

BAB I

PENDAHULUAN

1.1 Latar Belakang

PT Kuantia Prima Indonesia adalah perusahaan yang bergerak di bidang jasa konsultasi dan pengembangan pendidikan. Perusahaan ini menggunakan teknologi sebagai dasar utama dalam memberikan layanan yang mendukung proses pembelajaran, pengajaran, serta manajemen pendidikan agar lebih efektif dan modern. Program magang di PT Kuantia Prima Indonesia telah berjalan sejak tahun 2022. Seleksi peserta magang dilakukan melalui jalur mandiri maupun kerja sama dengan beberapa institusi perguruan tinggi, salah satunya adalah Universitas Negeri Surabaya (UNESA). Selain itu, perusahaan ini juga pernah terlibat dalam program Magang dan Studi Independen Bersertifikat (MSIB). Dalam setiap periode pelaksanaan magang, rata-rata PT Kuantia Prima Indonesia menerima sekitar 20–25 peserta magang, sedangkan jumlah pegawai aktif yang terlibat dalam operasional perusahaan sekitar 13 orang. Peserta magang tersebut kemudian dibagi ke dalam tiga divisi sesuai dengan kebutuhan perusahaan serta minat masing-masing peserta, di mana setiap divisi dipandu oleh seorang mentor yang merupakan pegawai internal perusahaan (Sumber: Wawancara, 2025).

Seiring dengan meningkatnya jumlah peserta magang di PT Kuantia Prima Indonesia, perusahaan menghadapi tantangan dalam pengelolaan dan pemantauan kegiatan magang secara efektif. Proses administrasi dan monitoring peserta magang saat ini masih dilakukan secara terpisah dengan memanfaatkan berbagai media yang berbeda. Beberapa aktivitas seperti pengumpulan tugas, pencatatan kehadiran, dan pelaporan kegiatan harian dilakukan menggunakan *spreadsheet*, *google form*, dan *WhatsApp*. Kondisi ini menyebabkan data magang tersebar di berbagai tempat dan tidak tersimpan dalam satu sistem yang terpusat. Akibatnya, perusahaan mengalami kesulitan dalam memperoleh gambaran perkembangan peserta magang secara menyeluruh. Selain itu, proses yang belum terintegrasi ini menyebabkan berbagai kendala, seperti kesalahan pencatatan, kurangnya efisiensi waktu, dan lambatnya penyusunan laporan presensi [1].

Kondisi tersebut menimbulkan sejumlah permasalahan, di antaranya perlambatan dalam proses administrasi, ketidakseragaman format dokumen, serta rendahnya efektivitas pemantauan kegiatan magang baik dari sisi peserta maupun mentor. Peserta magang diwajibkan mengirimkan tugas dan laporan melalui google form dan spreadsheet, sementara mentor harus melakukan pemeriksaan serta memberikan masukan tanpa adanya sistem pencatatan perkembangan yang terstruktur. Selain itu, tidak adanya sistem yang mampu mengintegrasikan data absensi, tugas, dan *logbook* kegiatan menyebabkan proses pembimbingan menjadi kurang optimal dan sulit untuk dievaluasi secara objektif. Data yang tidak saling terintegrasi harus di *cross-check* secara manual yang sangat memakan waktu [2].

Berbagai penelitian terdahulu menunjukkan bahwa penerapan sistem informasi berbasis web mampu menggantikan proses manual yang tidak efisien dan sulit dipantau. Pengembangan sistem monitoring magang berbasis web terbukti mampu meningkatkan kerapihan pencatatan serta memudahkan proses pengawasan melalui pengelolaan data yang terpusat dan terstruktur [40]. Selain itu, penelitian mengenai pengembangan sistem berbasis metode Scrum menunjukkan bahwa perancangan *backend* yang baik berperan penting dalam mengelola alur proses, integrasi data, serta menjaga konsistensi dan keandalan sistem secara keseluruhan [41]. Meskipun kedua penelitian tersebut berhasil meningkatkan efisiensi pengelolaan sistem, pembahasannya masih bersifat umum dan belum secara khusus menitikberatkan pada perancangan *backend* sebagai komponen utama dalam pengelolaan proses bisnis dan integrasi data pada sistem monitoring magang. Oleh karena itu, diperlukan sistem monitoring magang yang terstruktur dan terintegrasi untuk mendukung proses pemantauan kegiatan magang secara berkelanjutan.

