

## BAB II LANDASAN TEORI

Bab ini mengkaji penelitian-penelitian terdahulu yang relevan serta membahas berbagai konsep dan teori yang menjadi dasar dalam pelaksanaan penelitian. Pembahasan difokuskan pada pemahaman teoritis dan empiris yang berkaitan dengan topik penelitian, sehingga mampu memberikan gambaran yang komprehensif terhadap permasalahan yang dikaji. Selain itu, landasan teori yang disajikan berfungsi sebagai kerangka konseptual dalam mendukung proses analisis serta interpretasi hasil penelitian.

### 2.1 Penelitian Terdahulu

Penelitian yang dilakukan oleh Saputri et al. (2024) berfokus pada peningkatan kinerja aplikasi survei berbasis web dengan memanfaatkan framework Laravel dan teknologi cloud. Metode pengembangan yang digunakan adalah prototype, yang mencakup tahapan analisis kebutuhan, perancangan sistem, implementasi, serta pengujian. Perancangan sistem divisualisasikan menggunakan Data Flow Diagram (DFD) dan Entity Relationship Diagram (ERD).

Hasil penelitian menunjukkan bahwa penerapan teknik optimasi seperti caching, optimasi query, dan load balancing mampu meningkatkan kecepatan respons aplikasi secara signifikan hingga mencapai 50%. Meskipun demikian, penelitian tersebut masih terbatas pada pengukuran performa aplikasi dan belum mengkaji aspek keamanan maupun pengalaman pengguna secara lebih mendalam.

**Tabel 2.1 Ringkasan Penelitian Terdahulu 1**

No	Nama dan Tahun	Judul Penelitian	Metode	Hasil Penelitian	Analisis Gap
1	Saputri et al. (2024)	Optimasi performa aplikasi survei berbasis Laravel dan cloud	Prototype (analisis kebutuhan, perancangan, implementasi, pengujian), DFD, ERD	Aplikasi survei web dengan <i>caching, query optimization, load balancing</i> , meningkatkan kecepatan respons hingga 50%	Fokus pada <i>optimasi performa aplikasi survei berbasis Laravel dan cloud</i> , namun hanya mengukur kecepatan respons. Belum ada pengujian terhadap aspek <i>security</i> (keamanan aplikasi survei) atau <i>user experience</i> (misal skalabilitas terhadap jumlah user aktif secara bersamaan).

2	Munandar et al. (2024)	Evaluasi implementasi AMI pada distribusi listrik PLN	Studi kasus, pengumpulan data primer-sekunder, analisis kinerja	AMI meningkatkan akurasi pencatatan konsumsi listrik <i>real-time</i> dan menurunkan biaya operasional pencatatan manual	Fokus pada AMI untuk distribusi listrik PLN terkait akurasi dan biaya, tetapi belum membahas integrasi AMI dengan aplikasi berbasis web atau sistem backend berbasis framework seperti Laravel atau lainnya.
3	Doe dan Smith (2023)	Optimasi <i>database</i> Laravel untuk aplikasi web	Eksperimen komputasi ( <i>raw query</i> , <i>query builder</i> , <i>ORM Eloquent</i> )	<i>Raw query</i> terbaik untuk dataset besar, <i>ORM Eloquent</i> lebih mudah namun lebih lambat, pentingnya pengindeksan dan query efisien	Membandingkan performa query di Laravel, namun fokus hanya pada kecepatan eksekusi query. Belum ada eksplorasi terhadap dampak optimasi database pada <i>resource usage</i> (memori/CPU) atau <i>konkurensi</i> (akses bersamaan).
4	Brown dan Zaki (2024)	Perbandingan performa Laravel dan <i>CodeIgniter</i>	Eksperimen performa (kecepatan eksekusi, memori, stabilitas)	<i>CodeIgniter</i> lebih cepat untuk aplikasi ringan, Laravel lebih stabil untuk aplikasi kompleks berfitur lengkap	Membandingkan performa Laravel dan <i>CodeIgniter</i> dari sisi kecepatan dan stabilitas, tetapi tidak membahas <i>biaya pemeliharaan</i> , <i>kemudahan integrasi ke cloud service</i> , atau <i>dukungan teknologi modern</i> seperti API atau <i>microservice</i> .
5	Li dan Gonzales (2024)	Optimasi aplikasi survei Laravel berbasis <i>cloud</i>	Eksperimen ( <i>caching</i> , <i>query tuning</i> , <i>load balancing</i> )	<i>Response time</i> turun dari 4,5 detik ke 2,2 detik, <i>availability</i> meningkat hingga 99,9%	Fokus pada optimasi <i>response time</i> dan <i>availability</i> aplikasi survei Laravel berbasis <i>cloud</i> , namun tidak membahas efektivitas <i>cost</i> (biaya <i>cloud</i> ) atau <i>monitoring &amp; alerting</i> untuk jaminan performa jangka panjang.

Berdasarkan penelitian terdahulu tersebut, dapat disimpulkan bahwa sebagian besar penelitian sebelumnya lebih menitikberatkan pada optimasi performa aplikasi berbasis web atau infrastruktur tertentu. Berbeda dengan penelitian ini, fokus utama

diarahkan pada perancangan dan pengembangan sistem informasi estimasi pemakaian gas rumah tangga yang bertujuan meningkatkan efisiensi operasional serta mengurangi ketergantungan terhadap pencatatan manual. Dengan demikian, penelitian ini memiliki kontribusi berbeda dalam penerapan sistem informasi berbasis web untuk mendukung proses bisnis pada sektor utilitas energi.

