak sebagai parameter utama dalam pencegahan cedera. Secara spesifik, eksekusi *pushup* sangat bergantung pada stabilitas kinematika skapula (tulang belikat) [58]. Penyimpangan postur seperti panggul yang merosot, tulang belakang yang tidak netral, atau rotasi bahu yang tidak stabil dapat menyebabkan pergeseran kinematik yang membebani jaringan lunak. Jika postur tidak dikoreksi, pola gerakan yang asimetris ini secara kumulatif berpotensi memicu sindrom nyeri bahu kronis (*chronic shoulder pain*) [58]. Oleh karena itu, pemeliharaan garis keselarasan tubuh dan rentang gerak yang konsisten menjadi kriteria teknis mutlak dalam evaluasi latihan ini.

2.11.2 Squat

Squat merupakan latihan fungsional fundamental yang melibatkan gerakan multi-sendi kompleks pada ekstremitas bawah. Dalam disiplin biomekanika, *squat* diklasifikasikan sebagai gerakan rantai kinetik tertutup yang menuntut koordinasi simultan antara persendian pergelangan kaki, lutut, panggul, hingga batang tubuh [59]. Sifat mekanis yang terintegrasi ini menjadikannya salah satu instrumen evaluasi yang paling komprehensif untuk mengukur kapasitas daya ledak, stabilitas postur dinamis, serta efisiensi transfer gaya mekanis dari tubuh bagian bawah menuju pusat tubuh [59].

Secara kinesiology, eksekusi *squat* bergantung pada aktivasi sinergis dari berbagai kelompok otot besar. Latihan ini secara efektif merekrut otot *quadriceps*, *gluteus maximus*, dan *hamstrings* sebagai penggerak utama, yang dibantu oleh aktivasi otot-otot inti untuk menjaga stabilitas rongga panggul [59]. Secara teoretis, kedalaman rentang gerak (*range of motion*) yang dicapai oleh pengguna, seperti perbedaan antara *half squat* (jongkok separuh) dan *full squat* (jongkok penuh), memberikan pengaruh yang signifikan terhadap aspek biomekanis dan fisiologis. Variasi kedalaman ini secara langsung memodulasi rekrutmen unit motorik, sinkronisasi serat otot, serta tingkat stabilitas sendi dinamis [60]. Literatur

menunjukkan bahwa *squat* dengan rentang gerak yang lebih dalam tidak hanya menuntut fungsi neuromuskular yang lebih masif, tetapi juga mengoptimalkan pengembangan gaya (*force development*) secara keseluruhan jika dibandingkan dengan eksekusi pada rentang gerak yang lebih pendek [60].

Kualitas dan keamanan biomekanis dari gerakan *squat* sangat ditentukan oleh pemeliharaan keselarasan anatomi tubuh. Parameter kinematika yang esensial mencakup penjagaan tulang belakang yang netral serta stabilitas bidang frontal pada lutut [61]. Kegagalan dalam mengontrol keseimbangan postur sering kali memicu deviasi gerakan seperti abduksi atau rotasi internal lutut yang berlebihan. Penyimpangan lintasan gerak ini secara teoretis akan meningkatkan beban geser pada persendian dan ligamen, sehingga berpotensi memicu cedera kronis [61]. Oleh karena itu, analisis kinematika terhadap sudut sendi menjadi standar mutlak untuk memastikan efisiensi gerakan dan memitigasi risiko cedera struktural pada ekstremitas bawah.

2.11.3 *Situp*

Sit-up merupakan salah satu bentuk latihan beban tubuh (*bodyweight exercise*) klasik yang dirancang untuk meningkatkan kekuatan dan daya tahan otot pada area pusat tubuh (*core muscular endurance*). Secara teoretis, latihan ini mengeksekusi gerakan fleksi tulang belakang dan pinggul untuk memindahkan batang tubuh dari posisi telentang menuju posisi duduk [62]. Dalam disiplin evaluasi kebugaran fungsional, *sit-up* sering dijadikan instrumen uji standar untuk mengukur kapasitas daya tahan otot perut secara objektif. Kajian literatur membuktikan bahwa pelaksanaan tes *sit-up* dengan durasi yang dikontrol (misalnya 60 detik) memiliki tingkat validitas konstruk yang sangat tinggi untuk mengevaluasi efisiensi sistem oksidatif otot batang tubuh, yang krusial untuk menstabilkan postur selama aktivitas fisik intens [62].