Berdasarkan hasil analisis permasalahan yang telah diidentifikasi, sistem yang dirancang dalam penelitian ini difokuskan untuk menyediakan seperangkat fitur yang mampu mengakomodasi kebutuhan seluruh pihak yang terlibat dalam program magang. Bagi peserta magang, sistem menyediakan akses akun pengguna untuk masuk ke dalam sistem, melakukan pencatatan kehadiran, mengunggah tugas, serta melihat materi dan daftar tugas yang diberikan oleh mentor. Dari sisi mentor, sistem mendukung pengelolaan peserta bimbingan, pengunggahan dan pembaruan tugas, pengelolaan materi pembelajaran, serta pelaksanaan penilaian

akhir sebagai bentuk evaluasi kinerja peserta. Adapun peran admin difungsikan untuk mengelola data peserta magang, data mentor, serta pengaturan akun pengguna guna memastikan seluruh proses berjalan secara terstruktur dan terkendali. Perancangan fitur-fitur tersebut bertujuan agar setiap peran dalam kegiatan magang dapat terlayani secara optimal melalui satu sistem terintegrasi dan terpusat. Selain itu, pengawasan kehadiran yang dilakukan secara manual seringkali tidak memungkinkan untuk melacak kehadiran secara *realtime* [3].

Dalam pelaksanaannya, pengembangan aplikasi magang berbasis web pada penelitian ini membutuhkan fleksibilitas tinggi terhadap perubahan kebutuhan pengguna, karena kebutuhan sistem berkembang secara bertahap seiring adanya masukan dari pengelola magang dan pengguna selama proses pengembangan. Metode pengembangan sistem lain seperti *Waterfall* dan *Rapid Application Development* (RAD) telah dipertimbangkan, di mana *waterfall* sesuai untuk kebutuhan yang stabil dan terdefinisi sejak awal, namun kurang mampu mengakomodasi perubahan kebutuhan yang muncul selama proses pengembangan [4], sedangkan RAD menekankan kecepatan pengembangan melalui pembuatan *prototipe* secara cepat, tetapi menuntut ketersediaan sumber daya yang tinggi serta berpotensi mengurangi kontrol terhadap kualitas dan dokumentasi sistem pada kebutuhan yang berkembang [4]. Berdasarkan kondisi tersebut, diperlukan metode pengembangan yang adaptif dan iteratif. Scrum memungkinkan terjadinya perubahan kebutuhan selama proses pengembangan tanpa mengganggu keseluruhan alur kerja, karena pengembangan dilakukan secara bertahap dan berulang. Melalui pembagian pekerjaan ke dalam *sprint* yang singkat, Scrum menghasilkan produk *increment* yang dapat langsung diuji dan dievaluasi, sehingga apabila pada *increment* awal sistem belum sepenuhnya memenuhi kebutuhan pengguna, perbaikan dan pengembangan dapat dilakukan pada *sprint* berikutnya berdasarkan hasil evaluasi [5]. Dengan evaluasi yang dilakukan secara berkala, risiko kesalahan pada tahap akhir pengembangan dapat diminimalkan, sehingga metode Scrum dipandang tepat untuk mendukung pengembangan aplikasi magang berbasis web secara efektif, adaptif, dan terkontrol.

Oleh karena itu, berdasarkan seluruh permasalahan yang telah dijelaskan pada latar belakang di atas, penelitian ini berfokus untuk merancang dan

membangun sistem monitoring kegiatan magang berbasis web yang terintegrasi dan mendukung proses administrasi, pembimbingan, serta evaluasi peserta secara lebih efektif. Pemantauan umumnya digunakan untuk memeriksa kinerja dan menetapkan tujuan [6]. Sistem ini dikembangkan dengan menerapkan metode Scrum sebagai pendekatan pengembangan perangkat lunak yang iteratif, kolaboratif, dan adaptif terhadap kebutuhan pengguna. Diharapkan sistem ini mampu mengatasi berbagai kendala yang muncul pada proses monitoring magang di PT Kuantas, seperti ketidakterpaduan data, proses manual yang tidak efisien, dan keterbatasan dalam pengawasan kinerja peserta. Dengan demikian, penelitian ini mengangkat judul **“RANCANG BANGUN *BACKEND* SISTEM *MONITORING* KEGIATAN MAGANG BERBASIS WEB MENGGUNAKAN METODE SCRUM (STUDI KASUS PT KUANTA PRIMA INDONESIA)”**.