### **2.1.1 Hasil Penelitian Saputri, et al. (2024)**

Penelitian yang dilakukan oleh Saputri et al. (2024) bertujuan untuk meningkatkan kinerja aplikasi survei berbasis web dengan memanfaatkan framework Laravel dan teknologi komputasi awan. Metode yang digunakan dalam penelitian tersebut adalah prototype, yang mencakup tahapan analisis kebutuhan, perancangan sistem, implementasi, serta pengujian performa. Perancangan sistem divisualisasikan menggunakan *Data Flow Diagram (DFD)* dan *Entity Relationship Diagram (ERD)*, sedangkan proses pengembangan aplikasi dilakukan dengan menggunakan Laravel, bahasa pemrograman PHP, serta basis data MySQL. Hasil penelitian menunjukkan bahwa penerapan teknik *caching*, optimasi query, dan load balancing pada lingkungan cloud mampu meningkatkan performa aplikasi survei berbasis web secara signifikan, dengan peningkatan kecepatan respons hingga 50% dibandingkan dengan sistem sebelum dilakukan optimasi.

Persamaan antara penelitian ini dengan penelitian terdahulu adalah adanya kesesuaian fokus penelitian dengan penelitian yang dilakukan penulis. Dalam hal pengembangan aplikasi berbasis web dengan *framework* Laravel untuk mempermudah pengelolaan data dan proses bisnis. Keduanya mengadopsi teknologi modern untuk meningkatkan efisiensi operasional serta memanfaatkan pendekatan pemodelan sistem melalui DFD dan ERD untuk mendukung perancangan sistem yang sistematis. Selain itu, kedua penelitian menekankan pentingnya performa aplikasi dalam memproses data dalam jumlah besar agar dapat mendukung kebutuhan operasional.

Perbedaan antara penelitian ini dan penelitian terdahulu terletak pada fokus kajian, di mana penelitian sebelumnya lebih menitikberatkan pada optimasi aplikasi survei berbasis web melalui pemanfaatan teknologi cloud dan penerapan teknik peningkatan performa, tanpa secara spesifik membahas domain energi atau estimasi data berbasis historis. Sementara itu, penelitian saya berfokus pada pengembangan

sistem informasi estimasi pemakaian gas rumah tangga berbasis Laravel dengan tujuan utama memperkirakan konsumsi gas secara otomatis dan akurat berdasarkan data historis, yang akan digunakan sebagai dasar perhitungan tagihan dan pengelolaan distribusi gas nasional.

### **2.1.2 Hasil Penelitian Munandar, et al. (2024)**

Pada penelitian yang dilakukan oleh Munandar et al. (2024), bertujuan untuk mengevaluasi implementasi *Advanced Metering Infrastructure* (AMI) pada distribusi listrik di PLN. Penelitian ini menggunakan pendekatan studi kasus dengan pengumpulan data primer dan sekunder, analisis infrastruktur AMI, dan evaluasi kinerja sistem melalui parameter keandalan, akurasi, dan efisiensi. Hasil penelitian menunjukkan bahwa AMI mampu meningkatkan akurasi pencatatan konsumsi listrik secara *real-time* dan mengurangi biaya operasional pencatatan manual.

Persamaan antara penelitian ini dan penelitian terdahulu terletak pada kesamaan tujuan, yaitu meningkatkan akurasi data konsumsi energi serta mengurangi ketergantungan terhadap proses pencatatan manual. Kedua penelitian tersebut menekankan peran penting sistem informasi atau teknologi dalam mendukung efisiensi operasional serta menghasilkan data yang akurat sebagai dasar pengambilan keputusan strategis.

Perbedaan antara penelitian ini dan penelitian terdahulu terletak pada ruang lingkup kajian, di mana penelitian sebelumnya menitikberatkan pada infrastruktur *Advanced Metering Infrastructure* (AMI) dalam distribusi energi listrik dengan teknologi *smart meter* dan sistem komunikasi *real-time*, sedangkan penelitian saya berfokus pada estimasi konsumsi gas berbasis data historis dengan pengembangan sistem berbasis web menggunakan Laravel, tanpa bergantung pada perangkat keras *smart meter* atau komunikasi *real-time*.

### **2.2.1 Hasil Penelitian John Doe dan Jane Smith (2023)**

Pada penelitian yang dilakukan oleh Doe dan Smith (2023), penelitian ini bertujuan untuk menganalisis strategi optimasi *database* pada *framework* Laravel untuk meningkatkan performa aplikasi berbasis web. Metode penelitian yang digunakan berupa eksperimen komputasi, dengan membandingkan tiga pendekatan *query*: *raw query*, *query builder*, dan *ORM Eloquent*. Hasil eksperimen

menunjukkan bahwa *raw query* memiliki performa terbaik pada dataset besar, sedangkan ORM *Eloquent* lebih mudah digunakan namun lebih lambat. Penelitian ini juga menyoroti pentingnya pengindeksan *database* dan penghapusan *wildcard* pada *query* untuk mengoptimalkan kinerja sistem berbasis Laravel.

