

Pengembangan Website E-Commerce Hisa Studio: Perpaduan React dan Firebase untuk Performa Maksimal

Arif Bakti Nugraha¹, Muhammad Adrisa NS²

¹ Jurusan Informatika, Universitas Informatika dan Bisnis Indonesia
Jl. Soekarno-Hatta No.643, Sukapura, Kec. Kiaracondong, Kota Bandung, Jawa Barat, Indonesia

² Jurusan Teknik Informatika, Universitas Nasional Pasim
Jl. Dakota No.8A, Sukaraja, Kec. Cicendo, Kota Bandung, Jawa Barat, Indonesia
arifbakti@unibi.ac.id

Abstrak

Pada era digital, website e-commerce menjadi kebutuhan penting bagi banyak bisnis, termasuk Hisa Studio. Penelitian ini mengusulkan sekaligus mengimplementasikan penggunaan React dan Firebase dalam pembangunan website e-commerce, di mana React digunakan sebagai library JavaScript untuk pengembangan antarmuka pengguna yang interaktif, sedangkan Firebase dimanfaatkan sebagai platform cloud yang menyediakan layanan backend, seperti autentikasi pengguna dan penyimpanan data melalui Firestore berbasis NoSQL. Proses pengembangan meliputi inisialisasi Firebase, pengelolaan autentikasi pengguna dengan Firebase Authentication, pengelolaan sesi, serta interaksi dengan Firestore untuk penyimpanan dan pengambilan data kategori produk. Selain itu, penelitian ini juga menambahkan fungsi pendaftaran pengguna menggunakan email dan password untuk meningkatkan pengalaman serta keamanan pengguna. Dengan kombinasi teknologi tersebut, Hisa Studio diharapkan mampu memperoleh keunggulan dalam hal antarmuka yang responsif dan sistem penyimpanan data yang efisien. Evaluasi dilakukan terhadap kinerja, keamanan, dan fungsionalitas sistem, sehingga hasil penelitian ini diharapkan memberikan kontribusi positif dalam pengembangan aplikasi web e-commerce berbasis teknologi modern.

Kata kunci: Website e-commerce, React, Firebase, Otentikasi, Firestore, Implementasi, Keamanan, Fungsionalitas

Abstract

In the digital era, e-commerce websites have become an essential need for many businesses, including Hisa Studio. This study proposes and implements the use of React and Firebase in developing an e-commerce website, where React serves as a JavaScript library for building interactive user interfaces, while Firebase functions as a cloud platform that provides backend services such as user authentication and data storage through the NoSQL-based Firestore. The development process includes Firebase initialization, user authentication management with Firebase Authentication, session management, and interaction with Firestore for storing and retrieving product category data. Furthermore, this research incorporates additional authentication features such as user registration with email and password to enhance user experience and security. By combining these technologies, Hisa Studio is expected to gain advantages in responsive user interface development and efficient data storage. The system was evaluated in terms of performance, security, and functionality, and the findings are expected to provide positive contributions to the development of modern web-based e-commerce applications.

Keywords: E-commerce website, React, Firebase, Authentication, Firestore, Implementation, Security, Functionality

I. PENDAHULUAN

Pesatnya perkembangan e-commerce telah merombak lanskap ritel modern secara fundamental, menawarkan peluang tak terbatas bagi bisnis untuk

terlibat dengan pelanggan di skala global. E-commerce, yang didefinisikan sebagai pembelian dan penjualan barang atau jasa melalui internet, telah menjadi bagian tak terpisahkan dari kehidupan konsumen, menyediakan kenyamanan, pilihan, dan

aksesibilitas yang tak tertandingi [1]. Keunggulan ini sangat dihargai oleh individu yang memiliki jadwal padat, seperti mahasiswa, yang mendapati e-commerce sebagai solusi efisien untuk memenuhi kebutuhan belanja mereka [2]. Di Indonesia, lanskap e-commerce berkembang pesat, dengan marketplace online yang bermunculan sebagai pusat yang ramai bagi pembeli dan penjual [3]. E-commerce telah mengubah cara bisnis beroperasi, memungkinkan mereka untuk memperluas jangkauan pasar dan meningkatkan efisiensi operasional [4][5]. Dalam era globalisasi saat ini, kemajuan teknologi telah memengaruhi setiap aspek kehidupan manusia, terutama di sektor perdagangan, mengubah sistem perdagangan konvensional menjadi platform digital melalui aplikasi [6].

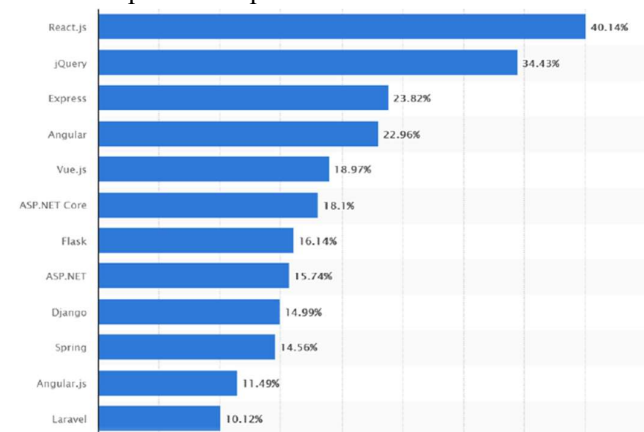
Munculnya teknologi digital telah memicu gelombang kewirausahaan baru, dengan banyak individu, khususnya generasi muda, memanfaatkan platform online untuk memulai dan mengembangkan bisnis mereka [7]. Transformasi digital ini telah memberdayakan individu untuk mengejar hasrat mereka dan menciptakan usaha mereka sendiri, berkontribusi pada pertumbuhan ekonomi dan inovasi [8]. Strategi pemasaran melalui media sosial telah muncul sebagai fenomena bisnis yang menguntungkan, memberikan peluang besar bagi perusahaan untuk menjangkau audiens target mereka secara efektif. Pergeseran perilaku konsumen menuju transaksi online didorong oleh kepraktisan dan efisiensi yang terkait dengan belanja online, yang menghemat waktu, tenaga, dan biaya [9]. Bisnis kini harus mengadaptasi praktik bisnis dan pemasaran mereka agar tetap relevan di era digital baru [10].

Perkembangan teknologi dan penggunaan internet berkembang pesat di Indonesia [10]. E-commerce telah memungkinkan bisnis untuk menjangkau basis pelanggan yang lebih luas, melampaui batas geografis dan demografis tradisional. Dengan menjangkau pasar yang lebih luas, bisnis dapat meningkatkan penjualan dan profitabilitas mereka, yang mengarah pada pertumbuhan dan ekspansi. Keberadaan teknologi digital juga menjadi komponen penting bagi pelaku bisnis untuk memasarkan produk secara online [11]. Ekonomi digital di Indonesia telah mengalami perubahan signifikan seiring dengan perubahan perilaku masyarakat, dimana sebagian besar konsumen mencari informasi produk secara online sebelum melakukan pembelian [12]. Perkembangan teknologi informasi dan media elektronik mengajak dan memaksa masyarakat untuk bergerak ke arah serba digital [13].

Dalam konteks ekonomi digital yang berkembang pesat, pengembangan platform e-

commerce yang efisien dan mudah digunakan sangat penting bagi bisnis untuk bersaing secara efektif. Platform e-commerce memungkinkan bisnis untuk mempresentasikan produk dan layanan mereka kepada audiens global, memfasilitasi transaksi online, dan mengelola hubungan pelanggan. Pemerintah Indonesia juga telah mengakui pentingnya transformasi digital dan telah meluncurkan inisiatif seperti "Making Indonesia 4.0" untuk mendorong inovasi teknologi dan mempercepat adopsi digital di berbagai industri [8]. Oleh karena itu, penelitian ini bertujuan untuk merancang dan mengembangkan situs web e-commerce untuk Hisa Studio menggunakan React dan Firebase, dengan tujuan untuk menyediakan platform yang menarik secara visual, fungsional, dan ramah pengguna untuk menjual produk mereka secara online.

