

Pengembangan Aplikasi Sistem Informasi Manajemen di Jaringan CV Balibul Menggunakan Framework React Native

1st Samuel Robert Romulus Sidabalok
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
samuelsidabalok@student.telkomuniversity.ac.id

2nd Anggunmekha Luhur Prasasti
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
anggunmekha@telkomuniversity.ac.id

3rd Astri Novianty
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
astrinov@telkomuniversity.ac.id

Abstrak — Penelitian ini berfokus pada penggunaan React Native, sebuah framework open source yang dikembangkan oleh Facebook, untuk pengembangan aplikasi sistem informasi manajemen yang efisien. React Native memungkinkan pembuatan aplikasi mobile lintas platform dengan menggunakan JavaScript, yang menyederhanakan pengembangan dan mengurangi biaya serta waktu yang dibutuhkan. Dengan menggunakan pendekatan yang membedakan React Native dari metode pengembangan aplikasi mobile lain seperti aplikasi web mobile, HTML5, atau aplikasi hybrid, penelitian ini mengeksplorasi bagaimana aplikasi native dengan performa tinggi dapat dibangun menggunakan komponen yang dirancang khusus untuk iOS dan Android. Kajian ini menggunakan metode yang sistematis mulai dari perancangan hingga pengujian aplikasi, bertujuan untuk menilai efektivitas React Native dalam meningkatkan produktivitas tim pengembangan. Hasil dari penelitian ini diharapkan dapat memberikan wawasan yang mendalam tentang pengaruh React Native terhadap siklus pengembangan aplikasi dan bagaimana teknologi ini dapat diintegrasikan dalam strategi pengembangan teknologi informasi untuk mendapatkan keunggulan kompetitif.

Kata kunci— React Native, pengembangan aplikasi mobile, aplikasi sistem informasi manajemen, lintas platform, JavaScript.

I. PENDAHULUAN

Dalam era digital yang terus berkembang, pengembangan aplikasi mobile yang efisien dan efektif menjadi sangat penting bagi keberhasilan operasional suatu bisnis. Framework React Native, yang dikembangkan oleh Facebook, muncul sebagai solusi yang sangat mumpuni dalam memenuhi kebutuhan ini. React Native memungkinkan pengembangan aplikasi lintas platform menggunakan JavaScript, yang tidak hanya menghemat waktu tetapi juga biaya pengembangan, dengan memungkinkan kode yang sama digunakan pada platform Android, iOS, dan Windows. Dalam kajian ini, metode yang digunakan meliputi serangkaian langkah pengembangan sistematis, mulai dari perancangan awal hingga implementasi, dengan penekanan khusus pada prosedur pengujian untuk memastikan keandalan dan efektivitas aplikasi. Hasil dan pembahasan dalam penelitian ini akan

mengungkapkan bagaimana React Native dapat mempengaruhi siklus pengembangan aplikasi dan kontribusinya terhadap peningkatan produktivitas tim pengembang.

Oleh karena itu, penelitian ini bertujuan untuk mengevaluasi penggunaan React Native dalam konteks pengembangan aplikasi sistem informasi manajemen, dengan fokus pada bagaimana framework ini dapat memfasilitasi pengembangan yang lebih cepat, lebih ekonomis, dan lebih efisien. Ini akan memberikan wawasan yang lebih dalam tentang bagaimana React Native dapat diintegrasikan dalam strategi pengembangan teknologi informasi perusahaan untuk mendapatkan keunggulan kompetitif di pasar yang sangat kompetitif saat ini.

II. KAJIAN TEORI

A. Pengenalan React Native

React Native adalah sebuah framework open source yang dikembangkan oleh Facebook. Framework ini digunakan untuk membangun aplikasi mobile yang dapat berjalan di berbagai platform, termasuk Android, iOS, dan dalam tahap pengembangan untuk platform Windows. Keunikan React Native terletak pada penggunaan bahasa pemrograman JavaScript yang memungkinkan pengembang untuk menulis kode sekali dan menjalankannya di berbagai platform [1]. *Framework* ini berbeda dengan pendekatan lain seperti pembuatan aplikasi mobile web, aplikasi HTML5, atau aplikasi hybrid. Dalam React Native, Pengembang benar-benar membangun aplikasi mobile native, yang berarti aplikasi tersebut memiliki tampilan dan performa yang hampir identik dengan aplikasi yang dibangun menggunakan bahasa pemrograman native seperti Objective-C (untuk iOS) atau Java (untuk Android).