Persamaannya adalah Penelitian ini memiliki kesamaan dengan penelitian saya, yaitu sama-sama membahas optimasi sistem berbasis Laravel untuk meningkatkan efisiensi pengelolaan data. Kedua penelitian mengadopsi teknologi modern untuk mendukung kinerja aplikasi berbasis web. Perbedaannya adalah Penelitian sebelumnya fokus pada optimasi teknis performa *database* dan *query* pada Laravel, tanpa konteks spesifik domain seperti energi atau gas. Sementara penelitian saya selain mengoptimasi performa juga mengembangkan sistem estimasi pemakaian gas rumah tangga dengan alur bisnis spesifik di PT XYZ.

### **2.1.3 Hasil Penelitian Emily Brown dan Ahmad Zaki (2024)**

Penelitian ini dilakukan oleh Brown dan Zaki (2024) dengan tujuan untuk membandingkan teknik optimasi performa antara *framework* Laravel dan *CodeIgniter*. Studi ini menggunakan metode eksperimen dengan menguji kecepatan eksekusi, penggunaan memori, dan kestabilan sistem pada aplikasi simulasi. Hasilnya, *CodeIgniter* unggul dalam kecepatan untuk aplikasi ringan, sedangkan Laravel lebih stabil untuk aplikasi kompleks karena fitur-fiturnya yang lebih lengkap.

Persamaannya adalah sama-sama berfokus pada optimasi aplikasi berbasis web untuk menghasilkan sistem yang efisien dan responsif. Kedua penelitian ini menekankan pentingnya performa dalam pengembangan perangkat lunak agar aplikasi dapat berjalan dengan cepat dan stabil. Selain itu, keduanya sama-sama mempertimbangkan faktor teknis seperti kecepatan eksekusi, penggunaan memori, dan kestabilan sistem dalam proses evaluasi. Penelitian sebelumnya maupun penelitian saya sama-sama menggunakan pendekatan yang berorientasi pada eksperimen untuk mengukur hasil secara objektif. Dengan demikian, baik penelitian terdahulu maupun penelitian saya memiliki tujuan yang sejalan, yaitu menghadirkan solusi teknologi yang mampu meningkatkan kualitas aplikasi web sesuai dengan kebutuhan pengguna.

Perbedaannya adalah Penelitian sebelumnya hanya membahas perbandingan *framework* dari sisi teknis umum tanpa fokus pada aplikasi estimasi gas atau integrasi data *smart meter*, sedangkan penelitian saya fokus pada pengembangan sistem spesifik untuk mendukung proses bisnis XYZ.

#### **2.1.4 Hasil Penelitian Chen Li dan Maria Gonzalez (2024)**

Penelitian yang dilakukan oleh Li dan Gonzales (2024) bertujuan untuk mengoptimalkan aplikasi survei berbasis web dengan memanfaatkan *framework* Laravel dan teknologi *cloud computing*. Metode yang digunakan berupa eksperimen peningkatan performa melalui implementasi *caching*, *query tuning*, dan *load balancing* berbasis *cloud*. Hasil penelitian menunjukkan adanya penurunan *response time* dari 4,5 detik menjadi 2,2 detik serta peningkatan *availability* hingga 99,9%.

Persamaan dari penelitian ini adalah penelitian tersebut serupa karena sama-sama menggunakan **framework Laravel** sebagai dasar pengembangan aplikasi. Keduanya berfokus pada pemanfaatan teknik optimasi untuk meningkatkan performa aplikasi berbasis web sehingga lebih efisien dan responsif. Selain itu, penelitian ini sama-sama menekankan pentingnya pengelolaan sumber daya sistem seperti memori dan kecepatan eksekusi. Hal ini menunjukkan bahwa baik penelitian terdahulu maupun penelitian saya memiliki perhatian yang sama terhadap kualitas teknis aplikasi yang dibangun. Dengan demikian, kesamaan utama terletak pada penggunaan *framework* Laravel serta penerapan optimasi sebagai strategi untuk mendukung performa sistem yang lebih baik.

Perbedaan dari penelitian ini adalah, Penelitian sebelumnya fokus pada aplikasi survei *online* dengan optimasi berbasis *cloud*, sedangkan penelitian ini fokus pada sistem estimasi konsumsi gas rumah tangga dengan optimasi *database* untuk mendukung volume data besar dari catatan stand meter pelanggan.

## **2.2 Landasan Teori**

Bagian ini menguraikan landasan teori yang digunakan dalam penelitian, yang berfungsi sebagai dasar dalam penyusunan kerangka penelitian serta perumusan hipotesis. Landasan teori yang dibahas meliputi konsep perancangan sistem informasi, metode Waterfall, *framework* Laravel, Laragon, serta teknologi pendukung lainnya seperti CSS, JavaScript, HTML, dan MySQL.

### **2.2.1 PT XYZ**

PT XYZ bergerak di bidang transmisi dan distribusi gas bumi di Indonesia. Sejak awal berdirinya pada tahun 1859 sebagai perusahaan gas kota di Batavia, PT XYZ telah berkembang menjadi perusahaan energi strategis dalam mendukung kemandirian energi nasional. PT XYZ mengelola jaringan pipa gas bumi yang panjangnya mencapai ribuan kilometer dan melayani berbagai segmen pelanggan, mulai dari industri besar, komersial, hingga rumah tangga melalui program jaringan gas (jargas). Dalam penyaluran gas rumah tangga, PT XYZ berupaya mengurangi ketergantungan masyarakat terhadap LPG dengan menyediakan alternatif energi yang lebih aman, murah, dan ramah lingkungan. Namun, dalam implementasinya, penyaluran gas rumah tangga menghadapi berbagai tantangan, seperti perlunya pembangunan infrastruktur pipa di daerah-daerah baru, biaya investasi awal yang tinggi, kesadaran masyarakat yang belum merata, serta kebutuhan akan sistem pemantauan dan pencatatan pemakaian gas yang akurat agar distribusi berjalan efisien dan transparan. Untuk menjawab tantangan tersebut, PT XYZ terus melakukan transformasi digital, termasuk penerapan smart metering dan sistem estimasi pemakaian gas berbasis teknologi agar layanan semakin optimal dan terpercaya.

