

BAB II

TINJAUAN PUSTAKA

2.1 Dasar Teori

Landasan teori adalah kumpulan definisi dan konsep yang disusun dengan cermat dan sistematis sehubungan dengan variabel-variabel penelitian ini. Landasan teori ini memberikan Landasan teori yang kokoh merupakan dasar penting dalam pelaksanaan penelitian. Oleh karena itu, sangat penting untuk menyusun landasan teori yang akurat dan tepat dalam suatu penelitian.

2.1.1 Website

Website, atau sering disebut sebagai *web*, adalah kumpulan halaman yang menampilkan berbagai informasi dalam bentuk data digital seperti teks, gambar, video, audio, dan animasi yang dapat diakses melalui internet. Secara istilah, *website* merujuk pada kumpulan halaman-halaman situs yang berada dalam sebuah domain atau subdomain dan tersedia di *World Wide Web* di internet. Setiap halaman dalam *website* biasanya ditulis dalam format HTML (*Hyper Text Markup Language*) dan dapat diakses menggunakan protokol HTTP, yang berfungsi untuk menghubungkan server *website* dengan pengguna sehingga informasi dapat ditampilkan. *Website* yang dipublikasikan membentuk sebuah jaringan informasi yang sangat luas. Halaman-halaman dalam sebuah *website* dapat diakses melalui URL yang dikenal sebagai *homepage*. URL ini mengorganisir halaman-halaman situs menjadi satu kesatuan, sementara *hyperlink* di dalamnya membantu pembaca memahami struktur keseluruhan dan alur informasi yang disampaikan oleh pembuat *website* [9].

Beberapa website memerlukan data input dari pengguna agar mereka dapat mengakses informasi lengkap di situs tersebut. Misalnya, ada beberapa situs bisnis, situs informasi, dan situs pemesanan yang membutuhkan informasi dari pengguna secara berkelanjutan untuk memberikan layanan yang optimal.

.

2.1.2 HTML

Hypertext Markup Language, atau HTML, dikenal sebagai Hypertext karena dalam skripnya, HTML memungkinkan teks untuk menjadi tautan yang dapat mengarahkan pengguna ke halaman lain hanya dengan mengkliknya. Teks yang berfungsi sebagai tautan ini disebut *Hypertext* karena pada dasarnya, sebuah *website* adalah dokumen yang mengandung banyak tautan untuk menghubungkan satu dokumen dengan dokumen lainnya. Istilah *Markup Language* digunakan karena skrip HTML memakai simbol-simbol tertentu, yang disebut "mark," untuk menandai bagian-bagian teks, memberikan tampilan atau fungsi khusus pada teks tersebut. Dalam praktiknya, simbol atau "mark" ini dikenal sebagai "tag." HTML merupakan bahasa dasar dalam pengembangan *website*. HTML menggunakan tanda-tanda untuk menandai bagian-bagian teks dan disebut sebagai bahasa dasar karena tanpa tambahan elemen lain, tampilan *website* menggunakan HTML saja akan kurang menarik [5]. Saat ini, HTML adalah standar internet yang didefinisikan dan diatur oleh *World Wide Web Consortium*. Standar HTML untuk *website* telah terbukti memiliki banyak keunggulan dibandingkan aplikasi desktop, seperti portabilitas yang lebih baik, kemudahan penerapan, pembaruan otomatis, dan kemudahan pemeliharaan di semua klien [18].

HTML terdiri dari berbagai kode yang memberikan instruksi kepada browser untuk menampilkan halaman dengan tampilan yang sesuai dengan yang diinginkan. Sebagai sebuah file digital, HTML dapat dibuka dan ditampilkan menggunakan berbagai jenis *browser website* seperti Mozilla Firefox, Microsoft Internet Explorer, Opera, Google Chrome, Safari, dan masih banyak lagi. Dokumen HTML dapat berisi berbagai elemen seperti teks, gambar, suara, atau video. Salah satu hal yang membedakan dokumen HTML dari dokumen lainnya adalah adanya elemen-elemen HTML yang dilengkapi dengan *tag-tag* tertentu. Elemen-elemen dan *tag-tag* ini memiliki fungsi untuk memformat atau menandai bagian tertentu dari dokumen HTML, sekaligus menentukan struktur dan tata letak dari bagian tersebut di dalam dokumen HTML [9]. Beberapa tugas utama yang dilakukan HTML dalam membangun *website* antara lain sebagai berikut:

- Menyusun *layout website*.
- Memformat teks dasar seperti pengaturan paragraf dan jenis huruf.

