

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. Rancang Bangun

Rancang bangun merupakan proses merancang dan mengembangkan suatu sistem atau aplikasi yang sebelumnya belum tersedia pada suatu instansi maupun objek tertentu. Selain itu, rancang bangun juga dapat diartikan sebagai tahap implementasi dari hasil analisis kebutuhan yang diwujudkan menjadi sebuah sistem atau aplikasi yang dapat diterapkan untuk memenuhi kebutuhan pengguna atau instansi yang memerlukannya (Lukman, dkk, 2023:130). Rancang bangun merupakan rangkaian proses yang terdiri atas dua tahapan utama, yaitu perancangan dan pembangunan sistem. Tahap perancangan dilakukan dengan menganalisis, mengidentifikasi, serta mendokumentasikan kebutuhan pengguna yang kemudian dituangkan ke dalam bentuk desain atau model sistem. Selanjutnya, tahap pembangunan dilakukan dengan mengimplementasikan rancangan tersebut menjadi sebuah perangkat lunak yang dapat digunakan sesuai dengan kebutuhan. Oleh karena itu, rancang bangun dapat diartikan sebagai proses mengubah hasil analisis kebutuhan menjadi suatu sistem atau aplikasi yang siap untuk diterapkan (Firmansyah, dkk, 2025:100).

Rancang bangun merupakan kegiatan mengubah hasil analisis kebutuhan menjadi bentuk perangkat lunak yang dapat

diimplementasikan dalam suatu sistem. Proses ini meliputi serangkaian tahapan untuk menerjemahkan hasil analisis ke dalam bahasa pemrograman sehingga menghasilkan sistem baru atau melakukan pengembangan serta perbaikan terhadap sistem yang telah ada agar dapat berfungsi sesuai dengan kebutuhan pengguna (Talia, dkk, 2024:1369). Rancang bangun merupakan proses mengimplementasikan hasil analisis kebutuhan ke dalam bentuk perangkat lunak yang dapat diterapkan pada suatu sistem. Proses ini mencakup serangkaian tahapan yang dilakukan untuk menerjemahkan hasil analisis ke dalam bahasa pemrograman sehingga dapat menghasilkan sistem baru maupun mengembangkan dan menyempurnakan sistem yang telah ada agar sesuai dengan kebutuhan pengguna (Syahputra, dkk, 2025:1912).

Berdasarkan kutipan-kutipan diatas, dapat disimpulkan bahwa rancang bangun merupakan suatu proses perancangan dan pembangunan sistem yang dilakukan dengan menerjemahkan hasil analisis kebutuhan ke dalam bentuk perangkat lunak atau sistem yang dapat digunakan. Proses ini meliputi tahap perencanaan, penggambaran, penyusunan, hingga implementasi sistem sehingga menghasilkan suatu aplikasi baru atau penyempurnaan dari sistem yang telah ada agar dapat berfungsi sesuai dengan kebutuhan pengguna atau instansi terkait.

2. Sistem Informasi

Sistem berasal dari bahasa Latin *Systema* dan bahasa Yunani *Sustema*, yang diartikan sebagai suatu kesatuan yang tersusun atas

berbagai komponen atau elemen yang saling berhubungan untuk mendukung aliran informasi, materi, maupun energi (Syah, dkk, 2025:202). Selain itu, sistem juga dapat dipahami sebagai sekumpulan elemen yang saling berinteraksi dan bekerja sama dalam mengolah masukan (*input*) menjadi keluaran (*output*) guna mencapai tujuan yang telah ditetapkan (Syah, dkk, 2025:203).

Sistem merupakan suatu kesatuan yang dibentuk oleh berbagai komponen atau subsistem yang saling terintegrasi dan tersusun secara teratur. Setiap komponen memiliki keterkaitan serta bekerja sama untuk mencapai tujuan yang sama, sedangkan subsistem merupakan bagian dari sistem utama yang menjalankan fungsi tertentu dalam mendukung kinerja keseluruhan sistem (Negara, dkk, 2021:1).

Informasi merupakan data yang telah melalui proses pengolahan sehingga memiliki makna dan nilai guna bagi pengguna. Informasi tersebut dapat dimanfaatkan sebagai dasar dalam proses pengambilan keputusan, baik untuk kebutuhan saat ini maupun sebagai pertimbangan dalam menentukan kebijakan di masa mendatang. Dengan demikian, informasi merupakan hasil pengolahan data yang menghasilkan nilai dan manfaat bagi penggunanya (Syah, dkk, 2025:203).

Sistem informasi merupakan suatu sistem yang diterapkan dalam organisasi untuk mendukung proses pengolahan transaksi, menunjang kegiatan operasional, membantu fungsi manajerial, serta mendukung aktivitas strategis organisasi. Selain itu, sistem informasi juga berperan

dalam menyediakan informasi yang dibutuhkan oleh pihak-pihak terkait sebagai dasar dalam pelaksanaan aktivitas dan pengambilan keputusan.

Sistem informasi dapat dijelaskan sebagai berikut:

- a. Suatu sistem yang dibangun oleh manusia dan tersusun atas berbagai komponen dalam organisasi yang saling berinteraksi dengan tujuan menghasilkan, mengolah, dan menyajikan informasi yang bermanfaat bagi pengguna.
- b. Sekumpulan prosedur yang diterapkan dalam suatu organisasi untuk menghasilkan informasi yang digunakan sebagai dasar pengambilan keputusan maupun pengendalian organisasi.
- c. Suatu sistem yang diterapkan dalam organisasi untuk mengelola proses transaksi, mendukung pelaksanaan kegiatan operasional dan manajerial, menunjang aktivitas strategis, serta menyediakan berbagai laporan yang diperlukan oleh pihak-pihak yang berkepentingan (Rauba, dkk, 2025:37).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa sistem informasi merupakan suatu kesatuan yang terdiri atas berbagai komponen yang saling terhubung dan bekerja secara terpadu dalam mengolah data menjadi informasi yang bermanfaat. Informasi yang dihasilkan digunakan untuk mendukung pelaksanaan kegiatan operasional, fungsi manajerial, serta pengambilan keputusan, sehingga dapat membantu organisasi dalam mencapai tujuan secara lebih efektif dan efisien.

3. Surat Pernyataan Ahli Waris (SPAW)

Berdasarkan UU No. 23 Tahun 2014, Kecamatan adalah perangkat daerah kabupaten/kota berwujud kewilayahan yang dipimpin Camat. Kecamatan bertugas melaksanakan urusan pemerintahan konkuren yang didelegasikan oleh bupati/wali kota, menyelenggarakan tugas umum pemerintahan, serta tugas pembantuan. Termasuk pada wilayah Kartoharjo Madiun, bahwa kehadiran kantor Kecamatan sebagai unit pemerintahan yang berperan langsung dalam memberikan pelayanan publik memiliki tanggung jawab besar dalam menjamin kepastian, kecepatan, dan ketepatan layanan administrasi kepada masyarakat. Salah satu bentuk layanan yang cukup penting dan sering diminta masyarakat adalah penerbitan Surat Pernyataan Ahli Waris (SPAW). Surat ini menjadi syarat utama dalam proses pengalihan hak atas tanah, pencairan dana warisan, serta pengelolaan aset yang ditinggalkan oleh pewaris.

Dasar hukum utama penetapan ahli waris bagi umat Islam di Indonesia adalah Pasal 49 huruf b UU No. 3 Tahun 2006 tentang Perubahan atas UU No. 7 Tahun 1989 tentang Peradilan Agama. Pengadilan Agama berwenang menetapkan ahli waris, harta peninggalan, dan pembagiannya berdasarkan hukum Islam. Undang-Undang Nomor 25 Tahun 2009 merupakan dasar hukum untuk mengatur pelayanan publik. Undang-Undang tersebut sebagai bagian dari perwujudan reformasi birokrasi di Indonesia serta ketentuan tentang prinsip-prinsip Kepemerintahan yang baik (*Good Governance*). Dalam Undang-Undang

No. 25 Tahun 2009 tentang Pelayanan Publik tersebut juga menegaskan mengenai ketentuan dan kewajiban yang diberikan kepada entitas pemerintah untuk mengadopsi sistem elektronik. Undang-Undang tersebut sebagai penegas bahwa para pelaku pelayanan publik perlu beradaptasi dengan penggunaan trend teknologi dan informasi komunikasi terkini (Rahayu, dkk, 2025:4202).