### **2.2.2 Sistem Informasi**

Sistem informasi, sebagaimana ditunjukkan pada Gambar 2.1, menurut Anggraeni (2020) merupakan suatu jaringan prosedur kerja yang saling berkaitan dan terintegrasi untuk mencapai tujuan tertentu. Sistem informasi dapat dipahami sebagai suatu kesatuan komponen yang saling terintegrasi dan bekerja secara terkoordinasi untuk mengelola data menjadi informasi yang bermanfaat. Sistem ini berperan dalam mendukung aktivitas operasional, pengelolaan transaksi, serta pengambilan keputusan pada suatu organisasi. Informasi yang dihasilkan merupakan hasil pengolahan data mentah sehingga memiliki nilai tambah dan relevansi bagi pengguna dalam menjalankan tugas maupun menentukan strategi organisasi (Anggraeni, 2020).



**Gambar 2.1 Sistem Informasi**  
(Sumber : admin, 2024)

### **2.2.3 Metode *Waterfall***

Metode *Waterfall*, sebagaimana ditunjukkan pada Gambar 2.2, merupakan pendekatan pengembangan perangkat lunak yang menerapkan tahapan proses secara berurutan dan sistematis. Setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Adapun tahapan dalam metode *Waterfall* adalah sebagai berikut:

1. Analisis Kebutuhan Perangkat Lunak

Tahap ini bertujuan untuk mengidentifikasi dan mengumpulkan kebutuhan sistem, termasuk kebutuhan fungsional dan nonfungsional, seperti dokumen dan antarmuka pengguna. Proses ini dilakukan untuk memahami kebutuhan pengguna secara menyeluruh serta menentukan solusi perangkat lunak yang sesuai dalam mendukung komputersasi sistem.

2. Perancangan Sistem (*Design*)

Pada tahap perancangan, dilakukan penyusunan desain perangkat lunak yang meliputi arsitektur sistem, struktur data, rancangan antarmuka, serta prosedur pengkodean. Perancangan sistem divisualisasikan menggunakan *Unified Modeling Language (UML)*, antara lain *Activity Diagram*, *Use Case Diagram*, *Sequence Diagram*, dan *Deployment Diagram*. Selain itu, perancangan basis data dilakukan dengan menggunakan *Entity Relationship Diagram (ERD)* dan *Logical Record Structure (LRS)*.

3. Implementasi atau Pembuatan Kode Program

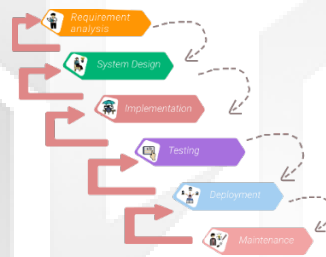
Tahap implementasi merupakan proses penerjemahan desain sistem ke dalam bentuk kode program. Hasil dari tahap ini adalah aplikasi atau perangkat lunak yang dikembangkan sesuai dengan rancangan yang telah ditetapkan pada tahap sebelumnya.

#### 4. Pengujian Sistem

Tahap pengujian bertujuan untuk memastikan bahwa sistem telah berjalan sesuai dengan kebutuhan yang ditentukan, baik dari sisi logika maupun fungsionalitas. Pengujian dilakukan menggunakan metode *Blackbox Testing*, yaitu metode pengujian yang berfokus pada keluaran sistem berdasarkan masukan tertentu tanpa memperhatikan struktur internal kode program. Metode ini dapat diterapkan pada berbagai tingkat pengujian, mulai dari pengujian unit hingga pengujian penerimaan.

#### 5. Pemeliharaan (*Maintenance*)

Tahap pemeliharaan mencakup upaya perbaikan, penyesuaian, dan pengembangan lanjutan terhadap sistem yang telah diimplementasikan. Tahap ini bertujuan untuk mengantisipasi perubahan kebutuhan, perkembangan teknologi, serta penyesuaian terhadap perangkat keras maupun perangkat lunak yang digunakan (Badrul, 2021).



**Gambar 2.2 Metode *Waterfall***  
(Sumber : Board Infinity, 2023)

#### 2.2.4 Laravel

Laravel, sebagaimana ditunjukkan pada Gambar 2.3, merupakan framework PHP yang dirilis dengan lisensi MIT dan digunakan untuk pengembangan aplikasi web berbasis *Model–View–Controller (MVC)*. *Framework* ini ditulis menggunakan bahasa pemrograman PHP dan dirancang untuk meningkatkan kualitas perangkat lunak dengan menekan biaya pengembangan serta pemeliharaan, sekaligus memberikan pengalaman pengembangan yang lebih efisien melalui sintaks yang ekspresif, sederhana, dan mudah dipahami (Hermanto, 2019).

Laravel dikembangkan oleh Taylor Otwell sebagai *framework* web PHP yang bersifat *open source* dan dapat digunakan secara bebas. *Framework* ini dirancang untuk membangun aplikasi web dengan menerapkan arsitektur *Model–View–Controller (MVC)*. Implementasi pola MVC pada Laravel memiliki karakteristik tersendiri, salah satunya dengan adanya mekanisme *routing* yang berfungsi sebagai

penghubung antara permintaan pengguna dan *controller*, sehingga permintaan tidak langsung diteruskan ke pengendali aplikasi (Cendani, 2023).



