

Perancangan Aplikasi Key Performance Indicator (KPI) Dashboard Berbasis Android Menggunakan React Native dan Backend API Untuk Memantau Kinerja Gudang dan Penjualan Barang Dengan Metode Rapid Application Development (RAD) (Studi Kasus : PT. Mega Dagang Internasional)

Supriyadi¹, Mochamad Adhari Adiguna²

¹Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia, 15310
e-mail: ¹supriyadi.rsmi@gmail.com

² Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia, 15310
e-mail: ²dosen01864@unpam.ac.id

Abstract

In a company, Key Performance Indicator (KPI) is one of the important elements to support the company's operations. KPI is an indicator that can show the performance of a company whether it is in good condition or vice versa. KPIs can be accessed in various ways, one of which is through mobile application media in the form of dashboard. PT. Mega Dagang Internasional is a company engaged in retail. PT. Mega Dagang Internasional currently does not have the KPI dashboard system needed to help the company's operations, especially in the warehouse performance and sales of goods section. PT. Mega Dagang Internasional itself requires an easily accessible KPI system, so the mobile application is the right choice. The data collection method used is the case study method, as well as data collection techniques using observation and interview methods. The application development method itself uses the Rapid Application Development (RAD) method with the stages of needs analysis, application design, development process and gathering feedback, and application implementation or finalization. In order for the implementation of the Rapid Application Development method to be successful, the application development process will use the help of an Android-based mobile application framework, namely React Native.

Keywords : *Android application; Key Performance Indicator; RAD method; React Native; Dashboard*

Abstrak

Pada suatu perusahaan, Key Performance Indicator (KPI) merupakan salah satu elemen penting untuk menunjang operasional perusahaan. KPI merupakan indikator yang dapat memperlihatkan kinerja suatu perusahaan apakah sedang dalam kondisi baik ataupun sebaliknya. KPI dapat diakses dalam berbagai cara, salah satunya yaitu melalui media aplikasi mobile dalam bentuk dashboard. PT. Mega Dagang Internasional merupakan perusahaan yang bergerak di bidang retail atau penjualan segala jenis barang. PT. Mega Dagang Internasional saat ini belum memiliki sistem dashboard KPI yang diperlukan guna membantu operasional perusahaan khususnya di bagian kinerja gudang dan penjualan barang. PT. Mega Dagang Internasional sendiri membutuhkan sistem KPI yang mudah diakses, sehingga aplikasi mobile menjadi pilihan tepat. Metode pengumpulan data yang digunakan yaitu metode studi kasus, serta teknik pengumpulan data menggunakan metode observasi dan wawancara. Untuk metode pengembangan aplikasi sendiri menggunakan metode Rapid Application Development (RAD) dengan tahap analisa kebutuhan, desain aplikasi, proses pengembangan serta pengumpulan feedback, dan implementasi atau finalisasi aplikasi. Agar implementasi metode Rapid Application Development ini berhasil, maka dalam proses pengembangan aplikasi akan menggunakan bantuan framework aplikasi mobile berbasis Android yaitu React Native.

Kata kunci: Aplikasi Android; Key Performance Indicator; metode RAD; React Native; Dashboard

1. PENDAHULUAN

Perkembangan teknologi hingga saat ini sudah berkembang dengan pesat, khususnya pada bidang pengembangan aplikasi. Hal ini dibuktikan dengan banyaknya *framework* baru yang bermunculan. Salah satunya adalah *framework* pada bidang aplikasi *mobile* yang bernama *React Native*. *React Native* sendiri merupakan *framework* yang menggunakan bahasa pemrograman *JavaScript*, dimana aplikasi yang dibuat dengan menggunakan *React Native* ini dapat dijalankan di dua sistem operasi *mobile* yang berbeda yaitu *Android* dan *iOS* hanya dengan satu project aplikasi yang sama [1]. Saat ini masih ada perusahaan yang belum menggunakan aplikasi *mobile* di tengah era teknologi yang sedang berkembang dengan pesat, sehingga diharapkan dengan adanya teknologi *framework* terbaru pada bidang aplikasi *mobile* ini dapat mempercepat proses pengembangan aplikasi *mobile* agar dapat segera digunakan oleh perusahaan-perusahaan yang sedang membutuhkan.

Aplikasi *mobile* mempunyai peranan yang penting dalam menunjang kegiatan operasional perusahaan karena dapat digunakan juga sebagai penyedia informasi yang *real time*, mudah diakses kapanpun dan di manapun karena hanya membutuhkan *smartphone* beserta akses internet, dimana zaman sekarang mayoritas orang sudah menggunakan *smartphone* dan akses internet sudah mudah diakses. Salah satu penerapan aplikasi *mobile* pada perusahaan yaitu digunakan sebagai penyedia informasi *Key Performance Indicator* (KPI) dalam bentuk *dashboard*.