Secara kinesiologi, eksekusi gerakan *sit-up* memicu aktivasi sinergis dari berbagai kelompok otot di area abdomen dan panggul. Otot *rectus abdominis* bertindak sebagai penggerak utama selama fase awal fleksi batang tubuh, sementara kelompok otot sekunder seperti *transversus abdominis* bekerja untuk menjaga stabilitas tekanan internal [63]. Menjelang akhir rentang gerak (*range of motion*) di mana torso diangkat hingga membentuk sudut 90 derajat terhadap lantai, otot-otot fleksor pinggul (*hip flexors*) mulai mengambil alih sebagian besar beban mekanis. Secara biomekanis, pencapaian sudut dan rentang gerak yang tepat selama eksekusi *sit-up* sangat menentukan persentase rekrutmen serat otot perut secara

maksimal, di mana teknik yang tidak terkontrol hanya akan mengalihkan tegangan menuju kompartemen panggul [63].

Kualitas dan keamanan mekanis dari *sit-up* sangat ditentukan oleh pemeliharaan kinematika sendi yang presisi. Eksekusi gerakan yang tidak tepat seperti tarikan paksa pada leher bagian belakang, lengkung punggung bawah yang ekstrem, atau hentakan momentum tubuh yang berlebihan secara teoretis dapat mendistribusikan gaya kompresi abnormal pada struktur tulang belakang [64]. Untuk mengatasi bias evaluasi manual dan mencegah risiko cedera akibat postur yang salah, studi biomekanika olahraga modern mulai menggunakan instrumen validasi berbasis sensor optik (seperti kamera pelacak kerangka gerak) untuk mengevaluasi kualitas repetisi *sit-up* secara seketika. Pengawasan objektif terhadap sudut sendi dan lintasan tubuh ini menjadi parameter mutlak untuk memastikan bahwa kontraksi otot perut terjadi secara efisien tanpa mengorbankan integritas struktural pengguna [64].

2.11.4 *High Knees*

High knees (angkat lutut tinggi) merupakan salah satu bentuk latihan pliometrik dan dril pemanasan dinamis yang fundamental dalam biomekanika olahraga. Gerakan ini dilakukan secara bergantian pada kedua tungkai dengan penekanan pada pengangkatan lutut hingga sejajar dengan panggul, yang dikombinasikan dengan tempo pergantian kaki yang cepat. Secara teoretis, latihan peregangan dinamis seperti ini berfungsi sebagai stimulus neuromuskular yang krusial. Kajian literatur terbaru membuktikan bahwa dril pemanasan dinamis memberikan efek akut yang jauh lebih superior dalam meningkatkan performa lari cepat (*sprint*) dan daya ledak otot ekstremitas bawah dibandingkan dengan peregangan statis atau PNF (*Proprioceptive Neuromuscular Facilitation*) [65]. Integrasi *high knees* dalam protokol latihan terbukti dapat mengoptimalkan kinematika langkah dan mempersiapkan efisiensi mekanis otot secara instan sebelum aktivitas fisik intensitas tinggi [65].

Secara kinesiologi, eksekusi *high knees* membutuhkan aktivasi otot yang terkoordinasi secara cepat. Otot fleksor pinggul (*hip flexors*), yang utamanya terdiri dari *iliopsoas* dan *rectus femoris*, bertindak sebagai penggerak utama (*prime movers*) untuk menghasilkan gaya angkat paha secara vertikal [66]. Sementara itu, pada fase penyangga, otot betis dan pergelangan kaki bekerja secara dinamis untuk mengelola energi kinetik dan menghasilkan daya pantul. Untuk memfasilitasi gerakan ekstremitas yang asimetris dan cepat ini, kelompok otot inti harus berkontraksi secara isometrik guna menjaga stabilitas gelang panggul agar tetap berada dalam posisi netral dan mencegah terjadinya rotasi tubuh yang tidak diinginkan [65], [66].

Kualitas dan efisiensi biomekanis dari *high knees* sangat bergantung pada pemeliharaan keselarasan postur tubuh (*postural alignment*). Secara kinematika, postur yang ideal mengharuskan batang tubuh (*trunk*) tetap dalam posisi tegak lurus yang netral, dengan sudut fleksi pinggul mencapai 90 derajat pada titik puncak gerakan [67]. Kompensasi mekanis yang sering terjadi, seperti kecenderungan tubuh untuk condong ke belakang atau dorongan lutut yang tidak maksimal, secara teoretis akan memindahkan proyeksi pusat massa menjauhi titik tumpu. Pergeseran ini tidak hanya menurunkan efisiensi transfer gaya ke permukaan tanah, tetapi juga mengurangi tingkat aktivasi pada otot target, yang berpotensi memicu ketidakseimbangan mekanis pada struktur tulang belakang [66], [67].