- Membuat daftar dan tabel.
- Menambahkan gambar, audio, dan video.
- Membuat tautan.
- Menyusun formulir.

2.1.3 Javascript

Javascript pertama kali dipromosikan oleh penemunya, yaitu perusahaan Netscape, pada tahun 1995. Pada awal kemunculannya, Javascript dikenal dengan nama *livescript* dan berfungsi sebagai bahasa pemrograman sederhana yang dirancang khusus untuk browser Netscape Navigator 2. Javascript adalah sebuah bahasa pemrograman yang berbentuk kumpulan skrip, yang berfungsi untuk melengkapi dan memperkaya fitur-fitur pada suatu dokumen HTML. Sepanjang sejarah internet, bahasa ini menjadi bahasa pemrograman pertama yang digunakan secara luas untuk pembuatan *website*. Javascript berfungsi sebagai bahasa pemrograman yang menambahkan kemampuan pada HTML, javascript memungkinkan eksekusi perintah di sisi pengguna, yaitu di browser, bukan di server *website*. Javascript sangat bergantung pada browser yang memuat halaman *website* yang mengandung skrip-skrip javascript, yang tentunya disematkan dalam dokumen HTML [17].

Javascript telah menjadi salah satu bahasa yang paling populer. JavaScript hanya mendukung proses *single thread*, javascript hanya dapat mengerjakan satu tugas dalam satu waktunya. Oleh karena itu, ketika suatu tugas dilakukan, maka tugas berikut menunggu sampai tugas sebelumnya selesai. Jika situs *website* memiliki tugas berat yang tidak mampu dilakukan oleh satu *thread* pun, mereka menjadi tidak responsif karena *overhead* [19].

2.1.4 CSS

CSS, atau *Cascading Style Sheets*, adalah skrip yang digunakan untuk merancang tampilan *website*. CSS berperan dalam menghubungkan elemen-elemen HTML dengan berbagai atribut yang tersedia, memungkinkan penyesuaian tampilan sesuai gaya yang diinginkan. Meskipun HTML juga dapat digunakan untuk mengatur tampilan *website*, kemampuannya terbatas. CSS

memberikan kontrol yang lebih rinci dibandingkan HTML, sehingga struktur *website* yang dibangun dengan HTML menjadi lebih terorganisir dan menarik [9].

Sebagian orang berpendapat bahwa CSS tidak termasuk dalam kategori bahasa pemrograman, karena strukturnya yang sangat sederhana, hanya terdiri dari kumpulan aturan yang mengatur gaya elemen pada HTML. Cara kerja CSS dalam memodifikasi HTML adalah dengan memilih elemen HTML yang akan diatur, kemudian memberikan atribut yang sesuai dengan tampilan yang diinginkan. Dalam menyusun elemen HTML, skrip CSS dibagi menjadi tiga bagian utama, yaitu *Selector* untuk memilih elemen yang akan diberi aturan, *atribut* yang merupakan aturan yang ditetapkan, dan *value* sebagai nilai dari aturan yang ditentukan [20].

CSS biasanya digunakan untuk mengatur tata letak dan tampilan halaman web yang dibangun dengan HTML. Dengan CSS, kita bisa mengatur berbagai elemen visual, seperti ukuran gambar, warna teks, warna tabel, ketebalan dan warna border, warna *hyperlink*, efek saat cursor mouse melewati elemen, jarak antar paragraf, jarak antar teks, margin di bagian kiri, kanan, atas, dan bawah, serta berbagai parameter lainnya. CSS adalah bahasa *style sheet* yang berfungsi untuk mengatur dan mendesain tampilan dokumen. Penggunaan CSS memungkinkan halaman yang sama ditampilkan dengan format yang berbeda [5].