Framework React Native menyediakan berbagai perangkat komponen yang dirancang khusus untuk platform iOS dan Android, sehingga pengembang dapat membuat tampilan yang benar-benar seperti aplikasi native. Pengembang dapat merender User Interface (UI) dengan menggunakan JavaScript dan *framework* ini menggunakan blok bangunan UI dasar yang sama seperti aplikasi iOS dan Android, sehingga proses pengembangan lebih cepat dan

efisien karena pengembang dapat memanfaatkan komponen yang telah ada.

B. Bahasa Pemrograman JavaScript

JavaScript adalah bahasa pemrograman yang sangat serbaguna dan menjadi kekuatan utama di balik pengembangan web modern. Tidak hanya digunakan untuk membuat dan mengelola website, JavaScript juga memiliki kemampuan untuk membangun aplikasi mobile melalui framework seperti React Native, yang memungkinkan pengembang untuk menggunakan satu kode dasar untuk berbagai platform. Ekosistem JavaScript sangat kaya dengan berbagai perpustakaan dan kerangka kerja, seperti Angular, Vue.js, dan Node.js, yang semuanya memperluas kemampuan bahasa ini dan membuat proses pengembangan menjadi lebih efisien dan menyenangkan [2].

Sebagai bahasa pemrograman yang dinamis, JavaScript memungkinkan pengembang untuk menciptakan aplikasi web yang interaktif dan dinamis. Hal ini mencakup penambahan fitur-fitur menarik seperti animasi yang halus, kemampuan interaksi pengguna yang responsif, dan manipulasi data secara real-time. Dalam konteks pengembangan web, JavaScript digunakan untuk membuat halaman web yang tidak hanya statis, tetapi juga bisa merespon dan berinteraksi dengan pengguna secara langsung. Misalnya, dengan JavaScript, pengembang dapat membuat formulir yang dapat memvalidasi input secara langsung tanpa perlu memuat ulang halaman, atau membuat galeri gambar yang dapat berubah dengan mulus sesuai dengan interaksi pengguna. Semua ini menjadikan JavaScript sebagai bahasa yang esensial dan tak tergantikan dalam dunia pengembangan aplikasi web dan mobile modern.

C. Visual Studio Code

Visual Studio Code adalah sebuah platform yang berfungsi sebagai editor kode yang ringan namun sangat kuat, dirancang untuk berjalan pada berbagai sistem operasi desktop, termasuk Windows, macOS, dan Linux. Editor ini dikenal karena kemampuannya yang luar biasa dalam mendukung pengembangan perangkat lunak dengan berbagai fitur canggih yang memudahkan pengembang dalam menulis, mengedit, dan memanipulasi kode.

Salah satu keunggulan utama dari Visual Studio Code adalah dukungan built-in yang diberikan untuk beberapa bahasa pemrograman populer seperti JavaScript, TypeScript, dan Node.js. Dukungan ini memungkinkan pengembang untuk mulai bekerja dengan bahasa-bahasa tersebut tanpa perlu menginstal tambahan ekstensi atau konfigurasi yang rumit. Selain itu, Visual Studio Code memiliki ekosistem ekstensi yang sangat kaya dan luas, yang memberikan dukungan untuk berbagai bahasa pemrograman lainnya seperti C, C#, C++, Java, dan Python [3].

Bukan hanya bahasa pemrograman, Visual Studio Code juga menawarkan dukungan untuk berbagai runtime dan lingkungan pengembangan lainnya, termasuk .NET dan Unity. Ini memungkinkan pengembang untuk bekerja dalam berbagai proyek pengembangan perangkat lunak, mulai dari aplikasi web dan mobile hingga permainan dan aplikasi desktop, semuanya dalam satu editor yang konsisten dan kuat. Visual Studio Code juga dilengkapi dengan fitur-fitur canggih seperti debugging, kontrol versi terintegrasi, dan terminal built-in, yang semuanya dirancang untuk

meningkatkan produktivitas pengembang dan membuat proses pengembangan menjadi lebih efisien dan menyenangkan. Dengan berbagai fitur dan dukungan yang ditawarkannya, Visual Studio Code telah menjadi pilihan utama bagi banyak pengembang di seluruh dunia dalam upaya mereka untuk menciptakan perangkat lunak yang berkualitas tinggi.