1.2 Rumusan Masalah

Berdasarkan latar belakang penelitian yang telah dijelaskan, rumusan masalah yang dapat diidentifikasi dalam penelitian ini berfokus pada bagaimana merancang dan membangun *backend* sistem monitoring kegiatan magang berbasis web di PT Kuantas Prima Indonesia dengan menggunakan metode Scrum?

1.3 Batasan Masalah

Agar skripsi ini terarah dan fokus pada tujuan utama, maka batasan masalah dalam penelitian ini ditetapkan sebagai berikut:

1. Skripsi ini berfokus pada perancangan dan pembangunan *backend* sistem monitoring peserta magang, dengan cakupan pada arsitektur sistem, alur bisnis, serta mekanisme pengolahan dan integrasi data. Pengembangan difokuskan untuk menghasilkan sistem yang terstruktur, terpusat, dan mampu mendukung proses administrasi magang secara efisien.
2. Lingkup pengguna pada sistem ini terdiri dari tiga peran utama, yaitu Admin, COO, Mentor, dan Peserta Magang. Admin mengelola data peserta, membuat dan mengatur akun, menentukan divisi penempatan, serta memantau kelengkapan administrasi. COO mengelola kuis kompetensi, dan mengatur hari libur & jam kerja, Mentor memberikan dan mengelola tugas, memantau absensi dan progres peserta, serta memberikan evaluasi dan penilaian kinerja. Peserta magang melakukan absensi, menerima dan

mengumpulkan tugas, mengisi *logbook* harian, serta melihat perkembangan dan catatan evaluasi dari mentor.

3. Skripsi ini menitikberatkan pada penerapan metode Scrum dalam proses pembangunan sistem yang meliputi *sprint planning*, *sprint backlog*, *sprint execution*, *sprint review*, dan *sprint retrospective*, dengan jumlah iterasi *sprint* yang dibatasi hingga 5 *sprint*. Fokus penelitian diarahkan pada proses pengembangan dan pencapaian *increment* sistem pada setiap *sprint*. Evaluasi performa server dan analisis keamanan tingkat lanjut tidak dibahas secara mendalam, selain penerapan mekanisme autentikasi dan otorisasi dasar untuk memastikan keamanan akses pengguna.

1.4 Tujuan Penelitian

Skripsi ini bertujuan untuk merancang & membangun sistem monitoring kegiatan magang berbasis web di PT Kuantu Prima Indonesia menggunakan metode Scrum guna mendukung proses administrasi dan pembimbingan peserta magang.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Menghasilkan *backend* sistem monitoring magang berbasis web yang mampu mengintegrasikan proses bisnis utama, meliputi penyediaan materi pembelajaran, pengelolaan tugas peserta magang, pelaksanaan kuis pelatihan kompetensi, serta penilaian akhir magang, sehingga mendukung pengelolaan administrasi dan evaluasi peserta magang secara lebih terstruktur, efisien, dan terpusat.
2. Memberikan akses bagi Admin, Mentor, dan Peserta Magang dalam memantau dan mengelola aktivitas magang secara terpusat, serta mengurangi ketergantungan pada pengelolaan data menggunakan spreadsheet yang belum terintegrasi, sehingga meminimalkan potensi ketidakkonsistenan dan kesalahan data.
3. Mendukung proses pembimbingan dan evaluasi peserta magang melalui penyajian data yang lebih akurat, tersusun, dan mudah ditelusuri, sehingga meningkatkan akuntabilitas dan kualitas pelaksanaan program magang.

1.6 Sistematika Penulisan

Sistematika penulisan dalam skripsi ini disusun secara terstruktur agar pembahasan dapat tersaji dengan runtut dan mudah dipahami. Adapun sistematika penulisan laporan penelitian ini terdiri atas.

BAB I PENDAHULUAN

Bab ini berisi penjelasan umum mengenai latar belakang penelitian, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, serta sistematika penulisan skripsi secara keseluruhan.

BAB II TINJAUAN PUSTAKA

Bab ini memuat teori-teori dasar, konsep-konsep pendukung, serta hasil penelitian terdahulu yang relevan. Tujuannya adalah memberikan landasan teoritis yang menjadi acuan dalam pelaksanaan penelitian.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metode, tahapan, dan prosedur penelitian yang digunakan. Di dalamnya mencakup pendekatan penelitian, teknik pengumpulan data, analisis sistem, serta tahapan perancangan dan pelaksanaan penelitian.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil penelitian yang telah dilakukan beserta pembahasannya. Pada bagian ini dijelaskan hasil analisis, implementasi, dan evaluasi terhadap sistem atau objek yang diteliti untuk menjawab rumusan masalah.