Terdapat banyak teknologi yang dapat digunakan untuk membangun sebuah website e-commerce seperti React.js, jQuery, dan Express. Dilansir dari <https://www.monocubed.com/>, pada Gambar 1-1 Urutan penggunaan framework terpopuler 2023, terdapat urutan penggunaan framework populer dalam melakukan pengembangan website. Peringkat tertinggi merupakan React.js yang memiliki bahasa dasar berupa Javascript.



Gambar 1 Urutan penggunaan framework terpopuler 2023

Dari gambar 1 diatas, dapat dianalisis bahwa React.js menempati posisi dominan dengan persentase penggunaan sebesar 40,14%, diikuti oleh jQuery (34,43%) dan Express.js (23,82%). Hal ini mengindikasikan bahwa *library* berbasis komponen (React.js) dan *library* utilitas (jQuery) masih sangat relevan dalam ekosistem pengembangan *web* kontemporer.

Sementara itu, *framework* yang lebih komprehensif seperti Angular (22,96%) dan Vue.js (18,97%) juga menunjukkan adopsi yang signifikan, meskipun tidak sebesar React.js. Di sisi *backend*, Express.js sebagai *framework* Node.js unggul

dengan 23,82%, diikuti oleh ASP.NET Core (18,10%) dan Flask (16,14%).

Terdapat dua alasan mengapa React.js menjadi framework frontend yang akan digunakan dalam membangun website e-commerce Hisa Studio:

1. React.js merupakan framework yang sering dan banyak digunakan pada tahun 2023. Hal ini menandakan bahwa dokumentasi dan komunitas developer sangat mudah ditemukan. Dengan demikian akan lebih mudah dalam melakukan troubleshoot jika terdapat kendala dalam pengembangan website e-commerce Hisa Studio
2. React.js memiliki fitur component-based yang memungkinkan pengembang untuk membuat Encapsulated Component agar dapat membuat complex UI (User Interface) maupun atomic design. Sehingga dengan membuat komponen untuk setiap fitur pengembang dapat melakukan debug dengan lebih mudah.

Dari sisi backend, pengembangan dilakukan menggunakan layanan dari Firebase. Firebase merupakan layanan dari Google untuk membantu pengembang membangun aplikasi dengan memudahkan pekerjaan backend. Firebase memiliki banyak fitur yang dapat membantu pengembangan backend yaitu:

1. Firebase Authentication yang merupakan layanan backend dengan tampilan interfaces yang mudah untuk digunakan pada kebutuhan autentikasi pengguna ke aplikasi. Firebase Authentication mendukung autentikasi menggunakan identitas gabungan seperti akun Google, Facebook, dan sebagainya
2. Firebase Firestore adalah database management system NoSQL yang bersifat fleksibel dan terukur untuk pengembangan aplikasi. Firestore dapat digunakan untuk menyimpan database NoSQL berupa document dengan format JSON yang di hosting di cloud.

Hisa Studio, sebagai bisnis yang ingin meningkatkan kehadiran online dan memperluas basis pelanggannya, dapat memanfaatkan kekuatan teknologi e-commerce untuk mencapai tujuan ini. React, perpustakaan JavaScript populer untuk membangun antarmuka pengguna, akan digunakan untuk mengembangkan antarmuka depan situs web, memastikan pengalaman pengguna yang lancar dan interaktif. Firebase, platform pengembangan aplikasi seluler dan web dari Google, akan digunakan untuk backend situs web, menyediakan database yang aman dan skalabel, layanan otentikasi, dan fungsionalitas hosting. Dengan memanfaatkan teknologi ini, Hisa Studio dapat membangun platform e-commerce yang memenuhi kebutuhan

spesifik mereka dan memberikan pengalaman berbelanja yang mulus bagi pelanggan mereka [14][15]. Beberapa bisnis kecil dan menengah telah mulai menggunakan komputer dan internet untuk meningkatkan efisiensi operasional dan menjangkau pelanggan baru [8]. Selain itu, situs web ini akan dirancang untuk dioptimalkan untuk perangkat seluler, mengingat penggunaan smartphone yang meluas di Indonesia [16].

II. METODE PENELITIAN

A. Activity Diagram

Metode penelitian menggunakan Activity Diagram untuk memodelkan alur proses sistem secara visual dan sistematis. Activity Diagram membantu menggambarkan aktivitas, keputusan, dan aliran kerja dalam sistem sehingga memudahkan analisis dan perancangan sistem informasi [17]. Diagram ini digunakan untuk mengidentifikasi langkah-langkah proses bisnis dan interaksi antar pengguna dengan sistem secara jelas. Pendekatan ini efektif dalam mengurangi kesalahan desain dan meningkatkan pemahaman tim pengembang terhadap kebutuhan sistem [18]. Dengan Activity Diagram, proses pengembangan sistem menjadi lebih terstruktur dan terkontrol.

B. Autentikasi Pengguna dan Keamanan E-commerce

Autentikasi pengguna dan keamanan *e-commerce* telah menjadi fokus utama dalam penelitian. Fokus pada implementasi autentikasi multi-faktor (MFA) dan enkripsi data guna melindungi informasi pengguna dari ancaman siber. Pendekatan ini juga mengevaluasi efektivitas metode autentikasi dalam meningkatkan keamanan transaksi online dan mengurangi risiko kebocoran data [19][20]. Dengan metode ini, diharapkan sistem e-commerce dapat memberikan perlindungan optimal terhadap serangan siber dan menjaga kepercayaan pengguna.

C. Data

Menurut Pendit (1992), data adalah hasil observasi langsung terhadap suatu kejadian, yang merupakan perlambangan yang mewakili objek atau konsep dalam dunia nyata. Hal ini dilengkapi dengan nilai tertentu[21].

Menurut Ralston dan Reilly (Chamidi, 2004: 314), data didefinisikan sebagai fakta atau apa yang dikatakan sebagai hasil dari suatu observasi terhadap fenomena alam. Sebagai hasil observasi langsung terhadap kejadian atau fakta dari fenomena di alam nyata, data bisa berupa tulisan atau gambar yang dilengkapi dengan nilai tertentu[22].

D. E-Commerce

Menurut Akbar & Alam (2020), E-commerce merupakan pembelian, penjualan dan pemasaran barang dan jasa melalui sistem elektronik. Seperti televisi, radio, dan komputer atau jaringan internet. Transaksi bisnis yang terjadi di jaringan elektronik seperti internet. Setiap orang yang memiliki koneksi internet dapat berpartisipasi dalam kegiatan e-commerce[23]. Sedangkan pengertian e-commerce menurut Riswandi (2019) adalah satu set dinamis ekonomi, aplikasi dan proses bisnis yang menghubungkan perusahaan konsumen dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan/jasa dan informasi yang di lakukan secara elektronik[24].

E. Firebase Authentication

Firebase Authentication bertujuan untuk memudahkan pembangunan sistem autentikasi yang aman, sekaligus meningkatkan pengalaman login dan pengalaman aktivasi bagi pengguna akhir. Fungsi ini menyediakan solusi identitas end-to-end, mendukung login akun Email/sandi, autentikasi telepon, Google, Twitter, Facebook, GitHub, dan lainnya [25].