Berdasarkan kutipan-kutipan diatas, dapat disimpulkan bahwa kecamatan merupakan salah satu perangkat daerah yang berperan penting dalam pelaksanaan pemerintahan serta penyediaan pelayanan publik kepada masyarakat. Salah satu pelayanan yang diberikan adalah penerbitan Surat Pernyataan Ahli Waris (SPAW) yang memiliki fungsi penting dalam pengurusan hak waris dan administrasi aset peninggalan pewaris. Dalam pelaksanaannya, pelayanan tersebut didukung oleh dasar hukum yang mengatur kewenangan penetapan ahli waris serta penerapan pelayanan publik berbasis elektronik. Dengan demikian, implementasi sistem informasi berbasis teknologi dalam pelayanan publik merupakan salah satu upaya untuk meningkatkan kualitas layanan agar lebih cepat, tepat, efektif, dan efisien, sekaligus mendukung terwujudnya tata kelola pemerintahan yang baik (*Good Governance*).

4. Website

Website merupakan sekumpulan halaman *web* yang berada dalam satu domain dan digunakan sebagai media untuk menyajikan berbagai informasi yang dapat diakses melalui jaringan internet. Secara etimologis,

website merupakan kumpulan halaman yang tersusun dalam suatu domain atau subdomain pada *World Wide Web* (WWW). Umumnya, setiap halaman *web* dikembangkan menggunakan HTML (*HyperText Markup Language*) dan diakses melalui protokol HTTP (*HyperText Transfer Protocol*) yang berfungsi mengirimkan data dari *server* ke *web browser* pengguna. Keseluruhan halaman tersebut saling terhubung sehingga membentuk suatu jaringan informasi yang dapat diakses secara luas melalui internet (Syah, dkk, 2025:203).

Website merupakan kumpulan halaman yang saling terintegrasi dan dapat diakses oleh pengguna di berbagai lokasi selama terhubung dengan jaringan internet. Setiap *website* memiliki alamat unik yang dikenal sebagai URL (*Uniform Resource Locator*) sebagai identitas untuk mengaksesnya. Selain menyajikan informasi dalam bentuk teks, gambar, audio, dan video, *website* juga menyediakan berbagai fitur interaktif, seperti formulir, kolom komentar, maupun layanan percakapan (*chat*). Dalam proses pengembangannya, *website* terdiri atas dua komponen utama, yaitu *client-side* dan *server-side*. Komponen *client-side* merupakan bagian yang berinteraksi langsung dengan pengguna melalui *web browser* dan umumnya dibangun menggunakan teknologi HTML, CSS, serta *JavaScript* (Sukri, dkk, 2023:1).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa *website* merupakan kumpulan halaman *web* yang saling terhubung dalam suatu domain dan dapat diakses melalui jaringan internet menggunakan

web browser. *Website* berfungsi sebagai media penyampaian informasi sekaligus sarana interaksi pengguna dengan menyajikan berbagai jenis konten, seperti teks, gambar, audio, video, serta fitur-fitur interaktif. Pembangunan *website* memanfaatkan teknologi *client-side* dan *server-side* sehingga mampu memberikan akses informasi yang cepat, mudah, dan dapat digunakan secara daring sesuai kebutuhan pengguna.

5. **PHP Hypertext Preprocessor**

PHP digunakan sebagai bahasa pemrograman utama pada sisi *server (backend)*. PHP dipilih karena memiliki performa yang baik, bersifat *open source*, dan didukung oleh komunitas yang besar. Selain itu, integrasi PHP dengan berbagai *framework*, seperti Laravel, memungkinkan pengembangan aplikasi berbasis *web* yang lebih cepat, terstruktur, dan aman (Alajuri, dkk, 2025:5811). PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang memiliki karakteristik serupa dengan *JavaScript* dan *Python*. Perbedaannya terletak pada penggunaannya, di mana PHP umumnya digunakan untuk pengembangan aplikasi pada sisi *server (server-side)*, sedangkan *JavaScript* dapat diterapkan pada sisi *frontend* maupun *backend* (Agung, dkk, 2022:322).

PHP (*Hypertext Preprocessor*) merupakan salah satu bahasa pemrograman skrip yang dikembangkan dengan mengadopsi konsep dari beberapa bahasa pemrograman, seperti *C*, *Java*, dan *Perl*, sehingga mudah dipahami dan dipelajari oleh pengembang. Dalam pengembangan

aplikasi berbasis *web*, PHP berfungsi sebagai bahasa pemrograman sisi *server* (*server-side scripting*) yang digunakan untuk membangun halaman *web* dinamis serta mengelola interaksi dengan basis data. Selain itu, PHP memiliki keunggulan dalam mendukung integrasi dengan berbagai sistem manajemen basis data, termasuk MySQL, sehingga menjadi salah satu bahasa pemrograman yang fleksibel, andal, dan banyak dimanfaatkan dalam pengembangan aplikasi *web* berbasis data (Nova, 2025:2249).

Berdasarkan kutipan-kutipan diatas, dapat disimpulkan bahwa PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman *server-side* yang banyak digunakan untuk membangun dan mengembangkan aplikasi berbasis *web* karena kemampuannya dalam mengelola proses pada sisi *server* secara efisien. PHP memiliki keunggulan berupa sifat *open source*, mudah dipelajari, memiliki performa yang baik, serta mampu terintegrasi dengan berbagai basis data dan *framework* seperti Laravel. Dengan kemampuan tersebut, PHP dapat digunakan untuk membangun aplikasi *web* yang dinamis, terstruktur, aman, dan efisien dalam pengolahan data serta penyampaian informasi kepada pengguna.

6. Laravel

Laravel adalah salah satu *framework* pengembangan *web* yang populer, ditulis dalam bahasa pemrograman PHP. Diperkenalkan pertama kali oleh Taylor Otwell pada tahun 2011, Laravel telah menjadi pilihan utama banyak pengembang dalam membangun aplikasi *web* modern.

Laravel didasarkan pada beberapa konsep dan praktik terbaik dalam pengembangan perangkat lunak, seperti pola desain MVC (*Model- View-Controller*), ORM (*Object-Relational Mapping*), routing yang kuat, dan banyak lagi. Ini menyederhanakan proses pengembangan aplikasi web dengan menyediakan struktur yang jelas dan konsisten. Sejak diluncurkan, Laravel telah mengalami pertumbuhan yang pesat dan menjadi salah satu *framework* PHP paling populer di dunia. Ini terus berkembang dengan versi baru yang diperbarui secara teratur, menyediakan fitur-fitur baru, perbaikan bug, dan peningkatan kinerja. Laravel telah menciptakan ekosistem yang kuat di sekitarnya, dengan komunitas yang aktif dan beragam sumber daya seperti dokumentasi resmi yang kaya, paket tambahan (*packages*) yang dapat diintegrasikan, forum diskusi, dan banyak tutorial dan sumber belajar lainnya (Kurniawan, dkk, 2023:179).