BAB V PENUTUP

Bab ini berisi kesimpulan yang diperoleh dari hasil penelitian serta saran yang dapat digunakan sebagai masukan atau pengembangan penelitian selanjutnya.

DAFTAR PUSTAKA

Berisi referensi atau sumber pustaka yang digunakan dalam penyusunan skripsi, baik berupa buku, jurnal, maupun sumber daring yang relevan.

LAMPIRAN

Berisi dokumen pendukung penelitian seperti hasil wawancara, data uji coba, tangkapan layar, dan dokumen lain yang berkaitan dengan pelaksanaan penelitian.

BAB II

TINJAUAN PUSTAKA

Bab ini menyajikan kajian terhadap penelitian terdahulu untuk dikomparasikan dengan penelitian yang akan dilakukan, menguraikan landasan teori yang relevan sebagai dasar pengembangan solusi, serta alat dan teknologi yang akan dibutuhkan selama penelitian.

2.1 Dasar Teori

Dasar teori yang dijelaskan pada bab ini merupakan hal-hal yang mendukung dalam skripsi ini:

2.1.1 Profil Perusahaan PT Kuanta Prima Indonesia

PT Kuanta Prima Indonesia merupakan perusahaan yang berfokus pada transformasi pendidikan melalui layanan konsultasi, pelatihan, pengembangan kepemimpinan, riset, serta pendampingan berkelanjutan. Sebagai mitra bagi sekolah dan lembaga pendidikan, Kuanta memastikan setiap program yang ditawarkan telah melalui proses riset yang komprehensif untuk menjamin relevansi dan efektivitasnya. Perusahaan menyediakan layanan *coaching* yang dilakukan secara terstruktur, termasuk melalui platform digital, untuk memastikan penguatan kompetensi dan keberlanjutan hasil program. Dengan komitmen tersebut, Kuanta berupaya memberikan kontribusi nyata dalam meningkatkan kualitas pendidikan dan menciptakan masa depan pembelajaran yang lebih baik di Indonesia.

2.1.2 Sistem Informasi

Sistem informasi merupakan suatu kesatuan komponen yang saling berinteraksi untuk mengumpulkan, mengolah, menyimpan, dan mendistribusikan informasi sehingga dapat mendukung kegiatan operasional maupun pengambilan keputusan dalam sebuah organisasi. Sistem informasi adalah bagian dari suatu organisasi yang berfungsi memenuhi kebutuhan pengolahan transaksi harian, mendukung berbagai operasional, manajerial, serta kegiatan strategis organisasi tersebut. Fungsi utama sistem informasi adalah menyimpan, mengambil, mengubah, mengolah, serta mengkomunikasikan informasi yang diterima, baik melalui sistem informasi maupun perangkat-perangkat sistem lainnya [7].

Saat ini, sistem informasi tidak hanya digunakan untuk mencatat transaksi saja, tetapi juga menjadi dasar dalam proses digitalisasi bisnis perusahaan. Dengan sistem informasi, berbagai kegiatan operasional bisa dilakukan lebih cepat, tepat, dan hemat waktu, serta membantu mengurangi kesalahan yang biasa terjadi ketika menggunakan cara manual.

Dengan sistem informasi yang diterapkan dengan baik, perusahaan bisa meningkatkan kualitas pelayanan, menyempurnakan cara kerja, serta memperkuat kemampuan menganalisis data untuk mendukung pengambilan keputusan yang lebih baik.

2.1.3 Magang

Kegiatan magang menjadi salah satu bentuk pembelajaran yang memberikan kesempatan bagi peserta untuk menerapkan ilmu dan keterampilan yang telah dipelajari selama masa pendidikan di lingkungan kerja sesungguhnya. Magang merupakan kesempatan untuk mengalami proses pembelajaran melalui pengalaman yang telah dipelajari ke dalam lingkungan kerja, dan pembelajaran mandiri selama masa magang sangat penting agar peserta magang dapat mempertahankan kualitas pekerjaan secara berkelanjutan dan progresif [8].