F. JavaScript

Javascript dibuat dan didesain oleh Brandan Eich, seorang karyawan Netscape pada bulan September 1995. Awalnya bahasa pemrograman ini disebut Mocha, kemudian berganti nama lagi menjadi Mona, lalu berganti lagi menjadi 12 Livescript, dan pada akhirnya menyandang nama menjadi Javascript[26].

G. Node.Js

Node.js merupakan salah satu peranti pengembang yang bisa digunakan untuk membuat aplikasi berbasis Cloud. Node.js dikembangkan dari engine JavaScript yang dibuat oleh google untuk browser Chrome / Chromium (V8) ditambah dengan libUV serta beberapa pustaka internal lainnya. Dengan menggunakan Node.js, semua pengembangan akan dilakukan menggunakan JavaScript, baik pada sisi klien maupun server. Node.js dibuat pertama kali oleh Ryan Dahl (twitter.com/ryah) dan sampai saat ini dikembangkan oleh komunitas sebagai perangkat lunak bebas dengan pendanaan utama dari Joyent, Perusahaan tempat Ryan Dahl bekerja [27].

H. React.js

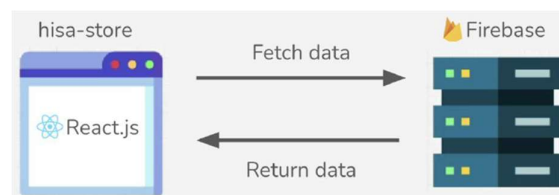
React.js atau React merupakan open-source Javascript library untuk mengembangkan antarmuka pengguna yang lebih interaktif dan mempermudah

developer dalam perancangan aplikasi. React.js digunakan untuk menangani view layer pada single page application dan aplikasi mobile. React.js dikelola oleh Facebook, Instagram, dan komunitas para developer [28]. React.js berusaha untuk memberikan kecepatan, kesederhanaan, dan skalabilitas. Beberapa fitur React.js yang biasa dikenal adalah JSX atau Javascript XML. JSX adalah extension untuk sintaksis ECMAScript. JSX membantu developer pada saat mengembangkan UI di dalam Javascript, dan juga dapat membantu developer ketika sedang melakukan error debugging. Konsep MVC atau Model View Control pada React.js hanya mempresentasikan bagian View saja, dan ini adalah bagian yang terbaik dari React.js karena lebih sederhana [29].

III. HASIL DAN PEMBAHASAN

A. Implementasi Tools

Pada Gambar 2 Skema Aplikasi Hisa Store, ditunjukkan skema implementasi aplikasi Hisa Store. Di sisi kiri, merupakan aplikasi Hisa Store yang dibangun menggunakan React.js, dan di sisi kanan, adalah Server Firebase yang menghosting database. Ketika pengguna ingin mengambil data, pengguna akan membuat permintaan dari aplikasi untuk mengambil data dari Firebase. Sedangkan React.js sebagai tools yang digunakan untuk membangun tampilan aplikasi atau front end akan mengatur state tampilan sesuai permintaan data oleh pengguna.



Gambar 2 Skema Aplikasi Hisa Store

Gambar tersebut menggambarkan penerapan pola arsitektur Client-Server modern. React.js berfungsi sebagai client-side rendering framework yang dinamis dan interaktif, sementara Firebase menggantikan peran server-side tradisional dengan layanan yang terkelola (managed services). Pola ini memisahkan dengan jelas concern antara presentation layer (React.js) dan data access layer (Firebase), yang merupakan prinsip dasar dalam pengembangan perangkat lunak yang terstruktur.

Data yang diminta dapat berupa profil pengguna, kategori produk, atau harga produk. Skema tersebut menunjukkan bahwa sebagian besar data aplikasi tidak akan disimpan di frontend karena pengguna yang menggunakan aplikasi berbeda-beda, sehingga data harus disimpan di database,

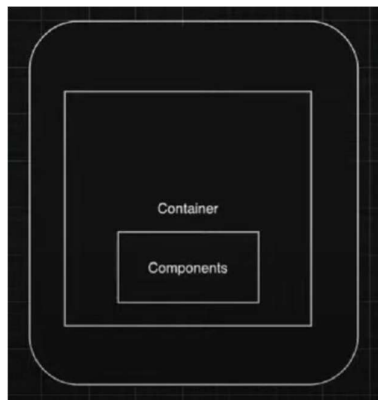
seperti Firebase. Namun, selain penyimpanan data, keamanan data sangat penting. Firebase, sebagai platform yang digunakan untuk menyimpan database, memiliki metode otentikasi untuk melindungi data sensitif.

B. Implementasi React.js

1) Konsep Dasar React.js

React atau ReactJS diciptakan oleh Facebook atau yang sekarang dikenal dengan Meta. Awalnya Facebook menggunakan bahasa pemrograman PHP dalam pembuatan frontend, dengan menggunakan engine jQuery. Namun Facebook kesulitan untuk menghandle sebuah website yang reaktif yang mempermudah user dalam menggunakan aplikasi sosial media mereka. Akhirnya dibuatkan library yang bernama React yang mampu

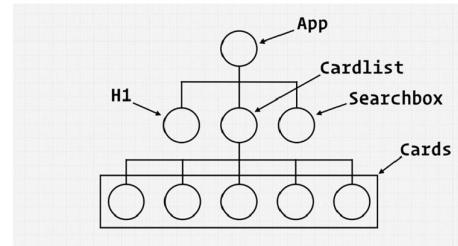
- a) *Component* : Di React, komponen adalah blok penyusun yang dapat digunakan kembali dan independen untuk membuat antarmuka pengguna. Komponen memungkinkan Anda membagi UI menjadi bagian yang independen dan modular, yang masing-masing dapat memiliki logika dan perilaku renderingnya sendiri. Aplikasi React biasanya dibangun menggunakan komposisi komponen-komponen ini.



Gambar 3 Ilustrasi Component

Gambar ini menggambarkan pola desain arsitektural di mana Container Components berperan sebagai pengelola data dan state, sementara Components (Presentation Components) berperan sebagai penerima dan penampil data. Penerapan pola ini mengarah pada pembuatan kode yang lebih terstruktur, modular, dan dapat dipelihara dalam pengembangan aplikasi berskala besar.

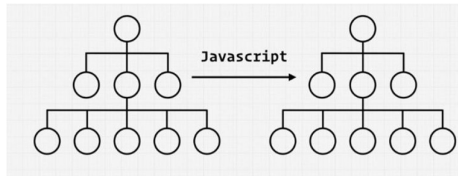
- b) *Document Object Model* : DOM (Document Object Model) merupakan susunan hirarki elemen-elemen HTML atau juga komponen-komponen yang membentuk tampilan website. Pada gambar Gambar 4 Ilustrasi DOM Tree, kita dapat melihat ilustrasi dari DOM.



Gambar 4 Ilustrasi DOM Tree

DOM memiliki struktur hirarki. Sehingga perubahan data pada Cards akan diterima oleh App dengan cara yang linear. Perubahan Cards akan diterima terlebih dahulu oleh Cardlist, setelahnya baru perubahan tersebut diterima oleh App. Perubahan yang dimaksud adalah bertambah atau berkurangnya komponen Cards dan Perubahan Properti Cards. Kegiatan ini memakan banyak resource dari segi waktu dan memori penggunaannya. Sehingga pada teknologi React, perubahan komputasi data ini akan diproses pada VDOM (Virtual Document Object Model).