Laravel adalah *framework* PHP yang digunakan untuk pengembangan sisi *backend*. *Framework* ini dipilih karena menawarkan arsitektur MVC (*Model-View-Controller*) yang terstruktur, mendukung keamanan yang lebih baik (misalnya, *CSRF protection*, *hashing*), dan menyediakan berbagai fitur bawaan seperti *routing*, *middleware*, dan migrasi *database*. Dengan menggunakan Laravel, pengelolaan logika aplikasi menjadi lebih terorganisasi dan mudah untuk diintegrasikan dengan layanan lain, seperti API atau *library* eksternal (Alajuri, dkk, 2025:5812). Laravel mengadopsi arsitektur *Model View Controller*

(MVC) yang memisahkan fungsi logika bisnis, tampilan antarmuka pengguna, dan pengelolaan data ke dalam komponen yang terpisah. Pendekatan ini mempermudah proses pengembangan, pemeliharaan, serta pengembangan sistem di masa mendatang karena struktur aplikasi menjadi lebih sistematis dan terorganisasi. Selain itu, Laravel menyediakan berbagai fitur pendukung, seperti *middleware*, mekanisme autentikasi berbasis peran (*role-based access control*), serta *Blade* sebagai *template engine* yang mempermudah pembuatan antarmuka pengguna. Beragam fitur tersebut menjadikan Laravel sebagai salah satu *framework* yang tepat untuk mengembangkan sistem informasi yang kompleks, aman, dan mudah dikelola (Simarmata, dkk, 2025:4534).

Laravel merupakan salah satu *framework* PHP yang bersifat *open source* sehingga dapat digunakan secara bebas oleh para pengembang. *Framework* ini dikembangkan oleh Taylor Otwell dan pertama kali diperkenalkan pada tahun 2011. Sejak dirilis, Laravel mengalami perkembangan yang sangat pesat dan berhasil menjadi salah satu *framework* PHP yang banyak diminati oleh pengembang aplikasi *web* di berbagai negara. Popularitas Laravel terus meningkat karena menawarkan berbagai kemudahan dan fitur yang mendukung proses pengembangan aplikasi secara lebih cepat, terstruktur, dan efisien.

Laravel menjadi salah satu *framework* yang banyak digunakan karena memiliki berbagai keunggulan dibandingkan *framework* PHP lainnya. Beberapa keunggulan tersebut antara lain sebagai berikut:

- a. Laravel menyediakan berbagai fitur bawaan, seperti *Artisan*, *Migration*, *Blade*, dan fitur lainnya yang mendukung proses pengembangan aplikasi secara lebih efisien.
- b. Laravel memiliki sintaks yang sederhana, ekspresif, dan mudah dipahami sehingga memudahkan pengembang, termasuk pemula, dalam mempelajari serta mengimplementasikan kode program.
- c. Laravel didukung oleh dokumentasi resmi yang lengkap dan terstruktur, mulai dari proses instalasi hingga penjelasan penggunaan setiap fitur pada setiap versi yang dirilis.
- d. Laravel memiliki komunitas pengguna yang sangat luas sehingga tersedia banyak pustaka (*library*) dan paket (*package*) yang dapat dimanfaatkan untuk mempercepat proses pengembangan aplikasi.
- e. Laravel terintegrasi dengan *Composer* sebagai *dependency manager*, sehingga memudahkan pengembang dalam mengelola, mengunduh, dan memasang berbagai *library* yang dibutuhkan secara otomatis melalui internet.
- f. Laravel menyediakan *Blade* sebagai *template engine* yang mempermudah pengembang dalam membangun tampilan antarmuka (*user interface*) serta menyajikan data pada halaman HTML secara lebih efektif (Abdulloh, 2022:448).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa Laravel merupakan *framework* PHP yang mengimplementasikan arsitektur *Model-View-Controller* (MVC) sehingga proses

pengembangan aplikasi web dapat dilakukan secara lebih terstruktur, efisien, aman, serta mudah dipelihara dan dikembangkan sesuai kebutuhan. Laravel dilengkapi dengan berbagai fitur bawaan, seperti *routing*, *middleware*, *Object Relational Mapping (ORM)*, migrasi basis data, autentikasi, serta *Blade template engine* yang mempermudah proses pengembangan, pengelolaan, dan pemeliharaan aplikasi. Selain itu, Laravel merupakan *framework open source* yang didukung oleh komunitas pengembang yang luas, dokumentasi yang lengkap, serta kemampuan integrasi dengan berbagai *library* dan layanan lainnya, sehingga menjadi salah satu *framework* PHP yang banyak digunakan dalam pengembangan sistem informasi berbasis *web*.

7. Bootstrap

Bootstrap merupakan *framework front-end* yang dirancang untuk memudahkan pengembang dalam membangun *website* yang responsif dengan pendekatan *mobile-first*. *Framework* ini menyediakan berbagai komponen antarmuka yang siap digunakan, seperti *grid system*, formulir (*form*), tombol, navigasi, dan elemen lainnya sehingga proses pengembangan tampilan *website* dapat dilakukan secara lebih cepat, konsisten, dan efisien (Masikhoh, dkk, 2025:18). Bootstrap merupakan *framework* CSS yang digunakan pada bagian *frontend* untuk mengembangkan tampilan antarmuka pengguna (*User Interface/UI*) agar lebih terstruktur, responsif, dan mudah digunakan. *Framework* ini menyediakan komponen bawaan seperti tombol, formulir, navigasi, dan

grid *system* yang responsif, sehingga mempermudah proses pengembangan antarmuka tanpa harus mendesain dari awal. Dengan Bootstrap, aplikasi dapat diakses dengan tampilan yang optimal di berbagai perangkat (desktop, tablet, dan smartphone) (Alajuri, dkk, 2025:5812).

Bootstrap merupakan *framework* CSS yang digunakan untuk membangun antarmuka *website* yang responsif dan memiliki tampilan yang konsisten pada berbagai jenis perangkat. *Framework* ini banyak digunakan oleh pengembang *web* karena menyediakan berbagai komponen antarmuka yang siap pakai serta mudah diimplementasikan. Dengan memanfaatkan Bootstrap, pengembangan sistem informasi berbasis *web* dapat menghasilkan tampilan yang lebih *user-friendly* dan mampu beradaptasi dengan berbagai ukuran layar, seperti komputer, tablet, maupun *smartphone*, sehingga meningkatkan kenyamanan pengguna dalam mengakses sistem (Hilmi, dkk, 2025:12). Bootstrap adalah *framework front-end* yang digunakan untuk membangun tampilan antarmuka pengguna yang responsif dan konsisten di berbagai perangkat. Bootstrap meningkatkan efisiensi dalam menghasilkan tampilan yang lebih menarik secara visual (Mourent dan Tanama, 2025:434).

Berdasarkan kutipan-kutipan diatas, dapat disimpulkan bahwa Bootstrap merupakan *framework front-end* berbasis CSS yang digunakan untuk mengembangkan antarmuka *website* yang responsif, konsisten, dan mudah dioperasikan pada berbagai ukuran perangkat. Dengan

menyediakan berbagai komponen bawaan, seperti *grid system*, tombol, formulir, navigasi, dan elemen antarmuka lainnya, Bootstrap mampu mempercepat proses pengembangan tampilan *website* sekaligus menghasilkan antarmuka yang lebih terstruktur dan mudah dipelihara. Selain itu, penggunaan Bootstrap mampu meningkatkan efisiensi pengembangan dan menghasilkan desain antarmuka yang lebih menarik, *user-friendly*, dan optimal pada desktop, tablet, maupun *smartphone*.

8. Sistem Basis Data

Database merupakan sekumpulan data yang disimpan secara terorganisasi di dalam sistem komputer sehingga dapat diakses, dikelola, dimanipulasi, dan dipelihara secara efisien melalui perangkat lunak pengelola basis data sesuai dengan kebutuhan pengguna. Secara konsep, *database* terdiri atas kumpulan data yang saling berhubungan dan tersusun dalam bentuk *file* sehingga mampu menghasilkan informasi baru sesuai dengan kebutuhan. Manajemen basis data merupakan proses pengelolaan sekumpulan data atau fakta yang saling berkaitan agar dapat disimpan, dimanipulasi, maupun diakses kembali oleh pengguna secara efektif (Putra, dkk, 2025:207).