**Gambar 2.3 Laravel**  
(Sumber : admin, 2023)

Laravel merupakan framework pengembangan aplikasi web berbasis PHP yang bersifat *open source* dan menerapkan arsitektur Model View Controller (MVC). Framework ini dirancang untuk mempermudah pengembangan aplikasi dengan menyediakan struktur kode yang terorganisasi, fitur keamanan bawaan, serta berbagai *tool* pendukung seperti ORM Eloquent, Blade Template, dan Artisan Command. Dengan dukungan dokumentasi yang lengkap, Laravel mampu mempercepat proses pengembangan dan memudahkan pemeliharaan sistem.

### 2.2.5 PHP 8.2

Gambar 2.4 memperlihatkan *Hypertext Preprocessor (PHP)*, yaitu bahasa skrip open source yang memiliki beragam fungsi dan banyak digunakan dalam pengembangan aplikasi web. PHP dapat diintegrasikan secara langsung dengan *Hypertext Markup Language (HTML)*. Keunggulan PHP dibandingkan bahasa sejenis terletak pada mekanisme eksekusi kode yang dilakukan di sisi server, sehingga pengguna atau klien tidak dapat mengetahui proses pemrograman yang berjalan di belakang layar. Bahasa pemrograman PHP (*Hypertext Preprocessor*) merupakan jenis skrip yang dijalankan di sisi server (*server-side scripting*) dan dipasang pada web server, sehingga memungkinkan pembuatan situs web yang dinamis dengan menyisipkan kode PHP ke dalam HTML dan memprosesnya sebelum dikirim ke browser pengguna. PHP dieksekusi oleh server web, dan hasil eksekusinya bukan kode mentah dikirimkan ke klien dalam bentuk HTML atau tipe konten lainnya (Smith, 2025). Selain itu, statistik terkini menunjukkan bahwa PHP masih mendominasi sebagai bahasa pemrograman sisi server untuk banyak situs

web sekitar 79% dari 10 juta situs web paling populer menggunakan PHP sebagai teknologi *server-side* (W3Techs, 2025). Dalam konteks penelitian ini, PHP dipahami sebagai bahasa pemrograman yang digunakan untuk mengembangkan aplikasi pembelajaran bahasa Inggris sebagaimana dijelaskan oleh Johar (2019).



**Gambar 2.4 PHP**  
(Sumber : Nao, 2025)

PHP merupakan singkatan dari Hypertext Preprocessor, yaitu bahasa pemrograman web yang bersifat open source dan dijalankan pada sisi server (*server-side*). PHP dapat diintegrasikan dengan HTML dan digunakan untuk membangun halaman web yang bersifat dinamis, di mana proses pembentukan halaman dilakukan saat klien mengajukan permintaan. Mekanisme eksekusi di sisi server ini memungkinkan penyajian data yang selalu terkini dan akurat kepada pengguna, sementara seluruh skrip PHP dijalankan sepenuhnya pada server (Cahyono & Jayanti, 2022).

Dalam penelitian ini, PHP versi 8.2 digunakan sebagai bahasa pemrograman utama. PHP adalah bahasa pemrograman *server-side* yang digunakan untuk membangun aplikasi web dinamis. Kode PHP dijalankan pada sisi server sehingga pengguna tidak dapat melihat proses pemrograman yang terjadi di belakang layar. Bahasa ini bersifat *open source*, fleksibel, serta mudah diintegrasikan dengan HTML dan berbagai sistem manajemen basis data. Dalam penelitian ini, PHP digunakan sebagai bahasa utama dalam pengembangan sistem karena memiliki performa yang baik dan kompatibilitas yang luas.

### **2.2.6 XAMPP**

Gambar 2.5 adalah XAMPP merupakan paket perangkat lunak bebas (*open source*) yang digunakan untuk menyediakan lingkungan server lokal guna mendukung pengembangan aplikasi berbasis web. XAMPP merupakan singkatan dari X (Cross-platform), Apache, MySQL/MariaDB, PHP, dan Perl. Perangkat lunak ini dikembangkan oleh komunitas Apache Friends dengan tujuan untuk

memudahkan instalasi dan konfigurasi server lokal yang lengkap dan siap digunakan (Apache Friends, 2024).

XAMPP sering digunakan oleh pengembang perangkat lunak, khususnya dalam tahap pengembangan dan pengujian (*development and testing*), karena memungkinkan aplikasi web dijalankan secara lokal tanpa perlu koneksi ke server eksternal. Hal ini mempercepat proses pengembangan dan mengurangi risiko gangguan akibat faktor eksternal.

XAMPP terdiri dari beberapa komponen inti, antara lain:

a. Apache

Apache adalah perangkat lunak server web yang digunakan untuk menangani permintaan (*request*) dari klien dan menyajikan halaman web kepada pengguna melalui protokol HTTP/HTTPS.

b. MySQL / MariaDB

Merupakan sistem manajemen basis data relasional (*Relational Database Management System / RDBMS*) yang berfungsi untuk menyimpan, mengelola, dan memanipulasi data. Dalam versi terbaru, XAMPP menggunakan MariaDB sebagai pengganti MySQL.

c. PHP

PHP adalah bahasa pemrograman sisi server (*server-side scripting*) yang digunakan secara luas dalam pengembangan aplikasi web dinamis.

d. Perl

Meskipun tidak sepopuler PHP dalam pengembangan web modern, Perl tetap disertakan untuk mendukung kompatibilitas dengan berbagai jenis aplikasi yang membutuhkannya.

e. phpMyAdmin

Sebuah aplikasi berbasis web yang digunakan untuk mengelola database MySQL/MariaDB melalui antarmuka grafis. phpMyAdmin memudahkan pengguna dalam melakukan query, membuat tabel, dan manajemen data tanpa perlu menggunakan baris perintah.