Key Performance Indicator yang selanjutnya penulis akan sebut sebagai KPI merupakan salah satu hal yang penting bagi suatu perusahaan. KPI dapat digunakan sebagai alat ukur untuk menentukan atau menilai kinerja perusahaan dalam berbagai bidang [2]. Dengan adanya implementasi KPI menggunakan aplikasi *mobile* ini diharapkan dapat membantu perusahaan dalam mengakses informasi KPI secara efisien, sehingga kualitas kinerja perusahaan akan terjaga dengan baik.

PT. Mega Dagang Internasional adalah sebuah perusahaan yang bergerak dalam bidang persediaan dan penjualan berbagai macam barang bekas maupun baru yang bertempat di Jl Raya Bogor KM 41,2 Cibinong, Pabuaran, Cibinong, Bogor, Jawa Barat. PT. Mega Dagang Internasional saat ini sudah memiliki sistem aplikasi berbasis web untuk mengakomodasi kebutuhan operasionalnya serta memiliki *Backend API* (*Application Programming Interface*) sendiri yang

bisa digunakan sebagai sumber data untuk aplikasi pihak ketiga [3], akan tetapi mereka masih belum mempunyai aplikasi *KPI Dashboard* berbasis aplikasi *mobile* terutama yang menggunakan sistem operasi *Android*, sehingga kesulitan dalam memantau kinerja perusahaan terutama kinerja gudang dan penjualan barang secara efisien.

Maka berdasarkan latar belakang yang telah diuraikan di atas, maka akan dilakukan penelitian untuk merancang aplikasi *KPI Dashboard* berbasis aplikasi *mobile*. Untuk sistem operasi *mobile* yang akan digunakan adalah *Android* dengan menggunakan *framework React Native* beserta *JavaScript* sebagai bahasa pemrograman dalam perancangan aplikasi tersebut.

2. METODE

Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini yaitu :

1. Metode Observasi

Metode pengumpulan data serta informasi dengan melakukan pengamatan secara langsung terhadap kegiatan-kegiatan yang berlangsung di tempat instansi yang bersangkutan.

2. Metode Wawancara

Metode pengumpulan data dengan melakukan wawancara secara langsung kepada pihak-pihak yang bersangkutan dengan penelitian yang akan dilakukan dan menggunakan pertanyaan-pertanyaan yang telah disiapkan oleh peneliti.

Metode Perancangan Aplikasi

Metode perancangan aplikasi yang digunakan pada penelitian ini adalah *Rapid Application Development* (RAD) dan menggunakan tools UML (*Unified Modelling Language*). RAD adalah suatu pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak (Kendall, 2010) [4]. RAD bertujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara perancangan dan penerapan suatu sistem informasi. Metode RAD sendiri memiliki tahapan-tahapan sebagai berikut :

1. Menentukan *project requirements*

Project requirements berisikan persyaratan yang harus dicapai dalam perancangan suatu aplikasi dan persiapan strategi untuk menghadapi permasalahan yang mungkin akan muncul nantinya. Persyaratan ini juga

mencakup *timeline* dan *budget* yang ada. Pihak-pihak yang terlibat dalam pembuatan *project requirements* adalah para *developer*, klien, dan juga *software user*. Semua pihak yang terlibat diharuskan untuk menyetujui persyaratan yang telah ditentukan di awal, agar tidak terjadi miskomunikasi dan dapat terhindar dari kesalahan-kesalahan yang akan menimbulkan kerugian baik dari segi waktu maupun biaya.

2. Membuat *prototype*

Perancang aplikasi akan mengembangkan prototipe dengan waktu yang singkat dengan fungsi aplikasi yang sesuai dengan keinginan klien. Prototipe ini akan diberikan kepada klien agar dapat mengetahui fungsi apa saja yang mereka butuhkan maupun tidak dibutuhkan, sehingga klien dapat mengetahui gambaran akhir seperti apa aplikasi yang akan mereka pakai nanti.

3. Pengumpulan *feedback*

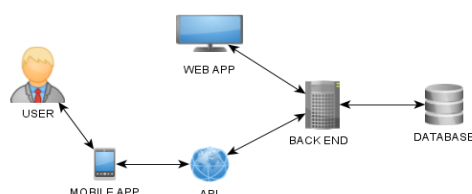
Pada tahap ini, perancang aplikasi akan melihat *feedback* yang telah disampaikan oleh klien. *Feedback* yang dimaksud di sini mencakup fungsi, fitur, visual, dan juga *user experience* dari aplikasi yang sedang dikembangkan. Tahap ini bisa dilakukan berulang kali sehingga hasilnya sesuai dengan keinginan klien dan dari tahap pembuatan prototipe bisa dilanjutkan ke tahap finalisasi atau penyelesaian akhir aplikasi.

4. Penyelesaian aplikasi

Tahap terakhir adalah implementasi hasil *feedback* dan membuat aplikasi yang siap digunakan sepenuhnya. Fungsi, fitur, visual, dan *user experience* akan dinilai kembali oleh klien. Pada tahap ini uji coba akan dilakukan jika memang dibutuhkan. Uji coba ini dilakukan untuk memastikan semua hal sudah terkontrol dan aplikasi sudah siap berlanjut ke tahap produksi dan digunakan sepenuhnya oleh klien.