Secara etimologis, istilah basis data (*database*) berasal dari kata *base* yang berarti basis dan *data* yang mengacu pada fakta atau informasi mengenai suatu objek di dunia nyata yang dapat direkam, disimpan, dan dikelola dalam media komputer sebagai sumber informasi. Basis data merupakan kumpulan data yang tersusun secara terorganisasi

berdasarkan hubungan logis sehingga membentuk suatu kesatuan yang dapat menggambarkan aktivitas dalam suatu organisasi atau lingkungan tertentu. Sementara itu, data merupakan representasi dari objek maupun peristiwa yang memiliki makna dalam ruang lingkup pengguna (Dale, dkk, 2024:18).

Sistem basis data adalah sistem berbasis komputer yang dirancang untuk menyimpan, mengelola, dan memelihara data sehingga dapat diolah menjadi informasi yang tersedia ketika dibutuhkan. *Database* merupakan kumpulan data yang disimpan secara elektronik dan diorganisasikan sedemikian rupa sehingga memungkinkan proses penyimpanan, pengelolaan, pencarian, serta pengaksesan data dilakukan secara cepat dan efisien. Pengelolaan *database* dilakukan menggunakan perangkat lunak yang dikenal sebagai *Database Management System* (DBMS) adalah perangkat lunak yang berfungsi untuk mengelola basis data dengan menyediakan berbagai fasilitas, seperti penyimpanan, pengaturan, pemrosesan, serta manipulasi data, sehingga data dapat diakses dan digunakan secara efektif sesuai dengan kebutuhan pengguna. Selain mendukung efisiensi dalam pengelolaan informasi, sistem basis data juga berperan penting dalam menunjang proses pengambilan keputusan pada berbagai bidang aplikasi. Berbagai jenis *database* yang umum digunakan antara lain MySQL, PostgreSQL, serta *database* NoSQL seperti MongoDB dan Cassandra (Nova, 2025:2249).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa database atau basis data merupakan sekumpulan data yang tersusun secara sistematis, saling berhubungan, dan disimpan dalam media komputer sehingga dapat diakses, dikelola, diperbarui, serta diolah sesuai dengan kebutuhan pengguna. Keberadaan basis data memudahkan proses penyimpanan, pencarian, dan pengelolaan informasi secara efektif dan efisien untuk mendukung berbagai aktivitas dalam suatu sistem informasi. Basis data berfungsi sebagai media penyimpanan informasi yang mendukung efisiensi pengelolaan data serta mempermudah penyediaan informasi secara cepat dan akurat. Pengelolaan basis data dilakukan menggunakan sistem manajemen basis data (DBMS) sehingga data dapat digunakan secara efektif dalam mendukung berbagai aktivitas dan pengambilan keputusan.

9. PostgreSQL

PostgreSQL merupakan salah satu sistem manajemen basis data relasional *open-source* yang menonjol dalam hal keandalan, kestabilan, serta kepatuhan pada standar SQL. Dibandingkan dengan basis data lain, PostgreSQL dikenal mampu menangani transaksi kompleks dan *volume* data yang besar dengan performa yang tetap konsisten. Penelitian terbaru yang membandingkan kinerja PostgreSQL dengan MySQL menunjukkan bahwa PostgreSQL memiliki keunggulan pada aspek pengelolaan data berskala besar, terutama dalam hal efisiensi dan akurasi *query*. Kemampuan ini membuat PostgreSQL menjadi pilihan tepat sebagai

basis data utama dalam perancangan sistem manajemen risiko rumah sakit berbasis *web*, karena dapat mendukung kebutuhan integrasi, keamanan, serta skalabilitas jangka panjang (Pradana dan Chotijah, 2025:3216).

PostgreSQL merupakan sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*) yang bersifat *open-source* serta memiliki reputasi tinggi dalam hal keandalan dan stabilitas. PostgreSQL juga mendukung implementasi prinsip ACID (*Atomicity, Consistency, Isolation, Durability*) secara menyeluruh, sehingga setiap transaksi data dapat diproses secara konsisten, aman, dan terjamin integritasnya. Selain mampu menangani transaksi yang kompleks, PostgreSQL juga memiliki kemampuan dalam mengeksekusi *query* analitik secara efisien. Dukungannya terhadap pengelolaan data dalam jumlah besar serta skalabilitas yang baik menjadikan PostgreSQL sebagai solusi yang tepat untuk mendukung kebutuhan pengelolaan data dan pengembangan sistem informasi dalam jangka panjang (Simarmata, dkk, 2025:4534).

PostgreSQL merupakan salah satu *Object Relational Database Management System (ORDBMS)* yang bersifat *open-source* dan banyak digunakan sebagai sistem pengelolaan basis data dalam pengembangan aplikasi. PostgreSQL dipilih karena mampu mengelola data secara efektif, mendukung berbagai fitur yang lengkap, serta memiliki kinerja yang andal untuk memenuhi kebutuhan aplikasi berbasis data. PostgreSQL

dikenal memiliki tingkat kestabilan yang tinggi, mudah dikembangkan, serta dilengkapi dengan berbagai fitur, seperti dukungan terhadap tipe data yang kompleks, integritas referensial, dan kemampuan mengeksekusi *query* secara optimal. Fleksibilitas yang dimiliki PostgreSQL memungkinkan pengembang menyesuaikan pengelolaan basis data sesuai dengan kebutuhan sistem, sehingga mampu mendukung pembangunan sistem informasi yang kompleks. Selain itu, PostgreSQL dapat diintegrasikan dengan berbagai bahasa pemrograman, termasuk Golang, untuk menghasilkan aplikasi *web* yang memiliki performa tinggi dan pengelolaan data yang andal (Agustian dan Firmansyah, 2024:5524).

Berdasarkan kutipan-kutipan diatas, dapat disimpulkan bahwa PostgreSQL merupakan sistem manajemen basis data relasional (*Relational Database Management System/RDBMS*) berbasis *open-source* yang dirancang untuk menyediakan pengelolaan data secara andal, stabil, dan mampu menangani peningkatan *volume* data maupun beban kerja secara efisien melalui kemampuan skalabilitas yang dimilikinya. PostgreSQL mendukung standar SQL, transaksi ACID, serta mampu menangani *query* kompleks dan *volume* data besar dengan performa yang konsisten. Selain itu, PostgreSQL memiliki fitur lengkap seperti integritas referensial, dukungan tipe data kompleks, dan fleksibilitas pengembangan sehingga cocok digunakan dalam pembangunan sistem informasi berbasis *web* yang membutuhkan keamanan, efisiensi, serta pengelolaan data yang optimal.

10. *Flowchart*

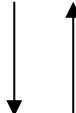
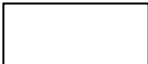
Flowchart sistem merupakan diagram yang digunakan untuk menggambarkan alur atau urutan proses yang terjadi dalam suatu sistem, mulai dari proses masukan (*input*), keluaran (*output*), hingga media penyimpanan data yang digunakan selama proses pengolahan informasi berlangsung. Adapun *flowchart* program merupakan bagan yang memanfaatkan simbol-simbol standar untuk menggambarkan urutan langkah kerja, logika, serta hubungan antarproses atau instruksi dalam suatu program, sehingga memudahkan pemahaman terhadap alur kerja dan proses yang akan diimplementasikan (Irfan, dkk, 2025:176).

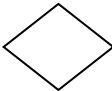
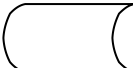
Flowchart merupakan salah satu metode penyajian algoritma dalam bentuk notasi grafis. Diagram ini menggambarkan tahapan atau urutan proses suatu program beserta hubungan antarproses menggunakan simbol-simbol yang telah distandarkan. Setiap simbol memiliki fungsi untuk merepresentasikan proses tertentu, sedangkan garis penghubung menunjukkan aliran atau urutan pelaksanaan proses. Penggunaan *flowchart* dapat membantu dalam menganalisis, memahami, dan memeriksa alur program sehingga meminimalkan kemungkinan adanya langkah yang terlewat selama proses perancangan sistem (Putra, dkk, 2025:208).