2.1.5 Basis Data

Basis data merupakan kumpulan data yang saling terkait dan disimpan bersama dalam suatu media tanpa memerlukan penempatan berurutan atau redundansi data yang berlebihan. Redundansi, jika ada harus diminimalkan dan dikendalikan. Data disimpan menggunakan metode khusus untuk memfasilitasi penggunaan dan tampilan ulang. Data paling baik digunakan oleh satu atau lebih program aplikasi dan disimpan secara independen dari program yang menggunakannya. Data disimpan sedemikian rupa sehingga memungkinkan data ditambahkan, diambil, dan dimodifikasi dengan cara yang mudah dan dapat terkontrol [21].

Basis data merupakan kumpulan data dimana saling terkait, biasanya disimpan menurut sebuah model data. Biasanya, *database* dipakai beberapa

program perangkat lunak melalui DBMS bisa satu atau lebih. Secara kolektif, basis data, DBMS, dan program perangkat lunak disebut sebagai *database* sistem. *Database* mengikuti satu atau beberapa model data, yaitu bagaimana dan data apa yang dapat disimpan, dan terkadang terjadi pengambilan dan manipulasi data. Model data dapat berupa konseptual, logis, atau fisik. Model konseptual seperti model hubungan entitas tidak menentukan bagaimana seharusnya data disimpan, tetapi lebih digunakan untuk menggambarkan keterkaitan dan karakteristik karakter data. Model data logis seperti relasional model terkait dengan bagaimana data disimpan dan disajikan, tetapi seringkali tanpa menjelaskan bagaimana data disimpan secara fisik yang komputasi node bertanggung jawab untuk menyimpan data, dimana data tersebut di disk [22].

2.1.6 MySQL

MySQL adalah sebuah *Relational Database Management System* atau biasa disingkat (RDBMS) bersifat *open source*. Perangkat lunak ini umumnya digunakan bersama dengan bahasa pemrograman server *website* seperti javascript atau JSP. MySQL, yang merupakan singkatan dari *My Structured Query Language*, adalah program yang digunakan untuk membuat dan mengelola *database*, sering disebut sebagai *Database Management System* (DBMS), dengan lisensi *open source*. Selain itu, MySQL juga dapat digunakan untuk mengakses *database* berbasis jaringan, sehingga memungkinkan penggunaan dalam aplikasi *multi-user* [9]. Salah satu kelebihan MySQL adalah gratis, handal, terus diperbarui, dan didukung oleh banyak forum yang membantu pengguna dalam mengatasi masalah. MySQL juga sering dibundel dengan server *website*, membuat proses instalasinya yang lebih mudah [13].

SQL adalah bahasa pemrograman khusus yang digunakan untuk mengelola data dalam sistem manajemen basis data relasional. SQL terdiri dari sintaks sederhana berupa instruksi-instruksi yang digunakan untuk memanipulasi data, yang dikenal dengan sebutan query. Bahasa query ini digunakan untuk mengimplementasikan basis data *purchase order* [21]. Terdapat beberapa kode yang membuat SQL lebih praktis seperti perintah yang dikodekan untuk mengurangi ukuran pesan, gunakan nilai biner 8-bit, untuk yang paling umum

kata kunci SQL SELECT yang digunakan. Pesan yang berbeda dapat diklasifikasikan menjadi beberapa jenis menurut muatannya [23].

2.1.7 Prisma

Prisma adalah ORM *open source* yang dirancang untuk Node.js. Ini berfungsi sebagai alternatif untuk menulis SQL secara manual, atau menggunakan alat lain untuk mengakses basis data, seperti pembuat query. Prisma saat ini mendukung PostgreSQL, MySQL, SQL server, SQLite, MongoDB, dan CockroachDB. Meskipun Prisma dapat digunakan dengan javascript biasa, prisma mencakup typescript dan memberikan tingkat keamanan tipe yang melampaui jaminan ORM lain dalam ekosistem *typescript*.