Pelaksanaan magang dirancang untuk mendukung peningkatan pengetahuan, kemampuan teknis, serta *soft skill* peserta magang. Proses pelaksanaan magang yang berfokus pada pembelajaran langsung melalui praktik memiliki tujuan utama untuk memberikan pengalaman kerja nyata, melatih keterampilan, dan mengembangkan kompetensi sesuai dengan bidang minat serta kondisi dunia kerja sesungguhnya [9]. Melalui kegiatan magang, peserta tidak hanya memperoleh pemahaman mengenai alur kerja operasional sebuah perusahaan, tetapi juga dilatih untuk beradaptasi, berkomunikasi, bekerja sama, dan bertanggung jawab terhadap tugas yang diberikan.

Keberadaan program magang juga memberikan manfaat dalam membantu organisasi mengenali calon tenaga kerja yang potensial, serta mendukung proses regenerasi sumber daya manusia. Magang menjadi bagian penting dalam proses pembentukan profesionalisme dan kesiapan kerja peserta, sekaligus menjadi media yang efektif untuk menghubungkan dunia pendidikan dan dunia industri.

2.1.4 *Website*

Website atau situs web dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan berbagai jenis informasi, seperti teks, gambar diam atau bergerak, animasi, suara, serta kombinasi dari semua elemen tersebut. Informasi ini bisa bersifat statis maupun dinamis dan saling terhubung membentuk suatu struktur yang terpadu. Setiap halaman dalam *website* saling terkait melalui jaringan halaman yang terhubung. Hubungan antara satu halaman web dengan halaman web lainnya disebut *hyperlink*, sedangkan teks yang digunakan sebagai media penghubung disebut *hypertext* [10].

Website pada dasarnya berperan sebagai sarana untuk menyampaikan informasi secara terorganisir dan memungkinkan pengguna mengakses berbagai jenis konten melalui jaringan internet. *Website* dibentuk oleh program browser yang didapat oleh pemakai komputer yang terhubung ke internet. Web merupakan *system* untuk menyebarkan informasi melalui internet. Halaman web biasanya file yang ditulis dalam format HTM dan dapat diakses melalui HTTP [11].

Dalam konteks pembuatan sistem informasi untuk pemantauan program magang, *website* merupakan pilihan yang tepat karena memungkinkan akses dari berbagai peran pengguna seperti admin, mentor, dan peserta magang. Dengan adanya *website*, informasi dapat ditampilkan secara terpusat, mudah diakses, serta bisa diperbarui kapan saja sesuai kebutuhan. Keterhubungan antar bagian *website* melalui *hyperlink* dan *hypertext* juga membantu navigasi yang lebih efektif, sehingga memudahkan pengguna dalam memantau aktivitas, kehadiran, tugas, serta laporan magang secara menyeluruh.

2.1.5 *Backend*

Backend merupakan bagian penting dalam pengembangan sistem informasi yang berfokus pada pengelolaan logika aplikasi, pemrosesan data, serta pengaturan komunikasi antara server dan basis data. Dalam pengembangan web, sebagian besar *framework* menggunakan model arsitektur MVC (*Model View Controller*). Model bertugas mengandung fungsi-fungsi yang terkait dengan *database*, seperti membuat, membaca, memperbarui, dan menghapus data. *Backend* merupakan bagian di mana proses berjalan dalam suatu sistem informasi atau aplikasi, di mana data dapat ditambahkan, diubah, atau dihapus. *Backend* juga bertanggung jawab

atas berbagai proses yang tidak terlihat oleh pengguna, seperti pengelolaan server dan basis data. *Backend* sangat penting dalam pengembangan sistem serta pengelolaan data di dalam sistem [12].

Backend pada dasarnya adalah tempat di mana proses suatu aplikasi atau sistem berjalan. Di *backend*, data diolah, ditambahkan, diubah, atau dihapus. *Backend* mengurus segala sesuatu yang biasanya tidak dilihat atau berinteraksi langsung oleh pengguna, seperti *database* dan server. Bahasa pemrograman yang biasanya dikuasai oleh seorang *backend developer* adalah bahasa pemrograman yang dapat digunakan untuk mengelola *database*, mengolah file, serta *input/output*, seperti PHP, ASP, NodeJs, atau C# [13].

Backend tidak hanya berfungsi menjalankan proses yang tidak terlihat pengguna, tetapi juga bertugas menyediakan layanan melalui API sehingga *frontend* atau aplikasi lain dapat mengakses dan memanfaatkan data secara terstruktur. *Backend* memastikan setiap proses bisnis berjalan sesuai alur yang ditetapkan, seperti autentikasi pengguna, validasi input, pemrosesan logika aplikasi, hingga distribusi data ke modul atau antarmuka lain.