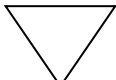
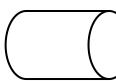
Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa *flowchart* merupakan diagram yang digunakan untuk memvisualisasikan urutan langkah, alur proses, serta hubungan antarproses dalam suatu

sistem maupun program dengan menggunakan simbol-simbol standar. *Flowchart* berperan sebagai alat bantu dalam memahami alur kerja sistem, menggambarkan proses pengolahan data, serta memudahkan analisis dan verifikasi logika program. Dengan demikian, proses perancangan, pengembangan, dan dokumentasi sistem dapat dilakukan secara lebih sistematis, terstruktur, dan mudah dipahami.

Tabel 2.1. Simbol-simbol *Flowchart*

Simbol	Nama	Keterangan
	Arus	Simbol yang berfungsi menghubungkan setiap simbol dalam <i>flowchart</i> sekaligus menunjukkan arah aliran atau urutan proses yang berlangsung. Dengan adanya simbol ini, hubungan antarproses dapat digambarkan secara sistematis sehingga alur kerja sistem menjadi lebih jelas dan mudah dipahami.
	Terminal	Simbol <i>flowchart</i> berfungsi sebagai inisiasi (mulai) atau penutup (berhenti) suatu kegiatan.
	penghubung	Simbol berfungsi untuk memulai dan menyimpulkan atau menghubungkan operasi di dalam lembar atau halaman yang sama.
	baris penghubung	Simbol memberikan tujuan untuk memulai dan mengakhiri atau menghubungkan operasi di beberapa lembar atau halaman.
	Proses	Simbol <i>flowchart</i> digunakan untuk menggambarkan operasi komputasi yang dilakukan oleh komputer atau

personal computer (PC).		
	kegiatan manual	Simbol ini digunakan untuk merepresentasikan aktivitas atau operasi yang dilakukan secara manual oleh pengguna tanpa melibatkan komputer sebagai media pemrosesan. Simbol ini menunjukkan bahwa suatu pekerjaan dilaksanakan secara langsung oleh manusia dalam alur sistem
	keputusan	Simbol tersebut berfungsi untuk memilih suatu prosedur sesuai dengan keadaan saat ini.
	keluar-masuk	Simbol diagram alur berfungsi untuk mewakili proses input dan output, terlepas dari peralatan spesifik yang digunakan.
	Manual Input	Simbol tersebut berfungsi untuk memungkinkan input data secara manual pada keyboard virtual.
	persiapan	Simbol penyimpanan (<i>storage</i>) berfungsi untuk merepresentasikan media atau ruang penyimpanan yang digunakan sebagai lokasi penyimpanan data dalam sistem. Simbol ini menunjukkan bahwa data disimpan, baik sementara maupun permanen, sehingga dapat diakses dan digunakan kembali pada proses pengolahan berikutnya
	proses terdefinisi	Simbol diagram alur melayani tujuan pelaksanaan suatu bagian (subprogram)/prosedur.
	penyimpanan online	Untuk menunjukkan input yang berasal dari media penyimpanan seperti disk

atau disimpan ke media penyimpanan.		
	unit pita magnetik	untuk menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
	kartu plong	Untuk menunjukkan bahwa masukan berasal dari kartu atau bahwa keluarannya ditulis ke dalam kartu.
	dokumen	Untuk menunjukkan bahwa masukan berasal dari dokumen fisik atau keluaran dihasilkan dalam bentuk cetakan.
	penyimpanan offline	Untuk menandakan bahwa informasi di dalam simbol ini akan disimpan.
	magnetik Disk	untuk input atau output data yang memanfaatkan magnetic disk.
	magnetik drum	untuk pemasukan atau keluaran data yang memanfaatkan Magnetic Drum.

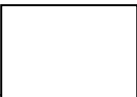
11. DFD (*Data Flow Diagram*)

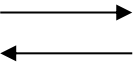
Data Flow Diagram (DFD) merupakan suatu model grafis yang digunakan untuk menggambarkan aliran data dalam suatu sistem. DFD memperlihatkan perpindahan data mulai dari sumber, proses pengolahan, penyimpanan, hingga tujuan akhir data, sehingga dapat memberikan gambaran yang jelas mengenai mekanisme kerja sistem secara keseluruhan. DFD disusun ke dalam beberapa tingkatan (level) sesuai dengan tingkat kompleksitas dan kebutuhan sistem yang akan dianalisis (Darmawan, 2023:284).

Data Flow Diagram (DFD) merupakan teknik pemodelan yang digunakan untuk merepresentasikan komponen-komponen dalam suatu sistem beserta aliran data yang berlangsung di antaranya. DFD menggambarkan perpindahan data mulai dari sumber, melalui proses pengolahan, media penyimpanan, hingga mencapai tujuan akhirnya. Dengan demikian, DFD dapat memberikan gambaran yang lebih jelas, sistematis, dan terstruktur mengenai mekanisme aliran data dalam suatu sistem (Irfan, dkk, 2025:175).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa *Data Flow Diagram* (DFD) merupakan alat pemodelan yang digunakan untuk memvisualisasikan aliran data dalam suatu sistem secara sistematis, mulai dari sumber data, proses pengolahan, media penyimpanan, hingga penyampaian informasi kepada pengguna. DFD membantu memberikan gambaran yang jelas mengenai bagaimana data bergerak dan diproses di dalam sistem sehingga memudahkan proses analisis dan perancangan sistem. DFD membantu dalam memahami hubungan antarproses serta mempermudah analisis dan perancangan sistem informasi. DFD digunakan untuk memodelkan hubungan antar komponen sistem secara terstruktur melalui beberapa level diagram sehingga memudahkan dalam memahami proses kerja dan aliran informasi pada sistem yang dirancang.

Tabel 2.2. Simbol-simbol *Data Flow Diagram*

Simbol	Nama	Keterangan
	<i>External Entity</i>	Entitas yang berada di luar batas sistem, namun memiliki hubungan atau pengaruh terhadap sistem. Entitas ini dapat berupa

		individu, organisasi, maupun sistem lain yang berinteraksi dengan sistem melalui pertukaran data, sehingga berperan sebagai sumber masukan (<i>input</i>) atau penerima keluaran (<i>output</i>)
	Proses	Komponen yang berfungsi mengolah atau mentransformasikan data masukan menjadi informasi atau keluaran yang diinginkan. Dalam pemberian nama proses, umumnya diawali dengan kata kerja yang diikuti oleh objek untuk menggambarkan aktivitas yang dilakukan. Setiap proses harus memiliki data masukan dan keluaran serta dapat berinteraksi dengan <i>External Entity</i> , <i>Data Store</i> , maupun proses lainnya melalui aliran data.
	Alur Data	Jalur yang menunjukkan perpindahan data atau informasi dari satu komponen ke komponen lainnya di dalam sistem. Data yang mengalir dapat berupa data yang diproses oleh komputer, seperti karakter, angka, maupun data lainnya, serta informasi yang berkaitan dengan identitas atau aktivitas tertentu
	<i>Data Store</i>	Komponen yang digunakan untuk menyimpan data atau berkas yang dibutuhkan oleh sistem. Media penyimpanan tersebut dapat berupa penyimpanan digital, seperti <i>hard disk</i> , <i>solid state drive</i> (SSD), atau basis data, maupun penyimpanan non-digital, seperti buku arsip dan dokumen. Dalam <i>Data Flow Diagram</i> (DFD), <i>Data Store</i> hanya dapat dihubungkan dengan komponen Proses melalui Aliran Data (<i>Data Flow</i>) dan tidak dapat dihubungkan secara langsung dengan komponen DFD lainnya

12. ERD (*Entity Relationship Diagram*)

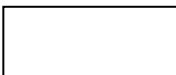
Entity Relationship Diagram (ERD) merupakan model konseptual yang digunakan dalam perancangan basis data untuk menggambarkan entitas, atribut, serta hubungan yang terbentuk antar entitas. Melalui ERD,

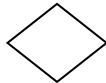
struktur basis data dapat dirancang secara lebih terorganisasi sehingga memudahkan proses implementasi, pengelolaan, dan pengembangan sistem sesuai dengan kebutuhan pengguna (Irfan, dkk, 2025:176).