XAMPP memiliki berbagai fungsi dan manfaat dalam proses pengembangan perangkat lunak, di antaranya:

- a. Menyediakan lingkungan server lokal untuk menjalankan dan menguji aplikasi web secara offline.
- b. Memungkinkan pengembangan berbasis Content Management System (CMS) seperti WordPress, Joomla, dan Drupal.
- c. Mendukung proses pembelajaran dan pelatihan dalam pengembangan aplikasi berbasis web.
- d. Memfasilitasi pengujian fitur-fitur aplikasi sebelum dilakukan deployment ke server produksi.

Beberapa keunggulan XAMPP yang membuatnya banyak digunakan dalam lingkungan pengembangan adalah:

- a. Mudah digunakan: Instalasi dan konfigurasi sangat sederhana, bahkan untuk pengguna pemula.
- b. Gratis dan open source: Tidak memerlukan lisensi dan dapat dimodifikasi sesuai kebutuhan.
- c. Cross-platform: Dapat dijalankan di berbagai sistem operasi seperti Windows, Linux, dan macOS.
- d. Komprehensif: Sudah mencakup seluruh komponen utama yang dibutuhkan dalam pengembangan web.



**Gambar 2.5 XAMPP**  
(Sumber : Apache Friends, 2024)

### 2.2.7 MySQL

Gambar 2.6 adalah MySQL merupakan sistem manajemen basis data relasional (*Relational Database Management System* atau RDBMS) yang bersifat open source dan banyak digunakan dalam pengembangan aplikasi berbasis web. MySQL dikembangkan pertama kali oleh perusahaan Swedia, MySQL AB, pada tahun 1995, dan saat ini dimiliki serta dikembangkan oleh Oracle Corporation

setelah akuisisi pada tahun 2010 (Oracle, 2024).

MySQL menggunakan bahasa Structured Query Language (SQL) sebagai standar untuk mengelola, mengakses, dan memanipulasi data dalam basis data. MySQL dirancang untuk bekerja secara cepat, andal, dan mudah digunakan, serta mendukung berbagai platform sistem operasi seperti Windows, Linux, dan macOS.

MySQL memiliki arsitektur berbasis *client-server*, yang memungkinkan pengguna atau aplikasi klien untuk mengakses server basis data melalui jaringan menggunakan protokol tertentu. MySQL menyediakan sejumlah fitur utama yang menjadikannya salah satu RDBMS paling populer di dunia, di antaranya:

a. Dukungan untuk Transaksi

MySQL mendukung transaksi ACID (Atomicity, Consistency, Isolation, Durability) melalui mesin penyimpanan seperti InnoDB, yang memungkinkan pengelolaan data secara andal dan aman.

b. Multi-user Access

MySQL memungkinkan banyak pengguna untuk mengakses database secara bersamaan tanpa saling mengganggu.

c. Keamanan Data

MySQL menyediakan kontrol akses berbasis hak istimewa (*privilege-based access control*) dan sistem otentikasi yang aman.

d. Replikasi dan Skalabilitas

MySQL mendukung fitur replikasi database yang memungkinkan distribusi data ke beberapa server untuk meningkatkan ketersediaan dan keandalan sistem.

e. Kompatibilitas dengan Berbagai Bahasa Pemrograman

MySQL dapat digunakan bersama berbagai bahasa pemrograman seperti PHP, Java, Python, dan C++ melalui pustaka API atau konektor resmi.

MySQL digunakan secara luas dalam berbagai aplikasi, terutama yang berbasis web. Beberapa penggunaan umum MySQL meliputi:

- a. Penyimpanan data pada situs web dinamis seperti blog, forum, dan toko daring.
- b. Backend database untuk Content Management System (CMS) seperti WordPress, Joomla, dan Drupal.
- c. Basis data untuk sistem informasi organisasi seperti sistem akademik, keuangan, dan logistik.

- d. Komponen utama dalam arsitektur teknologi seperti LAMP (Linux, Apache, MySQL, PHP/Python/Perl).

#### Keunggulan MySQL

- a. Kinerja tinggi: Mampu menangani jutaan baris data dengan performa tinggi.
- b. Stabil dan andal: Digunakan secara luas oleh perusahaan besar dan komunitas pengembang.
- c. Gratis dan open source: Meskipun tersedia versi komersial, versi komunitas MySQL dapat digunakan secara bebas.
- d. Komunitas luas: Didukung oleh komunitas global yang aktif dan dokumentasi yang lengkap.



**Gambar 2.6 My SQL**  
(Sumber : Oracle, 2024)

MySQL digunakan dalam penelitian ini sebagai sistem pengelolaan basis data karena memiliki antarmuka yang sederhana, ringan, dan mudah digunakan. Sistem ini mendukung pengelolaan berbagai jenis basis data, termasuk MariaDB dan PostgreSQL, sehingga memudahkan proses pembuatan, pengelolaan, serta manipulasi data. Selain itu, MySQL menyediakan fitur pendukung seperti *query editor*, fasilitas ekspor dan impor data, serta pengaturan pengguna dan hak akses yang efisien. Dengan pemanfaatan MySQL, pengelolaan basis data dapat dilakukan secara lebih terstruktur dan terorganisasi.