2.11.5 *Lunges*

Lunges (langkah terjang) merupakan salah satu bentuk latihan beban tubuh (*bodyweight exercise*) fungsional yang berfokus pada pergerakan ekstremitas bawah secara unilateral (satu sisi). Dalam disiplin kinesiologi, *lunges* diklasifikasikan sebagai gerakan rantai kinetik tertutup parsial yang menuntut perpindahan dinamis dari pusat massa tubuh ke atas dasar tumpuan yang menyempit [68]. Berbeda dengan latihan *squat* yang bersandar pada stabilitas tumpuan bilateral, sifat asimetris dari eksekusi *lunges* menciptakan tantangan biomekanis yang jauh lebih tinggi terhadap sistem kontrol postur, keseimbangan dinamis, serta sistem proprioepsi tubuh secara keseluruhan [68].

Secara kinesiologi, eksekusi gerakan *lunges* merekrut aktivasi sinergis dari berbagai kelompok otot besar di kompartemen paha dan panggul. Otot *quadriceps femoris* dan *gluteus maximus* bertindak sebagai penggerak utama untuk mengelola fase pengereman eksentrik saat tubuh diturunkan dan fase dorongan konsentrik saat kembali ke posisi awal [69], [70]. Karena beban tubuh ditopang secara unilateral, *lunges* juga secara mekanis memaksa rekrutmen masif pada otot-otot stabilisator sekunder seperti *gluteus medius*, *hamstrings*, dan otot inti guna mencegah panggul miring dan menjaga keseimbangan tubuh pada bidang frontal selama perpindahan beban berlangsung [69].

Kualitas dan keamanan mekanis dari eksekusi *lunges* sangat bergantung pada stabilitas sendi lutut dan aktivasi otot yang terkoordinasi secara sinergis. Secara kinematika, eksekusi *lunges* yang ideal mengharuskan batang tubuh tetap tegak dan penjagaan keselarasan lutut depan agar bergerak seimbang menopang gaya reaksi tanah. Kajian biomekanika dan elektromiografi terbaru menunjukkan bahwa stabilitas *forward lunge* sangat dipengaruhi oleh pola rekrutmen antara otot ekstremitas bawah dan otot inti selama fase tumpuan [70]. Kurangnya efisiensi dalam koordinasi neuromuskular atau kegagalan

dalam mempertahankan postur tubuh terbukti secara mekanis dapat memicu pembebanan berlebih (*overloading*) pada kompartemen sendi lutut dan pergelangan kaki [70]. Oleh karena itu, pengawasan presisi terhadap keselarasan anatomi tubuh dan optimalisasi sinergi otot sangat krusial untuk meningkatkan efisiensi performa gerakan serta memitigasi risiko cedera struktural akibat teknik yang kurang tepat [69], [70].

2.11.6 *Dips*

Dips merupakan bentuk latihan beban tubuh (*bodyweight exercise*) kompleks yang menitikberatkan pada pergerakan fungsional ekstremitas atas. Dalam disiplin biomekanika olahraga, gerakan ini diklasifikasikan sebagai latihan rantai kinetik tertutup (*closed kinetic chain*) di mana bagian utama tubuh diturunkan dan diangkat melintasi ruang, sementara kedua titik tumpu (telapak tangan) tetap statis pada penopang sejajar. Latihan ini menuntut subjek untuk menyokong dan mengendalikan sepenuhnya massa tubuh mereka secara mandiri melawan gravitasi, menjadikannya salah satu instrumen evaluasi paling komprehensif untuk mengukur daya ledak otot, kekuatan mendorong, dan stabilitas neuromuskular pada kompartemen atas tubuh [71].