Perlu dibedakan antara *backend* dan *fullstack development*. *Backend* adalah bagian aplikasi yang berjalan di server untuk memproses, menambah, mengubah, dan menghapus data, termasuk mengelola *database* yang tidak terlihat langsung oleh pengguna, dan biasanya dikerjakan oleh *backend developer* yang berfokus pada keamanan, desain sistem, serta manajemen data [14]. Sementara itu, *fullstack developer* merupakan pengembang yang menangani sisi *frontend* sekaligus *backend* dalam satu waktu, di mana selain membangun tampilan dan fitur pada website, *fullstack* juga dituntut untuk memastikan kecepatan akses serta keamanan sistem dari sisi server tetap terjaga [15]. Dalam skripsi ini, pengembangan yang dilakukan dibatasi hanya pada sisi *backend*, sehingga pembahasan difokuskan pada logika aplikasi, pengelolaan basis data, serta penyediaan layanan API tanpa mencakup pengembangan antarmuka pengguna (*frontend*).

2.1.6 Framework Laravel

Laravel merupakan salah satu *framework* berbasis *Hypertext Preprocessor* (PHP) yang banyak digunakan dalam pengembangan aplikasi web karena menyediakan berbagai fitur lengkap yang memudahkan proses pembangunan

sistem. Laravel adalah sebuah *framework* berbasis bahasa pemrograman PHP yang memiliki berbagai fitur lengkap dan sangat membantu para pengembang dalam membangun aplikasi web [16]. *Framework* ini dirancang untuk meningkatkan kualitas pengembangan aplikasi dengan mengurangi biaya pengembangan serta memudahkan proses pemeliharaan aplikasi. Selain itu, Laravel juga meningkatkan produktivitas kerja dengan menyediakan kode program yang rapi serta terstruktur [17]. Sehingga pengembang dapat bekerja lebih efisien dan konsisten dalam menulis kode.

Selain kemudahan dalam struktur dan sintaks, Laravel menyediakan berbagai komponen penting seperti *Eloquent ORM* untuk mengelola *database*, *middleware* untuk memperkuat proses pengamanan, hingga fitur *routing* yang fleksibel untuk mengatur alur permintaan dan *respons*. Laravel juga mendukung pengembangan API berdasarkan REST yang sangat penting dalam pembuatan *backend* yang terpisah dari *frontend*. *Framework* ini dilengkapi dengan sistem autentikasi, manajemen sesi, validasi data, serta mekanisme migrasi *database* yang memudahkan pengembang dalam mengelola perubahan struktur data secara teratur.

Laravel memberikan pembaruan fungsi yang dimana berfungsi untuk dapat berinteraksi dengan *database* yang disebut sebagai *migration*. Dengan *migration*, pengguna laravel dapat dengan mudah untuk melakukan perubahan sebuah *database* pada sebuah platform secara independen karena implementasi skema *database* telah diwakilkan kedalam sebuah *class*.

2.1.7 Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) merupakan bahasa pemrograman berbasis skrip yang berjalan di sisi server dan telah digunakan secara luas untuk pengembangan *website* maupun aplikasi berbasis web. *Hypertext Preprocessor* (PHP) adalah bahasa pemrograman skrip *server-side* yang dirancang untuk pengembangan *website*. Selain itu, *Hypertext Preprocessor* (PHP) juga dapat digunakan sebagai bahasa pemrograman umum. Saat ini, PHP merupakan singkatan dari *Hypertext Preprocessor*, yang merupakan contoh dari istilah rekursif, yaitu istilah yang mengandung kata yang sama dalam definisinya. PHP bisa digunakan secara gratis dan bersifat *open source* [18]. Dengan karakteristik tersebut, PHP

menjadi salah satu bahasa pemrograman yang paling populer dan diadopsi secara luas dalam pengembangan aplikasi web.

PHP bertugas mengolah logika aplikasi, mengelola permintaan dari pengguna, serta berinteraksi langsung dengan basis data untuk menghasilkan halaman web yang dinamis. PHP memiliki fleksibilitas yang tinggi karena dapat dijalankan di berbagai sistem operasi seperti Windows, Linux, dan macOS, serta mendukung berbagai jenis basis data, antara lain MySQL, PostgreSQL, dan MariaDB. Selain itu, PHP memiliki komunitas pengembang yang luas dan menyediakan banyak *library* pendukung, sehingga mempercepat proses pengembangan sistem.