Entity Relationship Diagram (ERD) merupakan salah satu metode pemodelan data yang digunakan untuk menyusun struktur basis data secara sistematis. ERD memberikan gambaran mengenai kumpulan data beserta hubungan antar data melalui representasi visual, sehingga dapat dijadikan sebagai kerangka kerja dalam membangun basis data yang saling terintegrasi dan mendukung kebutuhan sistem informasi (Widyastuti, dkk, 2024:53).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa *Entity Relationship Diagram* (ERD) merupakan model pemodelan basis data yang digunakan untuk menggambarkan entitas, atribut, serta hubungan antarentitas secara sistematis. ERD berperan sebagai acuan dalam merancang struktur basis data yang terorganisasi sehingga memudahkan proses pengembangan, pengelolaan, dan pemeliharaan sistem informasi.

Tabel 2.3. Simbol-simbol ERD

Simbol	Nama	Keterangan
	Entitas	Objek atau komponen yang menjadi sumber data dalam suatu sistem. Entitas dapat berupa individu, objek, sumber daya, lingkungan, maupun transaksi yang memiliki sejumlah atribut dan datanya akan dikelola dalam basis data

	Hubungan atau Relasi	Hubungan yang menghubungkan dua atau lebih entitas dalam suatu basis data. Relasi berfungsi untuk menunjukkan keterkaitan dan pola interaksi antarentitas sehingga struktur serta hubungan data dalam sistem dapat dimodelkan secara jelas dan terstruktur
	Atribut	Karakteristik atau informasi yang dimiliki oleh suatu entitas. Atribut berfungsi untuk menjelaskan atau memberikan identitas terhadap entitas sehingga setiap data dapat dibedakan berdasarkan karakteristik yang dimilikinya
	Garis Relasi	Simbol yang digunakan untuk menunjukkan adanya keterkaitan atau hubungan antara dua atau lebih entitas dalam <i>Entity Relationship Diagram</i> (ERD). Garis ini menggambarkan bagaimana entitas saling berhubungan sesuai dengan aturan atau kebutuhan yang terdapat dalam sistem.

13. *BlackBox*

Metode pengujian merupakan suatu pendekatan atau teknik yang digunakan untuk mengevaluasi perangkat lunak dengan menentukan data uji yang sesuai sehingga seluruh fungsi sistem dapat diperiksa secara menyeluruh. Pengujian dilakukan untuk memastikan perangkat lunak bekerja sesuai dengan kebutuhan serta memiliki peluang yang tinggi dalam mendeteksi kesalahan atau *error* yang mungkin terjadi (Rauba, dkk, 2025:38).

Black Box Testing merupakan metode pengujian perangkat lunak yang berorientasi pada pengujian fungsionalitas sistem tanpa meninjau atau menganalisis struktur maupun kode program yang digunakan. Pengujian ini bertujuan untuk memastikan bahwa setiap fungsi pada sistem dapat berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan. Selain itu, *Black Box Testing* juga digunakan untuk mengidentifikasi berbagai kemungkinan kesalahan, seperti ketidaksesuaian fungsi, kesalahan pada antarmuka pengguna, permasalahan struktur data, penurunan performa sistem, serta kesalahan yang terjadi pada proses inisialisasi maupun terminasi (Putra, dkk, 2025:208).

Pengujian *Blackbox* ialah suatu pengujian terhadap perangkat lunak berdasarkan fungsionalnya, dan tidak perlu menguji desain atau kode program. Pengujian *Blackbox* dilakukan agar bisa melihat setiap kegunaan dari *input* serta kegunaan *output* pada sistem informasi yang diperlukan, dan untuk mendeteksi kegunaan yang tidak sesuai, antarmuka yang sudah sesuai atau tidak, dan performa yang tidak sesuai (Subawa, dkk, 2025:88).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa metode pengujian merupakan serangkaian teknik yang digunakan untuk mengevaluasi perangkat lunak guna memastikan setiap fungsi sistem bekerja sesuai dengan kebutuhan serta mendeteksi kemungkinan terjadinya kesalahan. Salah satu metode yang umum digunakan adalah

Black Box Testing, yaitu metode pengujian yang menitikberatkan pada pengujian aspek fungsional sistem tanpa memperhatikan struktur internal atau kode program. Pengujian dilakukan dengan memverifikasi kesesuaian antara data masukan (*input*) dan hasil keluaran (*output*) sehingga dapat mengidentifikasi kesalahan pada fungsi, antarmuka, pengelolaan data, performa, serta memastikan sistem beroperasi secara optimal sesuai dengan kebutuhan pengguna.

14. *System Usability Scale (SUS)*

System Usability Scale (SUS) adalah metode pengukuran *usability* yang dikembangkan oleh John Brooke pada tahun 1986 dan digunakan sebagai alat evaluasi yang cepat, mudah diterapkan, serta andal dalam menilai tingkat kemudahan penggunaan suatu sistem berdasarkan persepsi dan pengalaman pengguna. Metode ini menggunakan sepuluh butir pernyataan yang dinilai dengan skala *Likert*, mulai dari nilai 1 (Sangat Tidak Setuju) hingga 5 (Sangat Setuju). Keunggulan SUS terletak pada proses pelaksanaannya yang sederhana dan efisien, sehingga mampu menghasilkan penilaian yang valid dalam waktu yang relatif singkat. Hasil pengukuran SUS dapat dianalisis secara kuantitatif maupun kualitatif untuk mengevaluasi tingkat kemudahan penggunaan serta mengidentifikasi kelebihan dan kekurangan suatu sistem. Secara umum, skor rata-rata SUS sebesar 68 dijadikan sebagai acuan minimal bahwa suatu sistem memiliki tingkat *usability* yang dapat diterima oleh pengguna (Putra, dkk, 2025:123).

System Usability Scale (SUS) adalah metode evaluasi yang digunakan untuk menilai tingkat usability suatu sistem atau aplikasi melalui pengujian yang melibatkan pengguna secara langsung. Penilaian dilakukan berdasarkan pengalaman pengguna dalam menggunakan sistem sehingga dapat menggambarkan tingkat kemudahan, efektivitas, dan kenyamanan penggunaan sistem. *System Usability Scale (SUS)* memiliki tingkat keandalan tinggi, banyak digunakan, efektif dalam mengukur pengalaman pengguna, serta efisien dari segi biaya dan implementasi. Metode ini diperkenalkan oleh John Brooke pada tahun 1986 dan hingga saat ini masih banyak digunakan sebagai salah satu standar dalam evaluasi *usability* pada berbagai bidang. *System Usability Scale (SUS)* terdiri atas sepuluh butir pernyataan yang dinilai menggunakan skala *Likert*, di mana responden memberikan penilaian berdasarkan pengalaman mereka terkait kemudahan penggunaan, efisiensi, dan tingkat kenyamanan dalam mengoperasikan suatu sistem. SUS memiliki keunggulan dalam mengukur usability secara cepat dan efektif, sehingga banyak digunakan dalam berbagai penelitian terkait pengalaman pengguna. Metode ini dapat diterapkan pada berbagai jenis sistem, termasuk aplikasi berbasis *web*, sistem informasi, hingga perangkat lunak lainnya. Salah satu alasan utama penggunaan SUS adalah kemampuannya dalam memberikan hasil evaluasi yang objektif dan mudah dipahami oleh pengembang sistem (Jayanti dan Putra, 2025:840). Rumus Rata-Rata Nilai Responden