#### **2.2.8 Hyper Text Markup Language**

Berdasarkan Gambar 2.7, HTML (*HyperText Markup Language*) merupakan bahasa standar yang digunakan dalam pengembangan halaman web untuk menampilkan berbagai informasi melalui peramban internet. HTML juga memungkinkan keterhubungan antarberkas, baik yang berada pada komputer lokal melalui *localhost* maupun yang terdapat pada situs lain di internet. Secara umum, HTML berfungsi untuk menyusun, mengatur, dan menyajikan data serta informasi

agar dokumen web dapat diakses dan ditampilkan dengan baik melalui layanan web (Lengkong, 2019).



**Gambar 2.7 HTML**  
(Sumber : W3C, 2024)

Pada penelitian ini, HTML dimanfaatkan sebagai bahasa utama dalam perancangan antarmuka pengguna (*user interface*) pada sistem yang dikembangkan. HTML digunakan untuk membangun struktur serta menyusun konten halaman web, meliputi teks, gambar, tabel, dan berbagai elemen lainnya. Dengan sifatnya yang fleksibel dan kompatibel dengan beragam teknologi web, HTML dapat diintegrasikan secara optimal dengan CSS untuk pengaturan tampilan serta JavaScript guna meningkatkan tingkat interaktivitas sistem. Penerapan HTML diharapkan mampu menghasilkan antarmuka yang mudah dipahami, responsif, dan nyaman digunakan oleh pengguna.

### **2.2.9 Cascading Style Sheet**

CSS (*Cascading Style Sheets*), sebagaimana ditunjukkan pada Gambar 2.8, merupakan bahasa desain web (*style sheet language*) yang digunakan untuk mengatur tampilan dan format halaman web yang ditulis menggunakan bahasa markup. CSS umumnya digunakan untuk mendesain halaman HTML dan XHTML, namun saat ini juga dapat diterapkan pada berbagai jenis dokumen XML, termasuk SVG, XUL, hingga platform Android.

CSS dirancang untuk memisahkan konten utama dari aspek tampilan dokumen, seperti tata letak (*layout*), warna, dan jenis huruf. Pemisahan ini bertujuan untuk meningkatkan aksesibilitas konten, memberikan fleksibilitas dan kontrol yang lebih besar terhadap tampilan halaman, serta mengurangi kompleksitas dalam penulisan kode dan struktur konten. Selain itu, CSS mendukung berbagai media tampilan, seperti layar (*on-screen*), cetak (*in-print*), dan suara (*by voice*), serta memungkinkan pengelolaan gaya tampilan secara terpusat melalui file CSS yang terhubung dengan konten web.

Tujuan utama penggunaan CSS adalah untuk memisahkan struktur konten dari desain visual halaman web, sehingga proses pengembangan maupun pemeliharaan website dapat dilakukan dengan lebih mudah dan efisien. Unsur-unsur desain web seperti warna, ukuran, dan format tampilan dapat diatur secara konsisten. Dengan adanya CSS, tampilan yang sama dapat digunakan kembali pada berbagai halaman web, sehingga mempermudah pengembangan halaman dalam jumlah besar dan dapat menghemat waktu pembuatan website (Lengkong, 2019).



**Gambar 2.8 CSS**  
(Sumber : Duggal, 2022)

Dalam penelitian ini, CSS digunakan sebagai alat untuk mengelola tampilan visual dan desain antarmuka pengguna (*user interface*) pada sistem yang dibangun. Pemanfaatan CSS memungkinkan pengaturan elemen HTML secara lebih terstruktur, mulai dari penyesuaian warna, jenis dan ukuran huruf, hingga pengaturan jarak (*margin* dan *padding*) serta tata letak halaman agar tampil lebih rapi dan responsif. Selain itu, CSS menyediakan fitur *media query* yang mendukung penyesuaian tampilan pada berbagai ukuran layar, baik pada perangkat desktop maupun mobile. Dengan penerapan CSS, antarmuka sistem diharapkan mampu memberikan pengalaman pengguna (*user experience*) yang lebih baik melalui tampilan yang konsisten, menarik, dan mudah dioperasikan.

### **2.2.10 JavaScript**

JavaScript merupakan bahasa pemrograman web yang dijalankan di sisi klien (*client-side*), sebagaimana ditunjukkan pada Gambar 2.9. Bahasa ini dieksekusi pada aplikasi klien berupa peramban web, seperti Google Chrome, Mozilla Firefox, dan Opera Mini. JavaScript mulai dikembangkan pada pertengahan tahun 1990-an dan meskipun memiliki nama yang serupa, JavaScript berbeda dengan bahasa pemrograman Java. JavaScript dapat ditulis secara langsung di dalam dokumen HTML maupun disimpan dalam berkas terpisah yang kemudian dihubungkan dengan halaman web terkait. Bahasa ini menyediakan berbagai fitur yang

memungkinkan pengaturan dan pengendalian interaksi pengguna dengan halaman web, sehingga mendukung terciptanya aplikasi web yang lebih dinamis dan interaktif (Sari et al., 2022).