Fase eksekusi *dips* memicu rekrutmen sinergis yang sangat tinggi pada beberapa kelompok otot utama secara simultan. Otot *triceps brachii*, *pectoralis major*, dan *anterior deltoid* bertindak sebagai penggerak utama selama siklus pengereman eksentrik (fase turun) dan dorongan konsentrik (fase naik). Secara teoretis, persentase dominasi rekrutmen serat otot ini secara langsung dapat dimodulasi oleh derajat inklinasi batang tubuh postur yang cenderung tegak lurus akan memfokuskan isolasi tegangan secara absolut pada otot *triceps*, sedangkan postur dengan kemiringan tubuh ke depan mendistribusikan beban gaya secara lebih dominan pada jaringan otot dada. Analisis elektromiografi (EMG) mutakhir membuktikan bahwa variasi *bar dips* menghasilkan puncak aktivasi otot yang jauh lebih superior dibandingkan variasi stabil lainnya, dan seiring terjadinya kelelahan fisiologis, sistem saraf pusat secara kompensatori akan meningkatkan intensitas rekrutmen motorik untuk tetap mempertahankan lintasan gerakan tubuh [71], [72].

Dari perspektif kinematika persendian dan keselamatan struktural, eksekusi *dips* yang presisi sangat bergantung pada kontrol rentang gerak (*range of motion*). Gerakan eksentrik pada *dips* secara inheren menempatkan sendi bahu pada posisi ekstensi mekanis yang sangat dalam [71]. Secara teoretis, jika subjek melakukan fleksi siku jauh melampaui batas rentang anatomis yang wajar (turun terlalu dalam melebihi sudut 90 derajat), kompartemen anterior bahu akan mengalami lonjakan tegangan geser yang ekstrem, berisiko

memicu peradangan kapsul sendi maupun sindrom pelampiasan bahu [72]. Oleh karena itu, standardisasi terhadap keselarasan fleksi siku, pemeliharaan depresi tulang belikat, serta pengawasan durasi siklus gerakan yang kini sering dilacak menggunakan validasi sensor kinematika menjadi parameter teknis mutlak untuk mengevaluasi efisiensi mekanis sekaligus memitigasi risiko cedera struktural pada ekstremitas atas [72], [73].

2.11.7 *Glute Bridges*

Glute bridge merupakan latihan fungsional rantai kinetik tertutup (*closed kinetic chain exercise*) yang menargetkan penguatan kompleks otot ekstremitas bawah bagian posterior. Gerakan ini dieksekusi dari posisi telentang telapak kaki menapak lantai, dilanjutkan dengan mengangkat panggul menjauhi bidang tumpu hingga batang tubuh sejajar dengan paha [74]. Dalam tinjauan biomekanika, latihan ini mengeksplorasi gaya reaksi tanah melalui tumpuan kaki untuk melawan gravitasi. Latihan ini diakui secara luas dalam rehabilitasi klinis dan pelatihan kebugaran karena kemampuannya memfasilitasi ekstensi panggul secara terisolasi tanpa memberikan beban kompresi aksial yang berlebihan pada kolom tulang belakang [75].

Secara kinesiologi dan analisis elektromiografi (EMG), *glute bridge* merekrut otot *gluteus maximus* dan *biceps femoris* sebagai penggerak utama, dengan dukungan stabilisasi dari otot *erector spinae* dan otot inti [74], [76]. Kajian literatur membuktikan bahwa modifikasi mekanis pada titik tumpu, seperti penggunaan posisi dorsofleksi pergelangan kaki (bertumpu pada tumit) dibandingkan telapak kaki penuh, secara signifikan meningkatkan rekrutmen serat otot *gluteus maximus*. Selain itu, transisi dari tumpuan bilateral menjadi unilateral terbukti menuntut aktivasi isometrik yang jauh lebih tinggi pada kelompok otot *obliques* internal dan *gluteus medius* guna memitigasi rotasi panggul pada bidang transversal [74].

Dari perspektif kinesiologi tulang belakang dan pencegahan cedera, keamanan mekanis gerakan ini sangat bergantung pada stabilitas dan rasio aktivasi antara otot stabilisator lokal dan otot global. Secara teoretis, ekstensi panggul yang ideal harus dicapai tanpa memicu kompensasi berlebih pada batang tubuh [75]. Kajian menunjukkan bahwa ketika gerakan dilakukan dengan teknik yang kurang stabil, sering terjadi hiperaktivasi abnormal pada kelompok otot punggung *erector spinae* tanpa diimbangi oleh penguncian dari otot perut. Kompensasi mekanis ini secara langsung dapat mentransfer beban tekanan menuju tulang belakang lumbal dan memicu nyeri punggung bawah [76]. Oleh karena itu, pengawasan objektif terhadap keselarasan panggul dan postur punggung yang netral menjadi

parameter mutlak untuk memastikan otot *gluteus* tetap bekerja sebagai penggerak utama tanpa membahayakan integritas struktural lumbal [75], [76].