- c) *Virtual Document Object Model* : Virtual DOM (Virtual Document Object Model) adalah bentuk kecil dari DOM (Document Object Model). Saat terjadi perubahan data, React membuat perubahan tersebut yang di proses pada "background". React akan membuat duplikasi dari DOM asli dalam bentuk Javascript dimana kode dalam bahasa Javascript ini akan merepresentasikan DOM yang asli. Dengan demikian perubahan komputasi data akan lebih mudah dilakukan tanpa harus melalui DOM asli terlebih dahulu. Secara sederhana duplikasi tersebut dapat dilihat pada Gambar 5 Ilustrasi Virtual DOM.

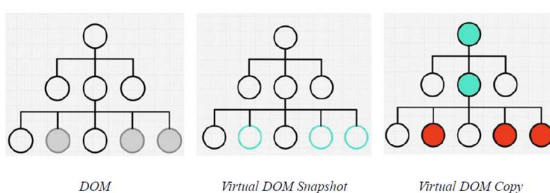


Gambar 5 Ilustrasi Virtual DOM

gambar ini dapat ditafsirkan sebagai ilustrasi manipulasi dan rekonstruksi struktur data tree menggunakan bahasa pemrograman JavaScript. Proses tersebut mencerminkan kemampuan JavaScript dalam melakukan operasi traversal, reformatting, dan restructuring terhadap data hierarkis. Aplikasi praktis dari konsep ini dapat ditemukan pada representasi Document Object Model (DOM) di dalam pengembangan web, di mana elemen-elemen HTML disusun dalam bentuk tree, lalu dimanipulasi untuk menghasilkan tampilan atau struktur baru sesuai kebutuhan.

VDOM terbagi menjadi dua segmen DOM yaitu Virtual DOM Snapshot, dan Virtual DOM Copy. Gambar 6 Ilustrasi Transisi Perubahan DOM Data, terlihat segmentasi dan partisi DOM pada teknologi react terbagi menjadi DOM (DOM yang asli), Virtual DOM Snapshot, dan Virtual DOM Copy. Pada saat terjadi perubahan data, maka data akan diubah pertama kali pada stage Virtual DOM Copy.

Setelah proses dilakukan pada Virtual DOM Copy maka React akan melakukan komparasi antara Virtual DOM Snapshot dengan Virtual DOM Copy. React seakan-akan melakukan penandaan dan validasi pada Virtual DOM Snapshot terkait data atau property yang berubah. Setelah validasi ini dilakukan, maka Virtual DOM akan melakukan transisi informasi yang telah terjadi ke DOM asli. Dengan demikian DOM asli hanya akan perlu melakukan satu kali perubahan data saja dan hanya me-render ulang komponen halaman yang mengalami perubahan data.



Gambar 6 Ilustrasi Transisi Perubahan DOM Data

Gambar ini menjelaskan bagaimana Virtual DOM bekerja sebagai lapisan abstraksi yang memungkinkan pembaruan elemen web secara declarative dan optimal, dengan menekan biaya komputasi akibat manipulasi langsung pada DOM nyata. Konsep ini merupakan salah satu landasan utama dalam performa tinggi framework modern seperti React, Vue, dan Angular.

- d) *State* atau *Hooks* : Di React, State dan Hooks memegang peranan penting dalam pengelolaan keadaan (state) komponen. State, sebagai data yang dimiliki oleh komponen, berfungsi untuk menyimpan informasi yang cenderung berubah seiring waktu, seperti input pengguna, hasil panggilan API, atau status aplikasi. Dengan bantuan fungsi `setState`, kita dapat memperbarui state, dan React secara otomatis akan menyelaraskan tampilan dengan perubahan tersebut. Sementara itu, Hooks, yang diperkenalkan dalam React versi 16.8, memungkinkan penggunaan state dan fitur React lainnya dalam komponen fungsional. Melalui hooks seperti `useState`, komponen fungsional dapat memiliki state, sementara hooks lain seperti `useEffect` digunakan untuk mengelola efek samping, dan `useContext` memungkinkan akses konteks. Dengan memanfaatkan state dan hooks, pengembang dapat menciptakan aplikasi React yang dinamis dan mudah dipelihara, membawa pengalaman pengguna yang lebih konsisten dan responsif.
- e) *Router* : Dalam pengembangan aplikasi web menggunakan React, pengelolaan navigasi antar halaman merupakan aspek penting untuk memastikan pengalaman pengguna yang responsif dan intuitif. Untuk mencapai hal ini, penggunaan React Router menjadi krusial. React Router adalah library yang memberikan kemampuan untuk mengatur rute dan navigasi dalam aplikasi React, memungkinkan pengembang untuk membuat tautan

antar halaman tanpa memuat ulang seluruh konten. Komponen-komponen inti seperti `BrowserRouter`, `Route`, `Link`, dan `Switch` memungkinkan pengembang untuk menentukan rute yang berbeda dan menyesuaikan tampilan komponen berdasarkan URL yang sesuai. Dengan demikian, pengguna dapat berpindah antar halaman dengan mulus dan cepat, meningkatkan keseluruhan pengalaman pengguna dalam menggunakan aplikasi web.

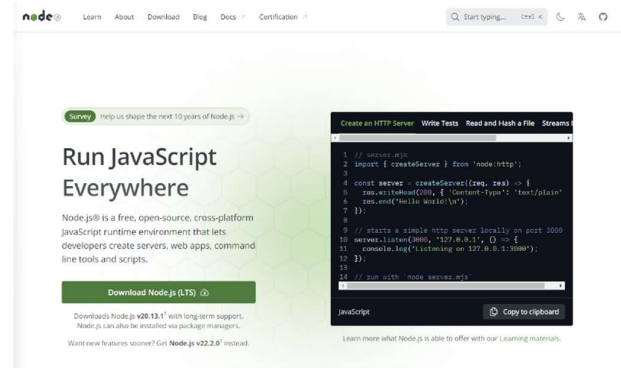
f) *Props* : *Props (singkatan dari properties) adalah cara dalam React untuk mengirim data dari satu komponen ke komponen lainnya. Props adalah objek yang berisi nilai-nilai yang diberikan oleh komponen induk kepada komponen children. Menggunakan props memungkinkan pengembang untuk membuat komponen yang dapat digunakan kembali dengan berbagai nilai input. Props bersifat read-only di komponen yang menerimanya, yang berarti mereka tidak dapat diubah di dalam komponen tersebut.*

2) Project Set Up

Banyak platform yang dapat digunakan untuk melakukan pembangunan proyek React. Dua diantaranya yang paling populer adalah CRA (Create React Application) dan Vite. CRA merupakan official generator untuk membuat proyek React yang direkomendasikan langsung dari Developer React. Sedangkan Vite merupakan sebuah community generator yang bisa membuat beberapa Javascript library seperti NextJS, VueJS dan lain-lain. Pada proyek ini pembangunan proyek dilakukan dengan menggunakan CRA. Selain karena CRA merupakan official generator CRA memiliki dokumentasi yang lebih lengkap dan lebih reliable.

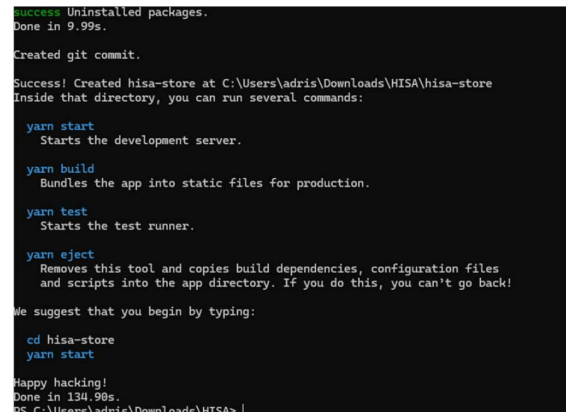
Memulai menggunakan React, dapat dimulai dengan melakukan instalasi NodeJS. NodeJS merupakan sebuah virtual environment yang digunakan sebagai engine untuk membuat virtual server pada lokal komputer dengan arsitektur v8 engine yang terintegrasi oleh web browser. Instalasi Node JS dapat dilakukan melalui tautan berikut <https://nodejs.org/en>. Tampilan

website Node JS dapat dilihat pada Gambar 7 Tampilan Website Node JS.