$$\text{Nilai rata-rata} = \sum \frac{xi}{N}$$

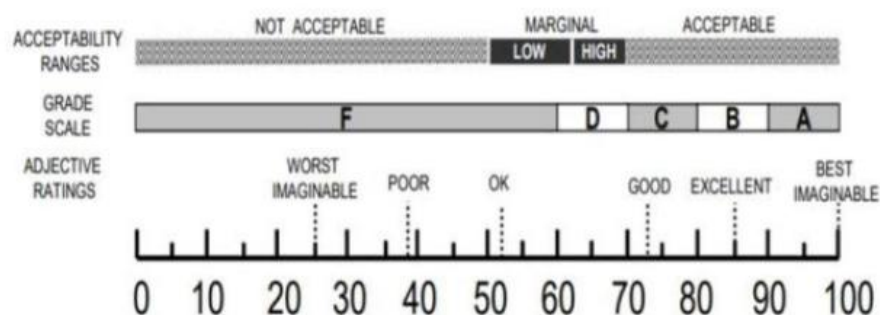
xi: nilai score responden

N: Jumlah Responden

Rumus ini memastikan bahwa nilai keseluruhan diperoleh dengan membagi total nilai responden dengan jumlah responden. Langkah ini penting untuk mengukur representasi rata-rata dari skor yang diperoleh.

$$\text{Skor SUS} = (\sum(\text{Skor Positif}-1) + \sum(5-\text{Skor Negatif}) \times 2.5$$

Dalam perhitungan *System Usability Scale* (SUS), metode penskalaan skor dilakukan secara sistematis untuk memperoleh penilaian komprehensif terhadap usabilitas suatu aplikasi. Untuk pernyataan bernomor ganjil (seperti pernyataan 1, 3, 5, 7, 9), skor responden akan dikurangi dengan angka 1, yang memungkinkan penilaian kontributif dari aspek positif. Sementara untuk pernyataan bernomor genap (seperti pernyataan 2, 4, 6, 8, 10), perhitungan dilakukan dengan mengurangi skor responden dari nilai 5, yang membantu mengidentifikasi persepsi negatif pengguna. Setelah melakukan perhitungan individual untuk setiap pernyataan, seluruh skor tersebut dijumlahkan dan dikalikan dengan konstanta 2.5. Proses ini menghasilkan nilai akhir dalam rentang 0 hingga 100, yang secara akurat menggambarkan tingkat usabilitas aplikasi dari perspektif pengguna.



Gambar 2.1. Interpretasi Skor SUS

Skor SUS diinterpretasikan dengan menggunakan beberapa tingkatan penerimaan. Skor dengan rentang 0–50 dianggap *not acceptable* (tidak dapat diterima), 51–70 masuk dalam kategori marginal (batas bawah), dan skor di atas 70 masuk ke dalam kategori *acceptable* (dapat diterima). Selain itu, skor SUS juga diklasifikasikan berdasarkan tingkat kualitas, yaitu mulai dari *worst imaginable* (terburuk yang bisa dibayangkan) hingga *best imaginable* (terbaik yang bisa dibayangkan). Rentang ini memberikan pemahaman yang jelas mengenai tingkat usability sistem berdasarkan pengalaman pengguna (Shalihah, dkk, 2025:2087).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa *System Usability Scale* (SUS) merupakan metode evaluasi *usability* yang digunakan untuk menilai tingkat kemudahan, kenyamanan, efektivitas, serta penerimaan pengguna terhadap suatu sistem berdasarkan pengalaman mereka selama menggunakan sistem. SUS menggunakan sepuluh butir pernyataan yang dinilai dengan skala *Likert* sehingga menghasilkan data kuantitatif yang dapat diinterpretasikan untuk

mengevaluasi kualitas *usability* suatu sistem. Metode ini dikenal memiliki keunggulan karena mudah diterapkan, proses pelaksanaannya relatif cepat, memiliki tingkat reliabilitas yang baik, serta mampu memberikan hasil evaluasi yang konsisten dan objektif, sehingga banyak dimanfaatkan dalam pengujian berbagai sistem maupun aplikasi berbasis teknologi informasi.

15. *Rapid Application Development (RAD)*

Rapid Application Development (RAD) merupakan salah satu metode pengembangan perangkat lunak yang menitikberatkan pada percepatan proses pembangunan sistem melalui pendekatan *prototyping*, keterlibatan pengguna secara aktif, serta pengembangan yang dilakukan secara iteratif. Metode ini terdiri atas empat tahapan utama, yaitu *requirements planning*, *user design*, *construction*, dan *cutover*, yang saling berkaitan untuk menghasilkan sistem sesuai dengan kebutuhan pengguna dalam waktu yang relatif singkat. Dengan karakteristik tersebut, RAD sangat sesuai diterapkan pada proyek yang memerlukan waktu pengembangan yang singkat serta mampu mengakomodasi perubahan kebutuhan selama proses pengembangan berlangsung (Firmansyah, dkk, 2025:101).

Metode *Rapid Application Development (RAD)* dipilih karena mampu mendukung proses pengembangan perangkat lunak dalam waktu yang relatif singkat melalui pendekatan iteratif dan kerja sama yang intensif antara pengembang dengan pengguna akhir. Keterlibatan

pengguna pada setiap tahapan pengembangan memungkinkan kebutuhan sistem dapat diidentifikasi dan disempurnakan secara berkelanjutan, sehingga perangkat lunak yang dihasilkan lebih sesuai dengan kebutuhan pengguna (Nugroho dan Rosyidah, 2025:288).

Rapid Application Development (RAD) merupakan salah satu metode pengembangan perangkat lunak yang menekankan proses pembangunan sistem secara cepat melalui pendekatan yang fleksibel. Metode ini memungkinkan penyesuaian kebutuhan pengguna dilakukan selama proses pengembangan sehingga sistem yang dihasilkan lebih sesuai dengan kebutuhan yang telah ditetapkan. Melalui pendekatan ini, pengembangan perangkat lunak dapat dilakukan secara cepat dan responsif terhadap perubahan kebutuhan selama proses pengembangan berlangsung (Bachtiar, 2025:269).

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa *Rapid Application Development* (RAD) merupakan metode pengembangan perangkat lunak yang berfokus pada percepatan proses pembangunan sistem melalui pemanfaatan *prototyping*, pengembangan secara iteratif, serta partisipasi aktif pengguna selama proses pengembangan berlangsung. Metode ini mencakup empat tahapan utama, yaitu *requirements planning*, *user design*, *construction*, dan *cutover*. Dengan pendekatan tersebut, RAD mampu menghasilkan perangkat lunak dalam waktu yang relatif singkat, tetap fleksibel terhadap perubahan kebutuhan, serta lebih sesuai dengan kebutuhan dan harapan

pengguna. Oleh karena itu, RAD menjadi salah satu metode yang tepat digunakan dalam pengembangan perangkat lunak yang memiliki batasan waktu pengerjaan namun tetap mengutamakan kualitas dan kesesuaian sistem dengan kebutuhan pengguna.