Prisma sendiri menyediakan banyak sekali *library* yang dapat digunakan untuk mengelola basis data. Adapun beberapa tugas yang dapat memungkinkan untuk dilakukan oleh prisma ialah mengakses *query*, melakukan migrasi, data *modelling*, dan *seeding*. Selain itu, prisma juga mampu untuk membuat dokumentasi pada tahap pengembangan menjadi lebih baik. Hal ini dikarenakan prisma selalu mencatat setiap perubahan yang terjadi dalam basis data kedalam sebuah *file* migrasi.

2.1.8 Node.Js

Node.js merupakan sebuah *platform* pengembangan *server* sisi yang memanfaatkan bahasa javascript untuk eksekusi kode di sisi *server*. Node.js dikembangkan oleh penemunya yaitu Ryan Dahl pada tahun 2009. Node.js muncul sebagai solusi untuk pengembangan *website* tradisional. Arsitektur Node.js menggunakan mesin javascript V8 dari Google Chrome. *Platform* ini telah menjadi pilihan utama untuk pengembangan *server* sisi yang membutuhkan kinerja tinggi, terutama dalam konteks aplikasi *real time* dan mikro layanan. Node.js, memudahkan pengelolaan dependensi dan distribusi paket javascript. Node.js adalah sebuah sistem perangkat lunak yang dirancang untuk pengembangan aplikasi web dan juga dikenal sebagai *runtime environment*. Aplikasi ini ditulis menggunakan kombinasi bahasa C++ dan javascript, serta mengadopsi model *event-driven* (berbasis event) dan asynchronous I/O. Berbeda

dari kebanyakan javascript yang dijalankan di dalam browser web, Node.js dieksekusi sebagai aplikasi server. Node.js dapat berjalan di server berkat dukungan dari *V8 engine* buatan google, serta beberapa modul bawaan yang sudah terintegrasi, seperti modul HTTP, modul filesystem, modul keamanan, dan modul penting lainnya [24].

Node.js dilengkapi dengan berbagai modul bawaan seperti NGINX, yang menangani kebutuhan umum seperti manajemen file, HTTP (*Hypertext Transfer Protocol*), SSL (*Secure Sockets Layer*), DNS (*Domain Name System*), kompresi, dan kriptografi. Penggunaan Node.js juga didukung oleh Node *Package Manager* (NPM), yaitu kumpulan kode yang memungkinkan pengembang untuk menginstal paket yang dibutuhkan dalam proyek mereka, sehingga mempermudah dan mempercepat proses pengembangan. Berkat kinerja yang unggul dalam hal kecepatan dan efisiensi sumber daya, Node.js telah diadopsi oleh banyak perusahaan besar, termasuk Netflix, LinkedIn, NASA, PayPal, Medium, dan lainnya [25].

2.1.9 Express Js

Express.js dikembangkan oleh TJ Holowaychuk dan pertama kali dirilis pada tahun 2010. Tujuan utamanya adalah menyediakan kerangka kerja atau framework yang sederhana namun kuat untuk membangun aplikasi *website* dengan menggunakan Node.js. Pada saat itu, express.js telah menjadi salah satu *framework* Node.js yang paling populer dan banyak digunakan. Express.js adalah sebuah *framework* Node.js yang berfungsi untuk memudahkan pengembangan aplikasi berbasis Node.js dengan menggunakan pola desain yang dapat disesuaikan dan sangat fleksibel. Selain itu, express.js juga merupakan *framework* yang sangat ringan, menjadikannya ideal untuk membuat aplikasi *web* dan API [24]. Express adalah kerangka kerja aplikasi web Node.js yang minimalis dan fleksibel, serta menyediakan serangkaian fitur yang kuat untuk pengembangan aplikasi *web* dan seluler. Ada beberapa alasan mengapa Express dipilih dalam pengembangan *startup ubaform* [25].