Gambar 7 Tampilan Website Node JS

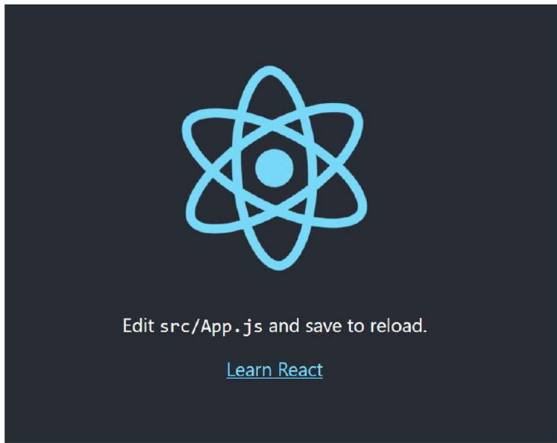
Jika proses unduh dan instalasi sudah selesai, selanjutnya membuka terminal windows lalu ketikkan `yarn create react-app hisa-store`. Proses instalasi React akan berjalan secara otomatis. Jika sudah selesai maka akan muncul return pada terminal seperti pada Gambar 8 Instalasi React Berhasil.



Gambar 8 Instalasi React Berhasil

Selanjutnya masuk ke direktori `hisa-store` dengan mengetik `cd hisa-store` pada terminal, kemudian masuk ke Visual Studio Code dengan mengetik `code .` pada terminal lalu tekan enter.

Pada Visual Studio Code atau code editor, akan terlihat banyak folder dan file yang sudah disiapkan secara default. Untuk menjalankan localhost dapat diketikkan pada terminal `yarn start`. Lalu akan muncul localhost dengan port 3000. Pada browser dapat diketikkan `http://localhost:3000/` maka akan muncul halaman awal React dengan tampilan seperti Gambar 9 Tampilan Awal React.



Gambar 9 Tampilan Awal React

Pada tahap ini proyek dapat dimulai dengan menghapus dan memodifikasi beberapa file yang tidak dibutuhkan pada Tabel 1 File yang Dimodifikasi.

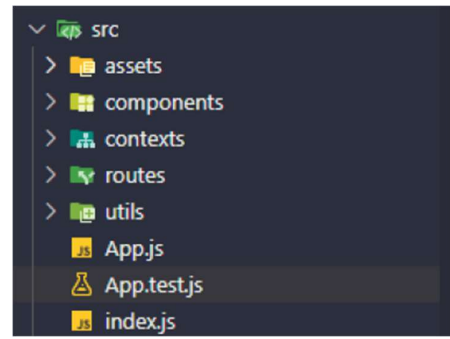
Tabel 1. File yang Dimodifikasi

No	File	Action
1	src/App.CSS	File dihapus karena akan menggunakan <i>styling component</i> .
2	src/App.js	Hapus component HTML pada return karena akan membuat halaman yang berbeda dari halaman default.

```

<div className="App">
  <header className="App-header">
    <img src={logo} className="App-logo" alt="logo" />
  </div>
  <div className="App-link">
    href="https://reactjs.org"
    target="_blank"
    rel="noopener noreferrer"
  >
    Learn React
  </div>
</div>
    
```

Pembuatan file .jsx harus disusun dengan rapih agar pengembang dapat dengan mudah melakukan debugging. Salah satu strategi yang dapat dilakukan adalah dengan memetakan files agar lebih strukturan dan independent. Pada Gambar 10 Struktur Folder Hisa Store dapat dilihat rekomendasi untuk penyusunan folder proyek website e-commerce Hisa Store.



Gambar 10 Struktur Folder Hisa Store

Dalam setiap folder terdapat kumpulan folder yang bersifat modular. Folder modular tersebut dihimpun berdasarkan kategorinya. Pada Tabel 2 Tabel Struktur Folder dapat dilihat pengelompokan file berdasarkan kebutuhan.

Tabel 2. Tabel Struktur Folder dan File


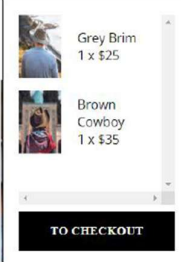
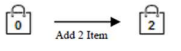


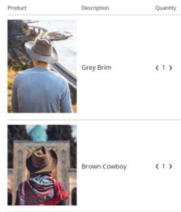

No	Derektori	Deskripsi
1	src/assets	Berisikan file statis seperti logo dan gambar dalam format .svg
2	src/components	Kumpulan komponen-komponen <i>reusable</i> .
3	src/contexts	Kumpulan global state yang mengatur jalannya informasi dari satu halaman ke halaman lain.
4	src/routes	Kumpulan kontainer dalam bentuk halaman.
5	src/utills	Kumpulan utilisasi yang salah satunya akan diisi oleh Firebase
6	src/App.js	Tempat mengatur lalu-lintas halaman <i>website</i> .
7	src/index.js	Tempat mengatur lalu-lintas data.
8	src/index.scss	Tempat mengatur <i>global style</i>


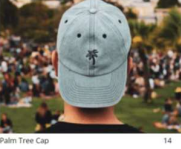
3) Component

Pada repository proyek website e-commerce Hisa Store, elemen-elemen pada tampilan website dibagi menjadi 12 komponen utama. Jumlah ini didapat berdasarkan jumlah komponen tampilan yang dinamis di setiap halaman website dibandingkan dengan halaman lainnya. Sehingga 12 komponen ini merupakan komponen yang reusable dan selalu digunakan di setiap halaman website. Penulisan nama komponen harus menggunakan Pascal Case. Daftar komponen yang dibuat dapat dilihat pada Tabel 3 Daftar Komponen. Pada setiap komponen terdapat styling component masing-

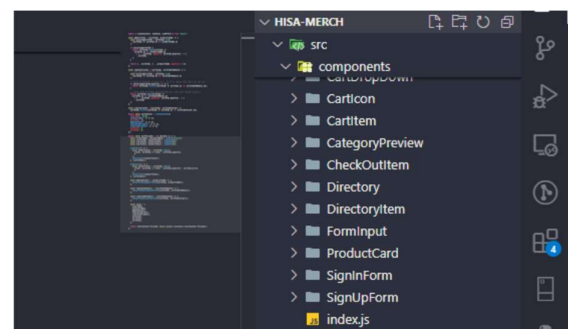
masing yang membuat design component yang dinamis