PHP juga menyediakan dukungan untuk pemrograman berorientasi objek (*Object-Oriented Programming*) yang memungkinkan pengembangan aplikasi yang lebih terstruktur dan mudah di-*maintain*. Sintaks PHP yang mirip dengan C dan Java membuat *programmer* yang familiar dengan bahasa tersebut dapat dengan cepat beradaptasi dengan PHP [19]. PHP sering digabungkan dengan *framework* modern seperti Laravel agar dapat menyediakan struktur kode yang lebih terorganisasi, aman, dan mudah dikelola.

2.1.8 MySQL

MySQL merupakan salah satu sistem manajemen basis data relasional yang paling banyak digunakan dalam pengembangan aplikasi web dan sistem informasi. MySQL (*My Struktur Query Language*) atau yang sering disebut “mai-se-kuel” adalah sebuah program pembuatan *database* yang bersifat *open source*, sehingga siapa saja boleh menggunakan dan mengaksesnya tanpa ada pembatasan. MySQL merupakan alat akses *database* yang mendukung jaringan, sehingga cocok digunakan dalam aplikasi yang memiliki banyak pengguna atau *multi user*. Saat ini, MySQL telah digunakan oleh hampir semua pengembang *database*, terutama dalam pengembangan aplikasi web [20].

MySQL mendukung bahasa pemrograman PHP, bahasa permintaan yang terstruktur, karena pada penggunaannya SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI. MySQL merupakan RDBMS (*Relational Database Management System*) server. RDBMS adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan

data pada suatu model relational. Dengan demikian, tabel-tabel yang ada pada *database* memiliki relasi antara satu tabel dengan tabel lainnya [21].

Sebagai sistem manajemen basis data yang mendukung bahasa SQL. MySQL dirancang untuk mengelola data dalam bentuk tabel-tabel relasional yang saling terhubung. MySQL memiliki keunggulan dalam hal kecepatan eksekusi *query*, kemudahan konfigurasi, serta kompatibilitas tinggi dengan berbagai bahasa pemrograman seperti PHP, Python, dan JavaScript.

2.1.9 Database Management System (DBMS)

Database Management System (DBMS) merupakan alat yang digunakan untuk mengelola data yang sudah tersimpan dalam basis data dengan menggunakan prosedur yang diatur secara komputerisasi. Tujuan utama dari DBMS adalah menyediakan akses yang efisien dan aman serta memastikan integritas data tetap terjaga. DBMS berperan dalam mengatur struktur data, menyimpan data dalam bentuk yang terorganisir, serta menyediakan mekanisme untuk menjaga konsistensi dan keamanan data [22].

Database Management System (DBMS) memiliki peran yang sangat penting dalam mendukung kelancaran operasional dan pengambilan keputusan di perusahaan. DBMS tidak hanya berfungsi sebagai alat penyimpanan data, tetapi juga sebagai sistem yang mengelola, mengolah, dan memastikan data selalu tersedia, akurat, dan aman. Salah satu upaya penting dalam meningkatkan efisiensi operasional perusahaan adalah dengan mengimplementasikan *Database Management System* (DBMS) [23].

Penggunaan *Database Management System* (DBMS) juga mendukung pengelolaan data secara terpusat sehingga memudahkan proses pemantauan, pemeliharaan, dan pengembangan sistem informasi. Dengan adanya DBMS, perusahaan dapat mengatur hak akses pengguna, melakukan pencadangan data (*backup*), serta menjaga keberlangsungan data apabila terjadi gangguan sistem. Hal ini menjadikan DBMS sebagai komponen penting dalam membangun sistem informasi yang andal, terstruktur, dan berkelanjutan.

2.1.10 RESTful API

REST API adalah sebuah implementasi dari API (*Application Programming Interface*). REST (*Representational State Transfer*) adalah suatu arsitektur metode

komunikasi yang menggunakan protokol HTTP untuk pertukaran data [24]. Arsitektur layanan web yang memungkinkan sistem saling berkomunikasi melalui protokol HTTP secara sederhana, cepat, dan terstandarisasi. REST memanfaatkan metode HTTP seperti GET, POST, PUT, DELETE untuk melakukan operasi pada sumber daya (*resources*) yang direpresentasikan dalam format umum seperti JSON.