D. Figma

Figma adalah sebuah platform desain yang beroperasi berbasis cloud, dirancang khusus untuk memenuhi kebutuhan para desainer UI/UX dalam mengembangkan antarmuka untuk aplikasi atau situs web. Platform ini dapat diakses melalui berbagai sistem operasi seperti Windows, Linux, dan macOS, asalkan terhubung ke internet. Figma kerap menjadi pilihan utama bagi profesional yang bergerak di bidang desain UI/UX, desain web, dan bidang terkait lainnya, berkat fitur-fiturnya yang lengkap yang mirip dengan yang ditawarkan oleh Adobe XD.

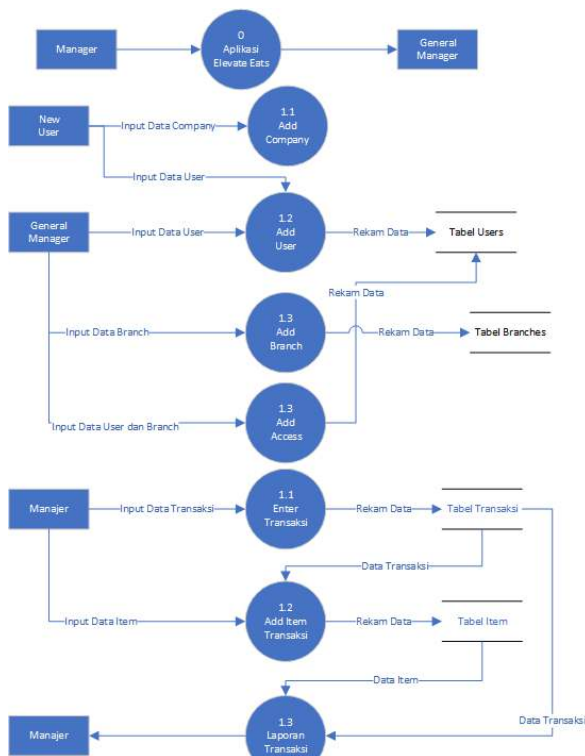
Salah satu keunggulan terbesar dari Figma adalah kemampuannya untuk memungkinkan kolaborasi secara real-time antar desainer yang berada di lokasi yang berbeda. Ini memungkinkan lebih dari satu orang untuk bekerja pada proyek yang sama secara bersamaan, yang sangat efisien untuk pengembangan prototipe situs web atau aplikasi secara cepat dan efektif. Kemampuan kolaboratif ini menjadikan Figma sebagai alat yang sangat berharga dan banyak dipilih oleh para desainer UI/UX untuk menciptakan desain yang inovatif dan responsif dalam waktu yang relatif singkat. [4]

III. METODE

Bagian ini akan menjelaskan mengenai implementasi framework yang digunakan pada aplikasi Elevate Eats sebagai Sistem Informasi Manajemen restoran CV Balibul.

A. Data flow diagram

Data flow diagram digunakan untuk menggambarkan bagaimana data bergerak dan bertransformasi dalam sebuah aplikasi [5]. Berikut ini adalah representasi dari data flow diagram yang mengilustrasikan proses flow data mulai dari fitur pendaftaran, penambahan cabang, hingga kegiatan transaksi dalam aplikasi tersebut.

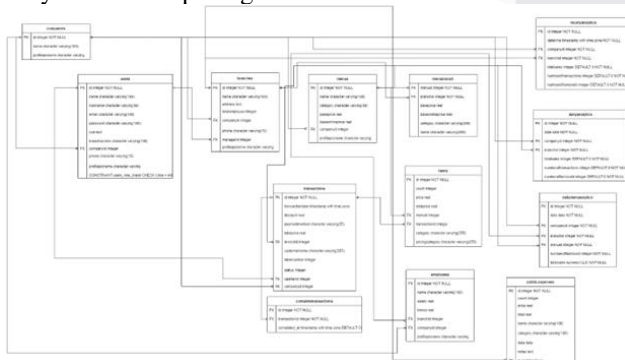


GAMBAR 1
Data flow diagram

Pada gambar data flow diagram diatas, terdapat dua aktivitas utama, yaitu registrasi dan transaksi. Dalam aktivitas registrasi, ada dua jenis yang terlibat. Jenis pertama adalah registrasi pengguna baru, yang mencakup pendaftaran perusahaan baru. Untuk jenis registrasi kedua, ini dilakukan oleh General Manager, yang memiliki kemampuan untuk mendaftarkan akun manajer baru serta memberikan akses dan menambahkan cabang. Registrasi yang dilakukan oleh General Manager ini tidak menciptakan perusahaan baru, melainkan menambahkan akun ke perusahaan yang sudah dikelola oleh General Manager tersebut.