Tabel 3. Daftar Komponen

No	Nama Komponen	Deskripsi	Sample Gambar
1	Button	Tampilan dan fungsionalitas tombol untuk melakukan submit isian formulir.	
2	CartDropDown	Kontainer yang digunakan untuk menampung komponen CartItem.	
3	CartIcon	Fitur untuk memberikan akses ke halaman Checkout. Selain itu pada fitur ini terdapat indikator jumlah belanja yang dilakukan	
4	CartItem	Fitur ini digunakan untuk melihat preview belanjaan yang akan dibeli.	
5	CategoryPreview	Kontainer yang digunakan untuk menampung komponen ProductCard	
6	CheckoutItem	Kotainer yang berisikan konten barang belanjaan yang akan dibeli. Pada halaman ini pengguna dapat mengatur total barang belanja yang akan dibeli pada setiap jenis produk.	
7	Directory	Komponen yang memiliki layout grid. Komponen ini merupakan komponen yang akan digunakan	

		di halaman utama.	
8	DirectoryItem	Komponen yang akan mengisi konten pada komponen Directory.	
9	FormInput	Komponen yang digunakan untuk melakukan input oleh pengguna.	<input type="text" value="Email"/>
10	ProductCard	Komponen yang digunakan untuk menampilkan produk yang akan dijual. Pada komponen ini terdapat gambar dan teks yang digunakan sebagai informasi produk.	
11	SignInForm	Komponen ini digunakan untuk pengguna agar dapat melakukan login kedalam sistem.	<p>Already have an account Sign up with your email and password</p> <input type="text" value="Email"/> <input type="password" value="Password"/>
12	SignUpForm	Komponen ini digunakan untuk pengguna agar dapat melakukan pendaftaran akun member Hisa Store agar dapat login kedalam sistem.	<p>Don't have an account? Sign up with your email and password</p> <input type="text" value="Display Name"/> <input type="text" value="Email"/> <input type="password" value="Password"/> <input type="password" value="Confirm Password"/>

Pada repository proyek website e-commerce Hisa Store, komponen-komponen tersebut disimpan pada directory src/components. Direktori components dapat dilihat pada gambar Gambar 11 Direktori Folder Component. Pada directory folder component, terdapat satu file dengan ekstensi js yang bernama index.js.



Gambar 11 Direktori Folder Component

4) Props

Pada Gambar 12 Contoh Pembuatan Props pada Komponen CartItem merupakan penulisan kode untuk membuat komponen dinamis menggunakan props. Pada contoh berikut, props untuk komponen CartItem adalah cartItem yang memiliki data type object berisikan variabel imageUrl, price, name, dan quantity.

```
const CartItem = ({ cartItem }) => {
  const { imageUrl, price, name, quantity } = cartItem;

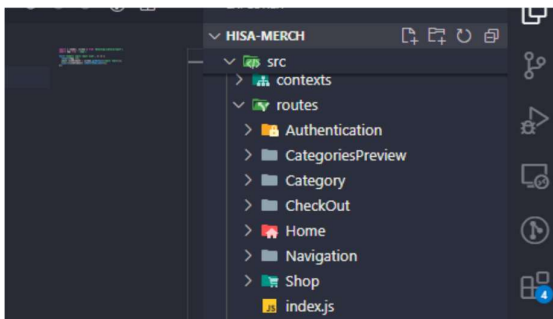
  return (
    <div className="cart-item-container">
      <img src={imageUrl} alt={` ${name} `} />
      <div className="item-details">
        <span className="name">{name}</span>
        <span className="price">
          {quantity} x {price}
        </span>
      </div>
    </div>
  );
};

export default CartItem;
```

Gambar 12 Contoh Pembuatan Props pada Komponen CartItem

5) Route

Route merupakan jalur dalam pengelolaan halaman website. Sebagai contoh pada route Category memiliki makna bahwa jika kita memasukkan alamat website pada web browser dengan alamat `http://localhost:3000/category`, maka akan muncul halaman kategori. Di sisi lain jika pada route Authentication kita memasukkan secara langsung route tersebut pada alamat web browser, kita akan kembali lagi ke halaman login. Hal tersebut dikarenakan untuk masuk ke halaman tersebut, dibutuhkan sebuah token authentication dari sisi backend untuk dilakukan validasi di sisi frontend. Struktur folder untuk routes dapat dilihat pada Gambar 13 Struktur Folder Routes.



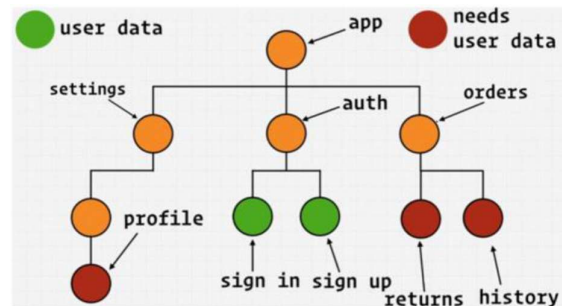
Gambar 13 Struktur Folder Routes

```
const App = () => {
  return (
    <Routes>
      <Route path="/" element={ <Navigation /> } />
      <Route index element={ <Home /> } />
      <Route path="shop/*" element={ <Shop /> } />
      <Route path="auth" element={ <Authentication /> } />
      <Route path="checkout" element={ <CheckOut /> } />
    </Routes>
  );
};
```

Gambar 14 Kode Routes

6) State atau Hooks

State atau hooks merupakan parameter yang menentukan kondisi atau status aliran data saat ini. State akan merubah kondisi props yang nantinya akan menjadi parameter pada komponen. State memiliki dua jenis bentuk, pertama adalah Global State, yang dapat menyimpan nilai kondisi untuk diberikan ke halaman lain yang tidak memiliki hubungan children-parent. Kedua adalah Local State yang dapat menyimpan nilai kondisi untuk diberikan ke halaman lain yang memiliki hubungan children-parent. Dengan demikian State atau Hooks inilah yang membuat React powerfull karena dengan menggunakan state rendering konten dapat dilakukan dengan sangat cepat, dinamis, dan optimal.

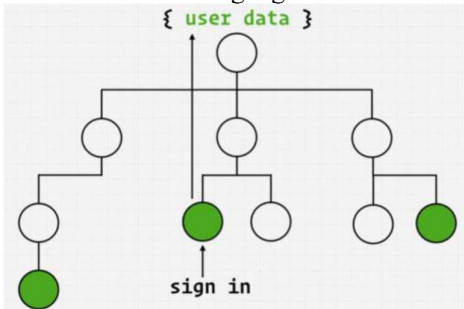


Gambar 15 Ilustrasi Distribusi Halaman dan Kebutuhan Data

Gambar ini memperlihatkan konsep data dependency dalam komponen aplikasi, yang sering muncul pada framework berbasis component-driven development seperti React.js. Pola ini juga menekankan pentingnya state management dalam mendistribusikan data pengguna ke berbagai komponen yang membutuhkannya. Tanpa manajemen data yang baik, proses pengiriman data antar komponen dapat menjadi kompleks (sering disebut prop drilling). Oleh karena itu, pendekatan ini biasanya dioptimalkan dengan menggunakan state management library

seperti Redux, Context API, atau MobX, agar distribusi data pengguna berlangsung lebih efisien dan terstruktur.

Kebutuhan data user pada halaman profile dan orders dapat dipenuhi dengan menggunakan global state atau context. Dari sini istilah global state akan menggunakan context agar pembahasan dapat berkoherensi dengan syntax pemrograman pada React yang menggunakan istilah context sebagai global state.



Gambar 16 Ilustrasi Mekanisme Global State atau Context

Pada Gambar diatas Ilustrasi Mekanisme Global State atau Context merupakan solusi untuk melakukan distribusi data. Context dapat digunakan sebagai sebuah variable yang berada di root atau parents dari virtual DOM sebuah aplikasi. Variabel tersebut dapat di-assign dengan sebuah nilai oleh sebuah children. Sehingga nilai yang diberikan oleh children dapat didistribusikan ke setiap halaman yang membutuhkan data tersebut.

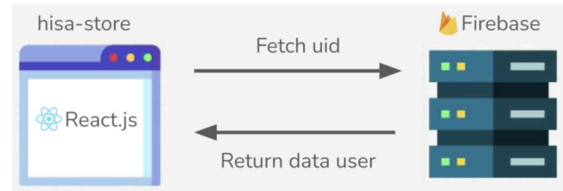
C. Implementasi Firebase

Dalam memproduksi sebuah aplikasi atau software, backend merupakan bagian dari software engineering yang mencakup kegiatan bussiness logic sebuah aplikasi. Baik pada authentication user, authorization user, password hasing, database, API setting, dan lain sebagainya. Pada saat membuat backend banyak sekali komponen fungsional yang harus dilakukan pengetesan. Seperti validasi API dengan berbagai method, koneksi aplikasi dengan database, pembuatan autorisasi pengguna, dan pengoalahan database. Sehingga backend memiliki peranan penting dalam sebuah aplikasi.