API dapat diartikan sebagai penghubung antara satu aplikasi dengan aplikasi lainnya, sehingga memungkinkan programmer menggunakan fungsi-fungsi dari sistem tersebut. Proses ini diatur dan dikelola melalui sistem operasi. Keunggulan dari API adalah memungkinkan aplikasi satu dengan aplikasi lainnya dapat saling berhubungan dan berinteraksi [25].

Dalam pengembangannya, RESTful API banyak digunakan pada aplikasi modern karena struktur *endpoint*-nya yang fleksibel, mudah diintegrasikan, dan mendukung pemisahan antara *frontend* dan *backend*. API ini memungkinkan berbagai platform seperti aplikasi web, *mobile*, maupun layanan pihak ketiga mengakses data secara konsisten melalui URL yang terstandarisasi. Dengan demikian, RESTful API menjadi fondasi penting dalam membangun sistem terdistribusi dan pengembangan aplikasi berbasis arsitektur layanan.

2.1.11 Unified Modeling Language (UML)

Unified Modelling Language (UML) merupakan salah satu standar bahasa yang secara luas diterapkan di berbagai sektor industri untuk menggambarkan kebutuhan, melakukan analisis dan desain, serta menjelaskan struktur arsitektur dalam pengembangan perangkat lunak yang berbasis pemrograman berorientasi objek. Dalam proses pengembangan teknik pemrograman berorientasi objek, muncul sebuah standar bahasa pemodelan yang digunakan dalam pengembangan perangkat lunak dengan pendekatan pemrograman berorientasi objek, yaitu *Unified Modelling Language* (UML). UML menyediakan serangkaian notasi diagram yang dapat digunakan untuk memodelkan struktur maupun perilaku sistem sehingga proses perancangan dapat dilakukan dengan lebih terstruktur, jelas, dan mudah dipahami oleh seluruh pihak yang terlibat [28].

UML bertujuan untuk menyediakan bahasa pemodelan yang independen dari berbagai bahasa pemrograman dan proses rekayasa. Selain itu, penggunaan UML memiliki penggunaan standarisasi praktik terbaik saat ini dalam pemodelan

dan menyediakan model siap pakai [29]. Dengan sifatnya yang standar dan universal, UML memudahkan komunikasi antara pengembang, analis sistem, dan pemangku kepentingan lainnya dalam memahami rancangan sistem yang akan dibangun.

Dengan menggunakan diagram seperti *use case*, *activity*, *sequence*, dan *class diagram*, para pengembang dapat mendefinisikan kebutuhan sistem secara tepat sebelum memasuki tahap implementasi. Setiap diagram memiliki peran masing-masing dalam menggambarkan fungsionalitas, alur proses, interaksi antar objek, serta struktur kelas, sehingga membantu mengurangi kesalahan interpretasi pada tahap pengembangan. Selain itu, penggunaan diagram UML juga mempermudah proses pemeliharaan dan pengembangan sistem di masa mendatang.

Alat bantu yang digunakan dalam perancangan berorientasi objek menggunakan UML pada penelitian ini dapat dijelaskan sebagai berikut:

a. *Use Case Diagram*

Diagram *Use Case* adalah gambaran grafis yang menampilkan beberapa atau semua aktor, *use case*, serta interaksi antara keduanya yang menjelaskan suatu sistem [30]. *Use Case Diagram* digunakan untuk menunjukkan fungsi-fungsi yang bisa digunakan oleh pengguna sesuai dengan peran atau aktor dalam sistem.

Diagram ini membantu menjelaskan bagaimana pengguna berinteraksi dengan sistem, fitur apa saja yang bisa diakses, serta batasan hak akses yang dimiliki setiap aktor. Dengan demikian, perancang sistem dapat memahami kebutuhan pengguna secara lebih baik dan memastikan setiap proses utama dalam sistem tercakup dengan baik. Selain itu, *Use Case Diagram* memberikan gambaran awal mengenai cakupan sistem yang akan dikembangkan, sehingga memudahkan proses analisis kebutuhan, perencanaan fitur, serta pengujian pada tahap penerapan.

Use Case Diagram juga berfungsi sebagai acuan awal dalam pengembangan sistem, baik pada tahap perancangan maupun implementasi, karena mampu menyelaraskan kebutuhan pengguna dengan fungsi sistem yang akan dibangun. Diagram ini membantu mengurangi kesalahan pemahaman antara pengembang dan pemangku kepentingan, serta