Pada segmen transaksi, manajer bertanggung jawab untuk memasukkan data transaksi dan data item. Data transaksi tersebut kemudian akan disimpan dalam tabel transaksi, sedangkan data item akan membawa informasi transaksi tersebut untuk disimpan di tabel item. Informasi ini selanjutnya diproses menjadi laporan transaksi, yang dapat diakses oleh semua manajer yang memiliki akses ke cabang tersebut.

B. Entity Relationship Diagram



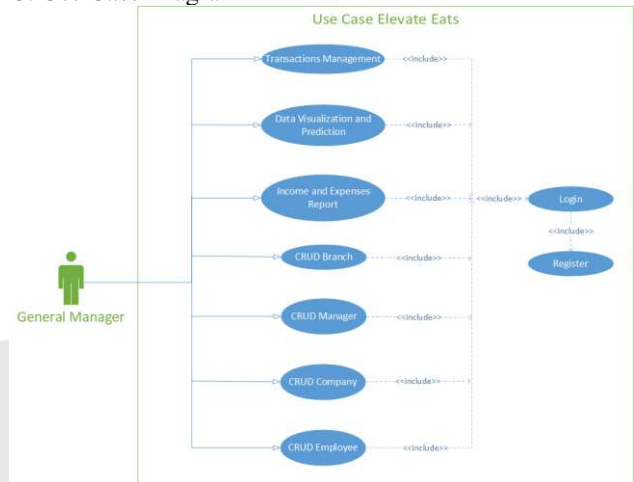
GAMBAR 2
Entity Relationship Diagram

Dalam proses pengembangan database untuk aplikasi, teknik yang sering diaplikasikan adalah Entity Relationship Diagram (ERD). ERD merupakan metode yang efektif untuk mendesain struktur database dengan menggambarkan hubungan antar entitas dalam database tersebut. Diagram ini bertujuan untuk secara visual menggambarkan entitas yang terlibat dalam database serta relasi yang terbentuk di antaranya, memberikan dasar yang kuat dalam membangun dan mengoptimalkan struktur data yang akan digunakan [6].

Dalam diagram entitas-relasi (ERD) untuk aplikasi yang direncanakan, terdapat tiga belas tabel yang berbeda. Tabel "Companies" berfungsi untuk menyimpan informasi mengenai setiap perusahaan yang terdaftar dan akan terhubung dengan tabel "Branches", yang bertugas menyimpan data mengenai cabang-cabang dari perusahaan tersebut. Tabel cabang ini juga akan terhubung dengan tabel "Menu", dimana menu bisa berbeda-beda tergantung pada cabangnya. Selain itu, tabel ini akan terhubung dengan tabel "Employees", yang mencatat data karyawan.

Selanjutnya, tabel "Users" akan menyimpan informasi tentang semua pengguna aplikasi beserta akses yang mereka miliki. Transaksi yang dilakukan akan dicatat dalam tabel "Transactions" dan "Items". Untuk pencatatan pengeluaran, digunakan tabel "Expenses". Terdapat juga beberapa tabel yang dikhususkan untuk keperluan analitik dan pemantauan, yaitu tabel "CompleteTransactions", "HourlyAnalytics", "DailyAnalytics", dan "DailyItemsAnalytics", yang masing-masing memiliki fungsi khusus dalam mengolah dan menganalisis data transaksi harian dan jam-jam tertentu.