Namun pembuatan website e-commerce Hisa Store akan memanfaatkan teknologi yang dibuat oleh Google yang disebut dengan Firebase. Dengan menggunakan Firebase pembuatan database tidak harus lagi dibuat dari awal, seperti pembuatan dari server lokal, membuat port, dan lain sebagainya.

Dengan demikian Firebase dapat mempercepat kegiatan produksi dari sisi backend.

Pada Gambar 17 Skema Autentikasi Akun Member, merupakan skema dalam otentikasi pengguna menggunakan akun member. Melakukan login menggunakan akun member dilakukan secara langsung oleh frontend ke backend (firebase).



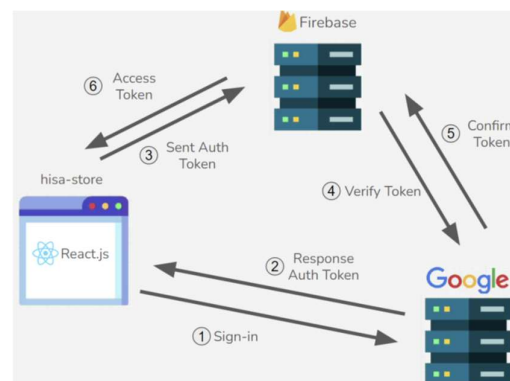
Gambar 17 Skema Autentikasi Akun Member

Gambar ini dengan efektif mengilustrasikan implementasi konkret dari pola Pemisahan Kepentingan (Separation of Concerns) dan arsitektur Klien-Server.

- React.js mengkhususkan diri pada presentation layer dan user interaction.
- Firebase mengkhususkan diri pada data persistence, authentication, dan business logic (melalui Cloud Functions).

Alur Feedback -> Fetch uid -> Return data user merupakan contoh operasi **READ** dalam paradigma **CRUD** (Create, Read, Update, Delete) yang fundamental. Visualisasi ini memberikan gambaran yang jelas tentang bagaimana data mengalir secara sekuensial dalam aplikasi modern, dari pemicu antarmuka hingga pengambilan data dari database dan kembali lagi untuk memperbarui antarmuka.

Berbeda dengan login menggunakan akun Google, prosesn otentikasi tidak hanya dilakukan antara aplikasi dengan firebase, namun juga dilakukan oleh Google. Pada Gambar 18 Skema Autentikasi Login Akun Google menunjukkan skema login menggunakan akun Google. Pada gambar tersebut, terdapat situs aplikasi Hisa Store, Firebase, dan Google di server Google.



Gambar 18 Skema Autentikasi Login Akun Google

Gambar ini menggambarkan alur otentikasi bertingkat yang mungkin diimplementasikan menggunakan Firebase Authentication. Prosesnya melibatkan pengiriman token, verifikasi, dan konfirmasi antara klien dan server. Kotak centang menandakan status penyelesaian setiap langkah, yang dapat digunakan untuk melacak kemajuan otentikasi. Protokol semacam ini meningkatkan keamanan dengan memastikan bahwa setiap langkah divalidasi sebelum melanjutkan ke langkah berikutnya, terutama untuk akses ke sumber daya yang sensitif.

Server Google memiliki banyak pengguna yang berbeda. Setiap pengguna memiliki data identitasnya berdasarkan data yang diberikan pada akun Google-nya pada saat mendaftar Gmail. Dengan demikian jika pengguna mencoba mengakses akun Google mereka dengan menggunakan email dan password yang benar maka Google akan mengetahui otentikasi pengguna tersebut dan akan memberikan respon berupa Token Authentication. Sehingga pengguna dapat mengakses dan mengelola akun Google tersebut. Google membantu pengguna mengautentikasi akun pengguna ke aplikasi lain sehingga pengguna tidak perlu membuat akun untuk aplikasi-aplikasi tersebut.

IV. KESIMPULAN

Berdasarkan kajian pada penelitian ini. Berikut kesimpulan yang didapat untuk menjawab permasalahan yang ditemukan. Pada Sisi Backend, Firebase Authentication sangat baik dalam melakukan autentikasi pengguna yang dapat dinilai pada hal berikut ini.

- Dengan Firebase Authentication, Hisa Store dapat menyimpan dan mengelola data pengguna dengan mudah dan aman.
- Proses implementasi melibatkan konfigurasi Firebase Authentication di aplikasi Hisa Store, yang memungkinkan pengguna untuk melakukan login melalui berbagai metode, seperti email dan kata sandi, serta akun Google.
- Setelah pengguna berhasil login, Firebase akan mengelola informasi otentikasi dan menyediakan token otentikasi untuk mengamankan akses ke data pengguna.

Pada Sisi Backend, Firebase Fire Storage sangat baik dalam melakukan pengguna penyimpanan database NoSQL yang dapat dinilai pada hal berikut ini.

- Firebase Firestore digunakan untuk menyimpan data Hisa Store dalam format NoSQL, yang memungkinkan penyimpanan dan penarikan data secara fleksibel dan skalabel.
- Setelah implementasi Firebase Firestore, data Hisa Store, seperti informasi produk, transaksi, dan detail pengguna, disimpan dalam koleksi dan dokumen Firestore.
- Dengan struktur penyimpanan NoSQL, Hisa Store dapat mengatur data dengan cara yang efisien dan dapat diperbarui secara real-time.

Pada Sisi frontend, React Js sangat baik dalam mengoptimalkan kemampuan rendering yang dapat dinilai pada hal berikut ini.

- Dalam mengimplementasikan ReactJS, Hisa Store dapat memanfaatkan kemampuan rendering yang cepat dan efisien.
- ReactJS memungkinkan Hisa Store untuk memisahkan logika aplikasi dari tampilan, dengan menggunakan komponen React untuk membuat antarmuka pengguna yang dinamis dan responsif.
- Dengan menggunakan teknik seperti virtual DOM dan reconciliation, ReactJS memastikan bahwa perubahan dalam data aplikasi tercermin secara efisien dalam tampilan, memungkinkan rendering yang cepat dan responsif.

Sistem ini memiliki kekurangan yaitu pembayaran yang belum dapat dilakukan. Selain itu sistem ini juga belum mendukung relational database sehingga format data base masih terlalu beragam. Namun kekurangan yang cukup signifikan bahwa sistem ini belum memiliki panel admin. Kapasitas admin harus memiliki latar belakang pemahaman firebase. Pencatatan rekap data baru dapat dilihat melalui firebase.