**Gambar 2.9 Java Script**  
(Sumber : Bahtiar, 2019)

Pada penelitian ini, Java digunakan sebagai bahasa pemrograman dalam pengembangan sistem karena memiliki tingkat kestabilan yang baik, bersifat *platform-independent*, serta mendukung paradigma *Object-Oriented Programming* (OOP). Melalui dukungan Java Virtual Machine (JVM), aplikasi yang dibangun dapat dijalankan pada berbagai sistem operasi tanpa perlu dilakukan penyesuaian kode. Selain itu, Java menyediakan ekosistem library dan framework yang luas, disertai dengan fitur keamanan serta kemampuan skalabilitas yang memadai. Pemanfaatan Java dalam penelitian ini diharapkan dapat menghasilkan sistem yang handal, fleksibel, dan mudah dikembangkan untuk kebutuhan di masa mendatang.

### **2.2.11 System Usability Scale**

Metode *System Usability Scale* (SUS) merupakan salah satu pendekatan yang digunakan untuk melakukan pengujian tingkat kebergunaan (*usability*) suatu aplikasi. SUS dikembangkan sebagai alat pengukuran *usability* yang bersifat cepat dan sederhana (*quick and dirty*). Instrumen ini berupa kuesioner yang digunakan untuk menilai tingkat *usability* sistem komputer berdasarkan persepsi subjektif pengguna. SUS pertama kali diperkenalkan oleh John Brooke pada tahun 1986 dan terdiri dari 10 butir pertanyaan. Hingga saat ini, SUS banyak digunakan karena memiliki sejumlah keunggulan, antara lain mudah digunakan, menghasilkan skor dalam rentang 0 hingga 100, tidak memerlukan perhitungan yang kompleks, tersedia secara gratis, serta telah terbukti valid dan reliabel meskipun digunakan dengan jumlah responden yang relatif kecil. Skala penilaian yang digunakan dalam

kuesioner SUS adalah skala Likert dengan rentang nilai 1 (sangat tidak setuju) hingga 5 (sangat setuju) (Frobenius et al., 2023).

*System Usability Scale (SUS)* menggunakan lima tingkat skala Likert dalam proses perhitungannya. Responden diminta untuk memberikan penilaian terhadap sepuluh pernyataan sesuai dengan persepsi mereka. Setiap item pernyataan memiliki skor kontribusi yang berkisar antara 0 hingga 4. Untuk pernyataan bernomor ganjil (1, 3, 5, 7, dan 9), skor kontribusi diperoleh dengan mengurangi nilai skala dengan angka 1, sedangkan untuk pernyataan bernomor genap (2, 4, 6, 8, dan 10), skor kontribusi diperoleh dengan mengurangi nilai skala dari angka 5. Selanjutnya, total skor kontribusi dikalikan dengan 2,5 sehingga menghasilkan nilai akhir SUS dalam rentang 0 hingga 100 (Sembodo et al., 2021).

**Tabel 2.2 Skor SUS**

Nilai	Skor	Keterangan
>81	A	Excellent
68-81	B	Good
68	C	OK/Fair
51-67	D	Poor
<51	E	Worst

(Sembodo et al., 2021)

### 2.2.12 *Blackbox Testing*

Blackbox Testing merupakan salah satu metode pengujian perangkat lunak yang berfokus pada fungsionalitas aplikasi tanpa memperhatikan struktur internal kode program. Metode ini digunakan untuk memastikan bahwa setiap fungsi dalam sistem dapat berjalan sesuai dengan kebutuhan dan harapan pengguna. Pada pengujian blackbox, tester hanya memberikan input tertentu pada sistem kemudian mengevaluasi apakah output yang dihasilkan sesuai dengan yang diharapkan. Keunggulan dari metode ini adalah dapat dilakukan tanpa perlu mengetahui detail teknis pemrograman sehingga lebih menekankan pada kesesuaian sistem dengan spesifikasi kebutuhan pengguna. Blackbox Testing umumnya digunakan untuk menguji validasi input, navigasi menu, integrasi antar modul, serta kesesuaian hasil keluaran sistem. Dengan demikian, metode ini sangat berguna dalam memastikan bahwa aplikasi telah memenuhi standar kualitas dari sisi fungsionalitas (Pressman & Maxim, 2019)

### **2.2.13 UAT (*User Acceptance Test*)**

*User Acceptance Test (UAT)* merupakan tahap akhir dalam proses pengujian perangkat lunak yang bertujuan untuk memastikan bahwa sistem yang telah dikembangkan benar-benar memenuhi kebutuhan dan harapan pengguna akhir. Pengujian ini dilakukan oleh pengguna atau perwakilan pengguna untuk menilai apakah sistem sudah berfungsi sesuai dengan spesifikasi kebutuhan bisnis dan siap untuk digunakan dalam lingkungan operasional sebenarnya.

Pada tahap UAT, fokus pengujian tidak hanya pada aspek teknis, tetapi juga pada kegunaan (*usability*), kelayakan fungsi (*functional suitability*), serta kemudahan penggunaan sistem. Hasil dari UAT menjadi dasar untuk menentukan apakah sistem dapat diterima (*accepted*) atau memerlukan perbaikan sebelum implementasi penuh dilakukan.

Dengan demikian, UAT berperan penting dalam menjembatani proses antara pengembang dan pengguna, memastikan bahwa aplikasi tidak hanya berjalan dengan benar secara teknis, tetapi juga memberikan nilai guna sesuai kebutuhan pengguna (Sommerville, 2020).