C. Use Case Diagram



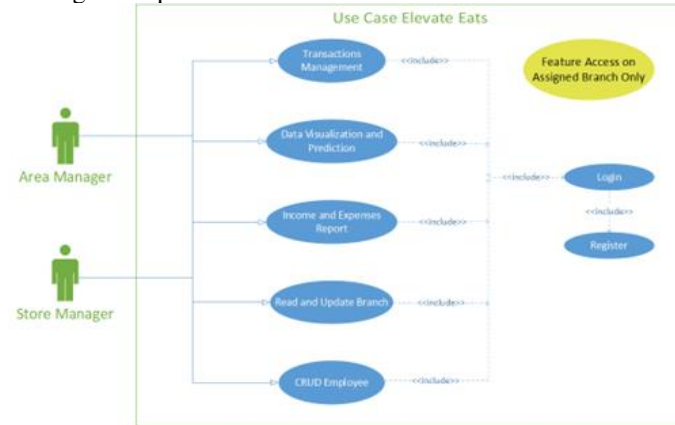
GAMBAR 3
Use case diagram pada role General Manager

Dalam diagram use case aplikasi Elevate Eats yang ditujukan untuk peran General Manager, terdapat beberapa fungsi utama yang dapat diakses. General Manager memiliki kemampuan untuk mengelola transaksi melalui "Transactions Management", yang termasuk dalam tugasnya. Mereka juga bertanggung jawab atas "Data Visualization and Prediction" yang membantu dalam analisis data dan perkiraan masa depan. Selain itu, General Manager dapat mengakses laporan "Income and Expenses Report" untuk memantau kinerja keuangan.

Dalam aspek pengelolaan sumber daya manusia dan operasional, General Manager memiliki akses untuk melakukan operasi CRUD (Create, Read, Update, Delete) pada entitas "Branch" (cabang), "Manager" (manajer),

"Company" (perusahaan), dan "Employee" (karyawan). Fungsi ini memungkinkan mereka untuk menambah, mengedit, melihat, dan menghapus data terkait dengan setiap entitas ini sesuai kebutuhan.

Selain itu, General Manager juga memiliki akses ke fungsi "Login" dan "Register", yang memungkinkan mereka untuk melakukan autentikasi dan mendaftarkan pengguna baru dalam sistem. Keseluruhan fungsi ini dirancang untuk memberikan kontrol penuh kepada General Manager atas operasi harian serta pengelolaan sumber daya manusia dan keuangan di aplikasi Elevate Eats.



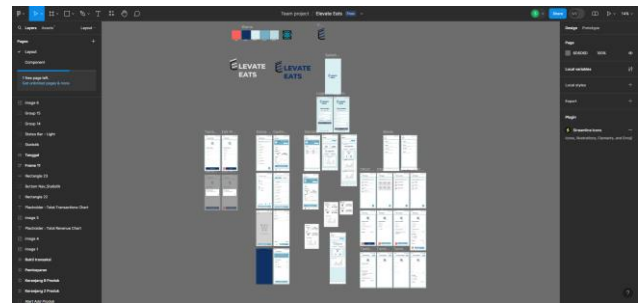
GAMBAR 4

Use case diagram pada role Area dan Store Manager

Dalam aplikasi Elevate Eats, peran Area Manager dan Store Manager mendukung fungsi operasional dan manajemen yang berbeda. Area Manager memiliki wewenang yang luas, termasuk mengelola transaksi melalui fungsi "Transactions Management" untuk berbagai cabang yang berada di bawah pengawasan mereka. Mereka juga dapat mengakses "Data Visualization and Prediction" untuk alat analisis dan proyeksi, serta "Income and Expenses Report" yang memberikan wawasan keuangan dari semua cabang yang mereka kelola. Selain itu, Area Manager dapat membaca dan memperbarui informasi cabang dan melakukan operasi CRUD pada karyawan di wilayah mereka.