Express.js adalah sebuah *framework* minimalis, dan *developer* telah membuat berbagai *package middleware* yang kompatibel untuk menyelesaikan

berbagai tantangan dalam pengembangan *website*. Ada juga beberapa *library* yang memudahkan *developer* dalam menangani *cookie*, *session*, *login* pengguna, parameter URL, data POST, *security header*, dan lain-lain. Selain fitur-fitur ini, salah satu alasan utama menggunakan Express dalam pengembangan *website* adalah popularitasnya. Popularitas *framework* sangat penting karena menjadi tanda apakah *framework* tersebut akan terus berkembang dan didukung di masa depan. Semakin populer sebuah *framework*, semakin banyak dokumentasi, *library* tambahan, dan dukungan teknis yang tersedia, sehingga memudahkan pengembangan serta pencarian tenaga kerja yang sesuai. Express pertama kali dirilis pada November 2010 dan saat ini telah mencapai versi 4.17.1 API [25].

2.1.10 Interface

Interface adalah sebuah program yang memungkinkan pengguna untuk berinteraksi langsung dengan perangkat mereka atau melalui jaringan. *Interface* juga dapat merujuk pada kontrol yang digunakan dalam program untuk memungkinkan interaksi pengguna dengan aplikasi tertentu. Secara umum, *interface* adalah perangkat atau sistem yang memungkinkan entitas yang tidak terhubung langsung untuk saling berinteraksi. Contoh sederhana dari *interface* adalah remote control, yang merupakan perangkat keras yang memudahkan Anda untuk mengendalikan televisi, komputer, atau perangkat lainnya. perangkat elektronik lainnya. *Interface* memainkan peran penting dalam aplikasi dan *website*. *User Interface* (UI) adalah tampilan aplikasi yang memfasilitasi interaksi pengguna dengan aplikasi tersebut [26].

2.1.11 Figma

Figma adalah aplikasi desain berbasis cloud dan alat prototyping yang digunakan untuk proyek digital. Aplikasi ini dirancang untuk memfasilitasi kolaborasi dalam proyek dan memungkinkan kerja *team* dari lokasi manapun, Figma membantu menghemat waktu dalam *verifikasi* desain. Pengguna dapat berkolaborasi dengan memberikan komentar, saran, dan bahkan mengubah desain secara bersamaan. Figma sering digunakan oleh desainer UI dan UX digunakan untuk merancang antarmuka pengguna pada *website* atau aplikasi *mobile*. Tidak

seperti Adobe Photoshop, Figma memungkinkan beberapa desainer untuk bekerja bersama secara bersamaan dalam dokumen yang sama, serta memberikan komentar dan saran secara *real-time*. dan melakukan perubahan tampilan secara *real-time*. Setiap perubahan disimpan secara otomatis di Figma [27].

2.1.12 Fronend

Frontend adalah elemen yang menghubungkan pengguna dengan sistem *backend*, biasanya berupa antarmuka pengguna tempat interaksi berlangsung. Pekerjaan seorang *frontend developer* sering melibatkan desain *user interface* (UI) dan *user experience* (UX). *Frontend developer* tidak terlibat dalam logika bisnis aplikasi, melainkan lebih fokus pada aspek antarmuka. *Frontend* beroperasi di sisi klien, yang berarti eksekusi dan pemrosesan dilakukan pada perangkat pengguna. Dalam pengembangan *frontend*, bahasa pemrograman seperti HTML, CSS, dan javascript sering digunakan. HTML digunakan untuk merancang struktur dasar halaman *web*, CSS untuk mengatur tata letak dan tampilan visual, dan javascript untuk menambahkan interaktivitas dan dinamisme pada halaman *web* [28].

Penting untuk dicatat bahwa *frontend* bekerja bersama dengan *backend*, yang berada di sisi server, untuk menyediakan fungsionalitas lengkap aplikasi. *Backend* bertanggung jawab atas pemrosesan data, logika bisnis, dan berbagai tugas server *side*. Sedangkan *frontend* bertanggung jawab atas presentasi informasi dan interaksi pengguna.