B. Kajian Empiris

Kajian empiris dalam penelitian ini adalah sebagai berikut:

1. Berdasarkan hasil penelitian yang berjudul "Optimalisasi Pengelolaan Data Surat Pernyataan Ahli Waris Dalam Meningkatkan Kualitas Pelayanan Publik Di Kantor Kecamatan Mampang Prapatan" menunjukkan bahwa Pada kualitas pelayanan SPAW di Kecamatan Mampang Prapatan telah memenuhi elemenelemen yang disampaikan menurut Parasurman antaralain: Terwujud (*Tangible*), Keandalan (*Reability*), Respon atau Tanggapan (*Responsiveness*), Jaminan (*Assurance*), dan Empati (*Empathy*) dan juga telah memenuhi elemen-elemen tata Kelola yang baik (*Good Governance*) sebagaimana yang disampaikan berdasarkan maklumat Komite Nasional Kebijakan *Governance*, yakni: Transparansi (*Transparency*), Akuntabilitas (*Accountability*), Responsibilitas (*Responsibility*), Independen (*Independent*), Kesetaraan dan Kewajaran (*Fairness*) (Rahayu, dkk, 2025:213).
2. Berdasarkan hasil penelitian yang berjudul "Pengembangan *Website* Pelayanan Administrasi bagi Masyarakat Kelurahan Sei Enam, Tanjungpinang" menunjukkan bahwa dalam upaya meningkatkan kualitas pelayanan publik di Kelurahan Sei Enam, *Website* Pelayanan Administrasi

bagi Masyarakat Kelurahan Sei Enam telah dirancang dan diimplementasikan dengan harapan agar dapat memberikan solusi terhadap berbagai permasalahan yang selama ini dihadapi, seperti keterlambatan proses administrasi, keterbatasan akses informasi, dan kebutuhan akan pelayanan yang lebih transparan dan efisien. Melalui *website* ini, masyarakat Kelurahan Sei Enam dapat mengakses layanan administrasi dan menyampaikan pengaduan secara lebih mudah dan cepat. Selain itu, fitur-fitur yang dirancang juga memungkinkan staf kelurahan untuk bekerja lebih efektif dalam mengelola data dan dokumen, sehingga mendukung terciptanya sistem pelayanan yang modern dan responsif (Alajuri, dkk, 2025:180).

3. Berdasarkan hasil penelitian yang berjudul "Rancang Bangun Sistem Informasi Desa Digital untuk Pendataan Penduduk dan Administrasi Surat Menyurat Menggunakan *Framework* Laravel" menunjukkan bahwa Penelitian ini berhasil membangun sebuah sistem informasi desa berbasis *web* yang dirancang khusus untuk meningkatkan efektivitas dan efisiensi pelayanan administrasi di Desa Banyupelle (Irfan, dkk, 2025:180).
4. Berdasarkan hasil penelitian yang berjudul "Perancangan Sistem Informasi Pelayanan Kantor Desa Berbasis *Website* Menggunakan PHP di Desa Serasah Kecamatan Pelayung Kabupaten Batang Hari" menunjukkan bahwa *web* ini dapat membantu perangkat desa dan masyarakat dalam membantu dan memudahkan penyebaran informasi yang dilakukan oleh

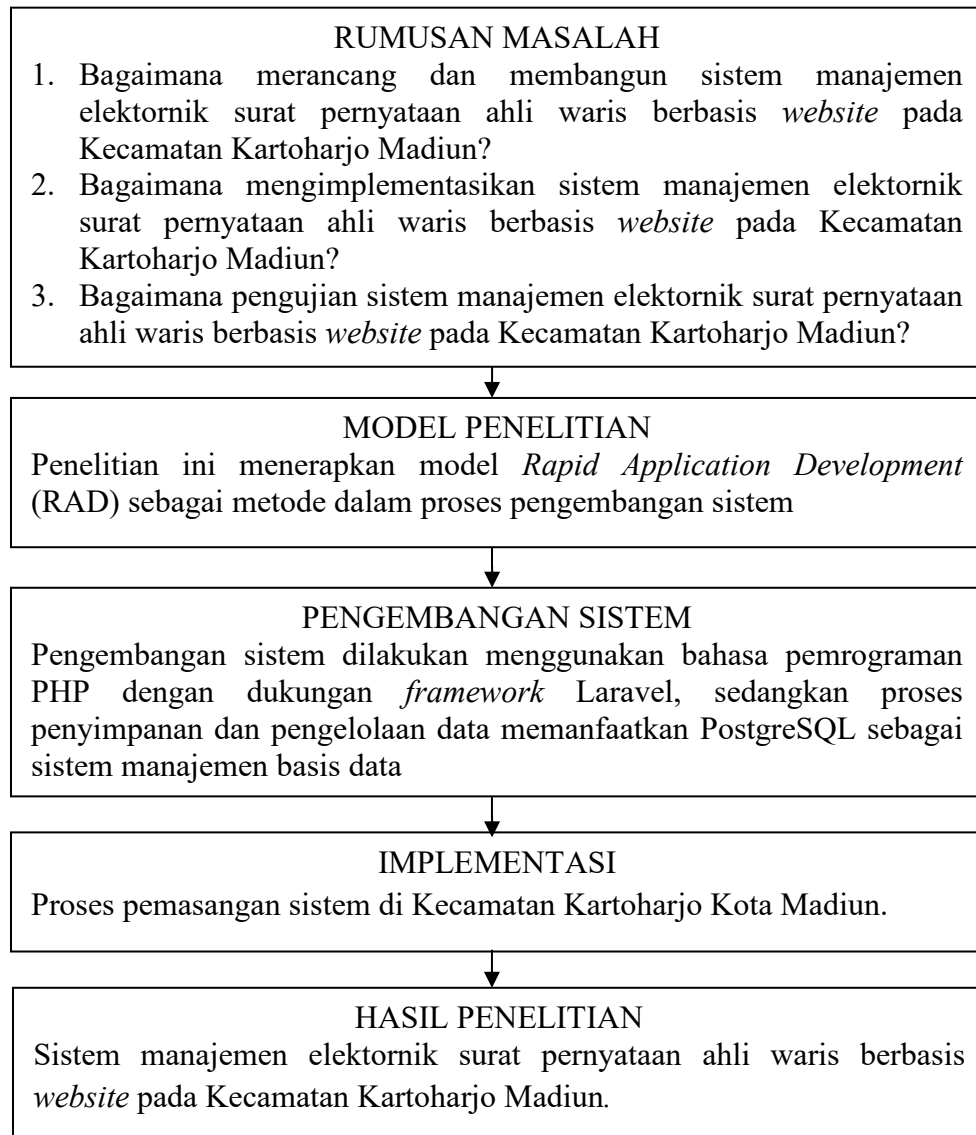
perangkat desa sehingga dapat juga diakses oleh publik (Syah, dkk, 2025:213).

5. Berdasarkan hasil penelitian yang berjudul "Sistem Informasi Pelayanan Administrasi Kependudukan Di Kelurahan Kastela Kecamatan Pulau Ternate Berbasis *Web*" menunjukkan bahwa Pembuatan Sistem Informasi Pelayanan Administrasi Kependudukan di Kelurahan Kastela Kecamatan Pulau Ternate Berbasis *Web*, *Output* dari sistem ini adalah memberikan kemudahan kepada kelurahan dengan untuk mudah mengakses informasi dan meng-input data melalui sistem yang telah dibuat. Keuntungan lain adalah untuk meningkatkan efisiensi pencatatan dan pemeliharaan data penduduk (Rauba, dkk, 2025:213).

Berdasarkan hasil penelitian-penelitian terdahulu, dapat disimpulkan bahwa penerapan sistem informasi berbasis *website* pada pelayanan administrasi pemerintahan mampu meningkatkan kualitas pelayanan publik melalui proses yang lebih efektif, efisien, transparan, dan responsif. Pemanfaatan teknologi informasi dalam pengelolaan data administrasi, seperti pelayanan surat pernyataan ahli waris, terbukti dapat mempercepat proses pelayanan, mempermudah akses informasi bagi masyarakat, meningkatkan akurasi pengelolaan data, serta mendukung kinerja aparat pemerintah dalam memberikan pelayanan yang lebih optimal. Selain itu, penggunaan framework modern seperti Laravel juga mampu mendukung pembangunan sistem yang terstruktur, aman, dan mudah dikembangkan sesuai dengan kebutuhan instansi.

C. Kerangka Berpikir

Kerangka berpikir dalam penelitian ini adalah sebagai berikut:



Gambar 2.2. Kerangka Berpikir