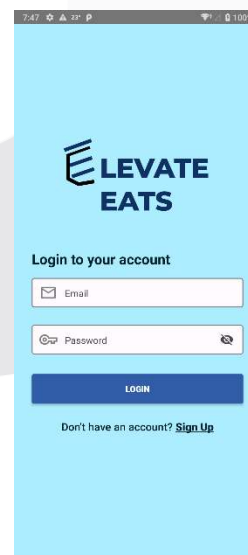
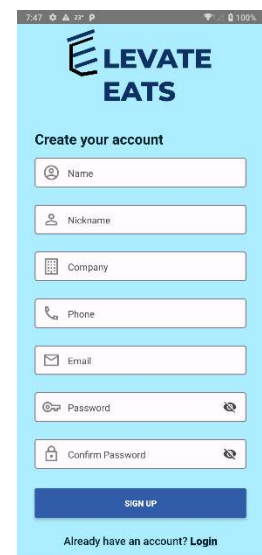
Sebaliknya, Store Manager mengelola operasi di tingkat cabang dengan akses terbatas pada "Transactions Management", "Data Visualization and Prediction", dan "Income and Expenses Report" yang khusus untuk cabang mereka. Mereka juga memiliki kemampuan untuk membaca dan memperbarui informasi tentang cabang yang mereka kelola serta mengelola data karyawan melalui fungsi "CRUD Employee". Baik Area Manager maupun Store Manager memiliki kemampuan untuk melakukan "Login" dan "Register", yang memungkinkan mereka mengakses sistem dan mendaftarkan pengguna baru sesuai kebutuhan, memastikan operasi cabang berjalan lancar dan aman.

D. User Interfaces

GAMBAR 5
User Interface

Alat atau tools yang digunakan untuk membuat desain antarmuka suatu aplikasi yaitu Figma. Penggunaan figma sebagai alat untuk membuat desain antarmuka karena aplikasi ini dapat digunakan melalui berbagai sistem operasi seperti Windows, Linux, dan macOS. Selain itu figma juga memiliki fitur yang lengkap seperti halnya aplikasi Adobe XD [4].

React Native digunakan sebagai framework UI karena React native memiliki keunikan yaitu penggunaan bahasa pemrograman JavaScript yang digunakan pengembang untuk membuat desain UI pada aplikasi. Sebagaimana yang diketahui, Javascript biasanya digunakan sebagai bahasa pemrograman untuk membuat desain pada sebuah website. Oleh karena keunikannya ini, pengembang dapat menulis kode sekali dan menjalakkannya di berbagai platform [1]. Selain itu, React native juga mendukung penggunaan komponen untuk setiap baris kode yang digunakan. Hal ini membuat kode terlihat lebih sederhana dan mudah untuk dipahami. Pengembang hanya perlu membutuhkan parameter dan argumen agar komponen tersebut dapat digunakan kembali pada halaman lainnya. Berikut merupakan implementasi bahasa pemrograman JavaScript pada framework React Native untuk membuat desain UI pada aplikasi:

GAMBAR 6
Tampilan UI halaman LoginGAMBAR 7
Tampilan UI halaman Register

Desain antarmuka pengguna (UI) dari aplikasi yang dirancang dengan mengadopsi prinsip-prinsip desain material yang telah ditetapkan dalam aplikasi desain Figma. Tujuan dari pendekatan ini adalah untuk mencapai tingkat

konsistensi yang tinggi, tidak hanya dalam hal visual tetapi juga dalam memberikan pengalaman pengguna yang mulus dan intuitif. Dalam pengembangan desain ini, setiap elemen UI, termasuk susunan tata letak, pemilihan palet warna, desain ikon, dan pemilihan jenis font, telah disesuaikan dengan teliti untuk memenuhi dan mendukung kebutuhan serta preferensi pengguna. Pendekatan ini memastikan bahwa aplikasi tidak hanya menarik dari segi estetika tetapi juga fungsional dan mudah digunakan, sehingga memaksimalkan kepuasan pengguna dalam interaksi sehari-hari mereka dengan aplikasi.

IV. HASIL DAN PEMBAHASAN

Pembuatan user interfaces pada pengembangan aplikasi menggunakan framework React Native dengan terintegrasi dengan endpoint dari ExpressJS.

A. Integrasi antara aplikasi dengan backend endpoint

Salah satu contoh integrasi antara aplikasi dengan backend adalah pada sistem login. User akan diminta untuk input email dan password yang telah tersedia. Kemudian aplikasi akan melakukan request ke backend dengan menggunakan method post.

Selain itu, dalam integrasi antara aplikasi dan backend diperlukan beberapa package tambahan yang diperlukan, misalnya perlu menambahkan library axios dan AsyncStorage. Library axios berguna untuk membantu dan mempermudah dalam melakukan integrasi atau meminta request ke backend. Pada dasarnya terdapat package yang sudah tersedia pada framework react native untuk melakukan request yaitu fetch(). Namun penggunaan axios lebih mudah baik secara sintaks maupun secara flow. Sedangkan package AsyncStorage berguna untuk menyimpan data pada penyimpanan local. Pengguna tidak harus melakukan login ulang ketika telah keluar dari aplikasi. Berikut merupakan implementasi integrasi antara aplikasi dan backend dengan melakukan request pada backend menggunakan method post.