2.1.13 Backend

Backend adalah bagian dari suatu aplikasi atau sistem di mana semua proses utama terjadi. Di bagian *backend*, data diproses, ditambahkan, diubah, atau dihapus. *Backend* menangani berbagai aspek yang biasanya tidak terlihat atau tidak berinteraksi langsung dengan pengguna, seperti *database* dan server. *Backend developer* adalah seorang programmer yang fokus pada keamanan, desain sistem, dan pengelolaan data. Mereka berperan penting pada pengembangan sistem atau aplikasi dinamis yang memiliki data yang sering berubah, seperti *website* dinamis seperti Facebook dan Google. Bahasa

pemrograman yang umum digunakan di *backend developer* termasuk PHP, ASP, NodeJs, atau C, yang berguna dalam pengolahan *database*, memproses data, dan melakukan operasi I/O [28].

2.1.14 Postman

Postman adalah alat yang digunakan untuk menguji RESTful API dengan cara membuat permintaan dalam format HTML tanpa menimbulkan kesulitan bagi pengguna. Anda tidak perlu menulis kode kompleks untuk memeriksa apakah API berfungsi dengan baik. Misalnya, untuk menampilkan permintaan GET dari *database*, Anda hanya perlu memasukkan URL aplikasi yang diinginkan di Postman dan memilih tab GET untuk melihat datanya. Jika Anda ingin menambahkan data baru, cukup pilih tab POST setelah mengisi informasi yang akan disimpan di *database*. Data yang disimpan dapat dilihat dengan mengetikkan URL aplikasinya, dan data yang ada akan ditampilkan dalam format JSON. Selain fungsi GET dan POST, Postman juga menawarkan opsi PUT, PATCH, dan DELETE untuk mengelola dokumen yang disimpan di MongoDB [30].

2.1.15 XAMPP

XAMPP adalah paket *software* yang bersifat *open source*, XAMPP sangat memungkinkan pengembang untuk membangun dan mengelola lingkungan pengembangan *website* lokal di komputer mereka. XAMPP dirancang untuk menyediakan server Apache untuk *web*, sistem manajemen basis data MySQL, dan interpreter bahasa pemrograman PHP dalam satu paket yang mudah *diinstal*. Fungsi utama XAMPP yaitu berperan menjadi server lokal (*localhost*) yang mencakup beberapa program, seperti server HTTP Apache, basis data MySQL, serta penerjemah bahasa pemrograman PHP dan Perl [29].

XAMPP atau yang singkatan dari X yang melambangkan dukungan untuk berbagai sistem operasi, serta Apache, MySQL, PHP, dan Perl. Program ini tersedia secara gratis di bawah GNU General Public License dan dirancang untuk menjadi server *web* yang mudah digunakan, memungkinkan tampilan halaman *web* yang dinamis. XAMPP dapat diunduh langsung dari situs resminya. Berikut

adalah penjelasan singkat tentang beberapa komponen dalam XAMPP: Server HTTP Apache, yang juga dikenal sebagai server *web* atau WWW, adalah server yang kompatibel dengan berbagai sistem operasi seperti Unix, BSD, Linux, Microsoft Windows, dan Novell Netware. Server ini berfungsi untuk menyajikan dan menjalankan situs *web*. dengan menggunakan protokol HTTP [29].

2.1.16 Localhost

Localhost adalah akses lokal yang diperoleh dari sebuah aplikasi untuk mengakses *server* lokal. Fungsinya adalah untuk mengevaluasi kelebihan dan kekurangan aplikasi sebelum dilakukan proses *hosting*. *Hosting* adalah lokasi di mana data situs *web* disimpan, memungkinkan akses melalui internet. Peneliti memilih menginstal aplikasi di *localhost* sebelum melakukan *hosting* karena hal ini memungkinkan mereka untuk mengidentifikasi kekurangan dan kelebihan aplikasi sambil mengurangi penggunaan internet jika terjadi kesalahan sebelum aplikasi diunggah ke *hosting* yang akan digunakan [31].

2.1.17 Apache JMeter

Aplikasi Apache JMeter adalah perangkat lunak open source, 100% aplikasi Java murni dirancang untuk memuat tes perilaku fungsional dan mengukur kinerja. Ini pada awalnya dirancang untuk pengujian aplikasi web tetapi sejak diperluas untuk fungsi tes lainnya.