REFERENSI

- [1] Setiawan, A., & Ardhiyansyah, M. "Perancangan Aplikasi Web E-Commerce Penjualan Produk Asuransi Kendaraan Terintegrasi Api Midtrans Menggunakan Metode Extreme Programming (Studi Kasus: Pt. Komet Bersama Indonesia)". (2022).
- [2] Sentosa, H. P., Latief, F., & Agunawan, A. "Pengaruh Kepercayaan Dan Kemudahan Terhadap Keputusan Pembelian Marketplace E-Commerce Shopee". Nobel Management Review, 4(1), 105. <https://doi.org/10.37476/nmar.v4i1.3881>. (2023).
- [3] Kurniawati, D. T., & Yaakop, A. Y. "THE EFFECT OF E-SERVICESCPE DIMENSIONS ON CUSTOMER TRUST OF TOKOPEDIA E-STORE DURING COVID-19 PANDEMIC". Jurnal

- Aplikasi Manajemen, 19(1), 1. <https://doi.org/10.21776/ub.jam.2021.019.01.01>. (2021).
- [4] Fatrina, D., Kamil, I., & Hasan, A. “Pengaruh Live-Chat dan E-WOM Terhadap Pengambilan Keputusan Transaksi Online Pada Matketplace E-Commerce”. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 5(3), 121. <https://doi.org/10.25077/teknosi.v5i3.2019.121-129>. (2020).
- [5] Saputra, I. G. N. I., Sasmita, G. M. A., & Wiranatha, A. A. K. A. C. “Pengembangan Sistem Keamanan untuk E-Commerce”. *Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi)*, 17. <https://doi.org/10.24843/jim.2017.v05.i01.p03>. (2017).
- [6] Jonathan, J. M. C. “Peranan Komisi Pengawas Persaingan Usaha dalam Kegiatan yang Dilarang dalam Wujud Predatory Pricing Sektor E-Commerce”. *Jurist-Diction*, 6(2), 297. <https://doi.org/10.20473/jd.v6i2.43616>. (2023).
- [7] Maryati, W., & Masriani, I. “PELUANG BISNIS DI ERA DIGITAL BAGI GENERASI MUDA DALAM BERWIRUSAHA: STRATEGI MENGUATKAN PEREKONOMIAN”. *Jurnal Mebis (Manajemen Dan Bisnis)*, 4(2), 125. <https://doi.org/10.33005/mebis.v4i2.62>. (2019).
- [8] Pradesa, E., Syahrani, T., & Sakti, R. E. “Transformasi Digital Adopsi Software As a Service Layanan Cloud Accounting oleh UMKM”. *BUDGETING Journal of Business Management and Accounting*, 5(1), 155. <https://doi.org/10.31539/budgeting.v5i1.7049>. (2023).
- [9] Meliana, L., Fadhilah, M., & Cahyani, P. D. “Analisis Pengaruh Product Quality dan Social Media Marketing terhadap Consumer Loyalty dengan Customer Satisfaction sebagai Variabel Intervening pada Lawoek Coffee Temanggung”. *Al-Kharaj Jurnal Ekonomi Keuangan & Bisnis Syariah*, 5(5), 2627. <https://doi.org/10.47467/alkharaj.v5i5.2591>. (2023).
- [10] Siska, E., & Prapto, D. A. W. “Pelatihan Digital Marketing Pada Marketplace Sebagai Strategi Peningkatan Penjualan Produk UMKM Pulo Gadung Jakarta Timur Pada Masa New Normal”. *Jurnal Abdimas Perbanas*, 2(2), 59. <https://doi.org/10.56174/jap.v2i2.422>. (2021).
- [11] Wicaksono, S. R. “TRANSFORMASI DIGITAL - SUDUT PANDANG ANALISIS SWOT”. In Zenodo (CERN European Organization for Nuclear Research). European Organization for Nuclear Research. <https://doi.org/10.5281/zenodo.7703463>. (2023).
- [12] Fuadi, D. S., Akhyadi, A. S., & Saripah, I. “Systematic Review: Strategi Pemberdayaan Pelaku UMKM Menuju Ekonomi Digital Melalui Aksi Sosial”. *Diklus Jurnal Pendidikan Luar Sekolah*, 5(1), 1. <https://doi.org/10.21831/diklus.v5i1.37122>. (2021).
- [13] Wiranata, Moh. K., & Asmarudin, I. “Kebebasan Berekspresi Melalui Media Digital dan Penerapannya Di Indonesia”. (2023).
- [14] Fahrudi, M. A., & Suartana, I. M. “Integrasi Endpoint Security Berbasis Agent dan Bot Messenger untuk Deteksi dan Monitoring Serangan pada Web Server secara Real-time”. *Journal of Informatics and Computer Science (JINACS)*, 275. <https://doi.org/10.26740/jinacs.v4n03.p275-282>. (2023).
- [15] Joeliaty, J., Arrifin, S. D. S., & Sarasi, V. “PEMBERDAYAAN MASYARAKAT MELALUI EVALUASI PELATIHAN UMKM BERBASIS DIGITAL”. *Jurnal Kajian Budaya Dan Humaniora*, 5(1), 72. <https://doi.org/10.61296/jkbh.v5i1.107>. (2023).
- [16] Adam, S. I., Mononutu, M. J., & Damping, G. “Aplikasi Jasa Titip Belanja Berbasis Mobile di Minahasa Utara”. *CogITO Smart Journal*, 8(2), 434. <https://doi.org/10.31154/cogito.v8i2.422.434-445>. (2022).
- [17] A. R. Al Tahtawi and A. S. Rohman, “Perancangan Pengendali PI-AW pada Simulator Mobil Listrik untuk Kendali Kecepatan dan Torsi,” *Jurnal Teknik Elektro, Komputer, dan Informatika (ELECTRANS)*, vol. 14, no. 1, pp. 1-7, 2016.
- [18] Stefani, D. M., Jusia, P. A., & Mery. “Perancangan Sistem Informasi Transaksi Pembelian Dan Penjualan Beserta Persediaan Stok Produk Pada CV. Sumber Rezeki Jambi”. *Jurnal Manajemen Teknologi Dan Sistem Informasi (JMS)*, 4(2), 743–753. (2024).
- [19] Budiman, N. “Analisis Dan Perancangan Sistem Informasi Evaluasi Kinerja Organisasi Universitas Jambi Smart (Saestu) Menggunakan Model Prototype”. Universitas Jambi. (2020).
- [20] Plurilock. “Multi-factor authentication (MFA)”. (2024).
- [21] Teranova Security. “130 cyber security statistics: 2024 trends and data”. (2024).
- [22] Pedit, Putu Laxman. “Makna Informasi: Lanjutan dari Sebuah Perdebatan,” dalam *Kepustakawanan Indonesia: Potensi dan Tantangannya*, eds. Antonius Bangun dkk. Jakarta: Kesaint-Blanc. (1992).
- [23] Chamidi, Safrudin. (2004). “Kaitan antara Data dan Informasi Pendidikan dengan Perencanaan Pendidikan,” *Jurnal Pendidikan dan Kebudayaan* (48)10, hlm. 311—328
- [24] Akbar, M. A., & Alam, S. N. “E-Commerce Dasar Teori Dalam Bisnis Digital”. Yayasan Kita Menulis. (2020).
- [25] Riswandi, D. “TRANSAKSI ONLINE E-COMMERCE PELUANG DAN TANTANGAN DALAM PRESPEKTIF EKONOMI ISLAM”. *Jurnal Econetica*. (2019).
- [26] Aswin Wibisurya. “Analisis dan Perancangan Aplikasi FriendFinder Pemain Musik Berbasis Android (Online)”, <http://eprints.binus.ac.id/36418/>. (2020).
- [27] Rudjiono, D., & Saputro, H., “PENGEMBANGAN DESAIN WEBSITE SEBAGAI MEDIA

- INFORMASI DAN PROMOSI (Studi Kasus: PT.Nada Surya Tunggal Kecamatan Pringapus)". Pixel :Jurnal Ilmiah Komputer Grafis. <https://doi.org/10.51903/pixel.v13i2.300> . (2021).
- [28] Khat, T. "Developing a frontend application using ReactJS and Redux". (2018).
- [29] Nursaid, F. F., Brata, A. H., & Kharisma, A. P. "Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan ReactJS dan React Native Menggunakan Prototype (Studi Kasus: Toko Uda Fajri)". Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN, 2548, 964X. (2020).