KODE 1

request ke backend dengan menggunakan method post

```
const PostLogin = async () => {
  try {
    setLoading(true);
    const action = await
    axios.post(`${API_URL}/${LOGIN_ENDPOINT}`
    , login, {
      headers: {
        'Content-Type': 'application/json',
        apikey: API_KEY, // Ensure header keys
        are correctly expected by your backend
      },
    });
    if (action.data) {
      await AsyncStorage.setItem('userToken',
      action.data.token);
      await AsyncStorage.setItem(
        'companyId',

      action.data.credentials.companyid.toString
      g(),
    );
    await AsyncStorage.setItem(
      'credentials',

      JSON.stringify(action.data.credentials),
    );
    navigation.replace('Bottom Tab');
  }
}
```

```
ToastAndroid.show(action.data.message,
  ToastAndroid.SHORT);
}
} catch (error) {
  console.log('err: ', error);
  ToastAndroid.show(
    'Login Error, Check your email or
    password!',
    ToastAndroid.SHORT,
  );
} finally {
  setLoading(false);
}
};
```

Kemudian setelah pengguna berhasil login menggunakan email dan password yang valid, pengguna dapat melakukan CRUD (Create, Read, Update dan Delete). Berikut merupakan kode untuk melakukan fetching data atau mengambil data dari database dengan melakukan integrasi antara aplikasi dan backend.

KODE 2

kode mengambil data dari database

```
useFocusEffect(
  useCallback(() => {
    async function fetchData(params)
    {
      setLoading(true);
      try {
        const action = await
        GetData({
          operation:
          BRANCH_ENDPOINT,
          endpoint: 'showBranches',
          resultKey: 'branchData',
        });
        setBranch(action);
      } catch (error) {
        setError('Branch not
        found');
      } finally {
        setLoading(false);
      }
    }
    fetchData();
  }, []),
);
```

B. Role Pengguna

Aplikasi Elevete Eats memiliki tiga role yang berbeda diantaranya yaitu General Manager, Area Manager dan Store Manager. Setiap role memiliki fungsi dan tanggung jawab yang berbeda-beda. Misalnya pada role General Manager, role ini memiliki akses tertinggi dalam aplikasi. General Manager dapat mengakses semua fitur dan cabang yang tersedia. Kemudian untuk role Area Manager hanya dapat mengakses untuk beberapa fitur dan cabang saja. Mereka hanya memiliki hak akses pada cabang tertentu saja dan tidak dapat melakukan CRUD pada menu company. Sedangkan untuk role Store Manager seperti halnya role Area Manager. Namun, role Store Manager hanya dapat mengakses satu cabang saja.

TABEL 1
Role pengguna

Role	Hak Akses
General Manager	Melakukan manajemen transaksi
	Mengakses fitur visualisasi dan prediksi data
	Mengakses fitur laporan pendapatan dan pengeluaran
	Melakukan CRUD cabang

Area dan Store Manager	Melakukan CRUD menu company
	Melakukan CRUD pegawai
	Melakukan CRUD manager
	Melakukan manajemen transaksi
	Mengakses fitur visualisasi dan prediksi data
	Mengakses fitur laporan pendapatan dan pengeluaran
	Melakukan CRUD cabang
	Melakukan CRUD pegawai

C. Fitur Aplikasi

TABEL 2
Fitur aplikasi

User Interfaces	Deskripsi
Splash Screen	Muncul ketika ikon aplikasi diklik dan akan berjalan dan menampilkan logo
Login page	Menginputkan email dan password pengguna kemudian akan masuk ke Dashboard
Register page	Menginputkan email, password, nama, nickname dan phone untuk melakukan pendaftaran akun
Dashboard	Terdapat grafik prediksi dalam tujuh hari kedepan
Drawer Bar	Menampilkan sejumlah fitur yang terdapat pada aplikasi
Report	Menampilkan dua tombol yang akan melakukan navigasi ke halaman pengeluaran dan unduh laporan
Statistik	Menampilkan dua tombol yang akan melakukan navigasi ke halaman prediksi dan analisis data
Bisnis	Menampilkan tombol CRUD cabang, manager, pegawai dan menu company