Apache JMeter adalah aplikasi open source berbasis java yang dapat dipergunakan untuk performance test. Bagi seorang QA engineer jMeter bisa digunakan untuk melakukan load/stress testing web *application*, FTP *application* dan *database* server test. Apache JMeter dapat digunakan untuk menguji kinerja baik pada sumber daya statis dan dinamis (web *services* (SOAP / REST), web bahasa dinamis - PHP, Java, ASP, NET, file, dll -, java objects, pangkalan data dan pertanyaan, FTP server dll). Hal ini dapat digunakan untuk mensimulasikan beban berat pada server, sekelompok server, jaringan atau objek untuk menguji kekuatan atau untuk menganalisa kinerja secara keseluruhan di bawah jenis beban yang berbeda [32].

2.2 Kajian Pustaka

Dalam penulisan skripsi ini, peneliti menggali informasi dari penelitian sebelumnya sebagai bahan perbandingan. Selain itu, peneliti juga menelusuri informasi dari berbagai buku dan skripsi untuk mendapatkan informasi yang telah ada sebelumnya mengenai teori yang berkaitan dengan judul yang digunakan, guna memperoleh landasan teori ilmiah.

1. Putra Pirma Arhandi. Mahasiswa Politeknik Negri Malang dengan judul “Pengembangan Sistem Informasi Perijinan Tenaga Kesehatan dengan Menggunakan Metode *Backend* dan *Frontend*.”

Hasil dari penelitian ini menunjukkan bahwa Pada proses komunikasi antara *frontend* dan *backend* terlihat bahwa data yang di kirim oleh *frontend* ke *backend* berupa data plain text dimana kemudian *backend* juga mengembalikan data dalam bentuk text berformat json. Cara komunikasi antara *backend* dan *frontend* yang seperti ini mempercepat pada proses transfer data dan performa aplikasi dimana dalam proses login hanya membutuhkan waktu selama 114 milli second.

2. Skripsi Nasution. Mahasiswa Universitas Islah Indonesia tahun 2021 dengan judul “Implementasi Mongo Db, Express Js, React Js dan Node Js (Mern) pada Pengembangan Aplikasi Formulir, Kuis, dan Survei Online.

Hasil penelitian ini menyimpulkan bahwa javascript mempermudah pengembang dalam memahami kode, baik di sisi *frontend* maupun *backend* secara bersamaan. Meskipun ada teknologi lain yang bisa dijadikan alternatif, seperti AngularJS untuk membangun sisi *frontend* berbasis SPA dan MySQL sebagai *database*, reactjs menawarkan fleksibilitas yang lebih tinggi. Ini karena reactjs adalah sebuah *library* yang cenderung lebih fleksibel dalam penggunaannya, termasuk dalam hal struktur folder dan penambahan *library* pendukung lainnya. Sementara itu, mongodb menunjukkan performa yang lebih baik dibandingkan MySQL dalam penelitian ini dan menawarkan kemudahan dalam struktur penyimpanan data dalam format JSON, yang lebih mudah untuk diolah.

3. Intan Kumalasari, Arif Azzani Saputra. Jurnal Pengabdian dan Pemberdayaan pada Masyarakat tahun 2023 dengan judul “Pelatihan Dan Pembuatan *Website* Menggunakan Html Dan Css.

Berkembangnya dunia informatika dari waktu ke waktu, berbagai hal dalam dunia pengembangan teknologi ini sendiri juga ikut berkembang, diantaranya adalah HTML, CSS, dan javascript yang umumnya digunakan untuk pembuatan *website* yang terus mendapatkan pembaharuan sebagai efek dari para programmer yang juga semakin berkembang. Perkembangan ini tentunya membuat kemudahan bagi manusia dalam menyelesaikan pekerjaan yang berkaitan dengan *website*.

Dari tiga kajian Pustaka di atas, perbedaan terhadap penelitian skripsi ini adalah perbedaan pada jenis *website*, begitu juga penggunaan kombinasi *frontend* dan *backend* yang masih belum diterapkan pada pembuatan *website*. Adapun pada penelitian ini menggunakan metode-metode yang sama dengan tiga kajian Pustaka di atas seperti penggunaan HTML, CSS, javascript, mysql, express js dan node js pada pembuatan *website*.