D. Alpha Testing

Pengujian alfa merupakan pengujian yang dilakukan oleh pengembang aplikasi, bertujuan untuk memastikan bahwa aplikasi berfungsi dengan baik sebelum diluncurkan ke pengguna akhir. Metode yang digunakan dalam pengujian ini adalah pengujian Black Box, di mana fokus utamanya adalah pada fungsionalitas aplikasi tanpa perlu memahami struktur internal kode program. Tujuan dari pendekatan ini adalah untuk mendeteksi dan mengeliminasi setiap bug atau kesalahan yang mungkin mengganggu pengalaman pengguna, sehingga memastikan bahwa aplikasi berjalan dengan lancar dan efisien saat digunakan oleh pengguna sebenarnya. Dengan melakukan pengujian alfa secara menyeluruh, pengembang dapat mengidentifikasi masalah

yang tidak terlihat selama fase pengembangan dan mengatasi masalah tersebut sebelum aplikasi secara resmi dirilis. Berikut merupakan hasil pengujian alpha untuk beberapa bagian:

TABEL 3
Alpha Test

No	Section	Status
1	Authentication	Pass
2	Transaction	Pass
3	Reporting	Pass
4	Management	Pass
5	Analysis	Pass

V. KESIMPULAN

Penggunaan framework React Native dalam pengembangan aplikasi sistem informasi manajemen telah menunjukkan keunggulannya dalam mempercepat proses pengembangan dan meningkatkan produktivitas tim pengembang. Keunggulan utama dari React Native adalah kemampuannya untuk menyederhanakan proses pengembangan dengan mengizinkan penggunaan komponen yang dapat digunakan kembali di berbagai platform. Ini memungkinkan pengembang untuk menulis satu set kode yang hampir sama untuk iOS dan Android, yang secara signifikan mengurangi waktu yang diperlukan untuk pengembangan aplikasi lintas platform. Selain itu, React Native memfasilitasi pengembangan aplikasi dengan fitur hot reloading, yang memungkinkan pengembang untuk melihat perubahan langsung tanpa perlu mengulang memulai seluruh aplikasi, sehingga mempercepat proses iterasi. Keseragaman kode antara platform ini tidak hanya mempercepat waktu pengembangan tetapi juga mengurangi kesalahan yang umumnya terjadi saat mentransfer logika atau UI dari satu platform ke platform lain. Pengelolaan kode yang lebih mudah dan kemampuan untuk menerapkan pembaruan secara cepat dan efisien juga meningkatkan produktivitas tim. Selain itu, komunitas pengembang yang luas dan sumber daya yang melimpah memastikan dukungan terus-menerus serta akses ke berbagai library dan alat yang dapat digunakan untuk memperkaya dan mempercepat pengembangan aplikasi. Dengan demikian, penggunaan React Native tidak hanya meningkatkan kecepatan dan efisiensi tetapi juga secara substansial mengangkat kualitas akhir produk yang dikembangkan, menyediakan platform yang robust untuk pengembangan aplikasi sistem informasi manajemen yang inovatif dan responsif.

REFERENSI

- [1] D. A, S. N and M. I , "PERANCANGAN APLIKASI POINT OF SALES (POS) BERBASIS ANDROID MENGGUNAKAN FRAMEWORK FLUTTER DI KAFFE ELANGSTA," 2023.
- [2] A. Kholmatov, "Widely Used Libraries In The JavaScript Programming," *Intent Research Scientific Journal-(IRSJ)*, vol. 2, no. 10, pp. 18-25, 2023.
- [3] M. Romzi and B. Kurniawan, "Implementasi Pemrograman Python," *Journal of Innovation & Knowledge*, vol. XI, no. 2, pp. 1-9, 2020.
- [4] R. Puspita and R. Astriani, "Perancangan Design UI/UX Pada Website Toko Mister Shop ID," *Journal Teknik dan Science*, vol. II, no. 3, pp. 35-46, 2023.
- [5] S. H. B, "Data Flow Diagram and its Importance," 2020.
- [6] A. K, F. A. Z, A. A. D and R. D, "Analisi Teknik Entity-Relationship Diagram dalam Perancangan Database: Sebuah Literature Review," *JURNAL INTECH*, vol. III, no. 1, pp. 8-11, 2022.