Arsitektur Aplikasi

Arsitektur aplikasi menggambarkan secara keseluruhan mengenai desain arsitektur aplikasi sehingga memudahkan proses pengembangan aplikasi [5].



Gbr 1. Arsitektur aplikasi

Komponen *Request* dan *Response* API

Dibawah ini merupakan komponen *request* dan *response* API yang telah dibuat dari aplikasi *Backend*. Penulis telah menganalisa komponen *request* dan *response* tersebut agar mempermudah proses perancangan dan pengujian aplikasi. API dianalisa dengan menggunakan aplikasi *Postman* yang berisi permintaan data dan hasil kembalian yang di butuhkan untuk data indikator KPI sebagai berikut :

Table I. Rancangan *request* dan *response* login

Method	POST
URL	http://127.0.0.1:8000 /api/auth/login
Headers	-
Body	email : admin@example.com password : #blm-pos

Table II. Rancangan *request* dan *response* sales

Method	POST
URL	http://127.0.0.1:8000/api/sales
Headers	Authorization
Body	-
Keterangan	Endpoint API ini berfungsi untuk mendapatkan data meliputi bagian penjualan yang terdiri dari indikator total penjualan dalam seminggu, barang dengan penjualan tertinggi dan terendah, outlet cabang dengan penjualan tertinggi dan terendah
Response	{ "salesWeeklyData": ["467", "144", "232", "0", "0", "0", "0"], "bestSellerProduct": "KANTONG KRES EK UK:35", "lowestSellerProduct": "LAMPU", "highestOutletSales": "TINUQU SUMBEREJO", "lowestOutletSales": "TINUQU DONOMULYO" }

Table III. Rancangan *request* dan *response* warehouse

Method	POST
URL	http://127.0.0.1:8000/api/warehouse
Headers	Authorization
Body	-
Keterangan	Endpoint API ini berfungsi untuk mendapatkan data meliputi bagian gudang yang terdiri dari indikator total lalu lintas barang di outlet cabang (barang masuk dari pemasok, barang yang didistribusi, dan barang yang diterima oleh outlet dari gudang pusat)
Response	{ "salesOfItems": ["882",0, 0], "inbound": 299, "distributed": 0, "received": 0 }

Table IV. Rancangan *request* dan *response* dashboard

Method	POST
URL	http://127.0.0.1:8000/api/dashboard
Headers	Authorization
Body	-
Keterangan	Endpoint API ini berfungsi untuk mendapatkan data singkat untuk ditampilkan dalam halaman utama aplikasi yang merangkum lalu lintas barang masuk dan keluar, total penjualan, total barang yang tidak bisa dijual maupun di pakai kembali, dan total karyawan yang sedang bekerja. Data tersebut ditampilkan secara realtime dalam hari saat aplikasi sedang dibuka
Response	{ "totalWarehouse": 299, "totalSalesQty": 882, "totalSalesPrice": "23222800.00", "totalActiveUsers": [],

	"totalStockWaste": 0 }
--	------------------------

Table V. Rancangan *request* dan *response* staff KPI

Method	POST
URL	http://127.0.0.1:8000/api/staff
Headers	Authorization
Body	-
Keterangan	Endpoint API ini berfungsi untuk mendapatkan data meliputi kinerja karyawan yang terdiri dari indikator total aktivitas pekerjaan karyawan dalam seminggu, karyawan dengan poin tertinggi dan terendah, dan daftar lengkap karyawan yang sedang online lengkap dengan poin dan outlet asalnya
Response	{ "activityWeeklyData": [80, 214, 186, 0, 0, 0, 0], "highestStaffPoint": "Budi", "lowestStaffPoint": "Desi", "staffPointList": [...] }

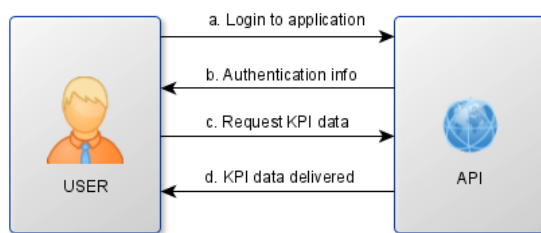
Table VI. Rancangan *request* dan *response* profile

Method	POST
URL	http://127.0.0.1:8000/api/users
Headers	Authorization
Body	-
Keterangan	Endpoint API ini berfungsi untuk mendapatkan data meliputi data profil pengguna yang sedang login yang terdiri dari foto profil beserta username
Response	{ "id": 1, "name": "Admin Example", "photo": "uploads/1/2020-06/whatsapp_image_2020_06_14_at_193257.jpeg",

	<pre> "email": "admin@example.com", "id_cms_privileges": 1, "outlet_id": null, "created_at": "2020-06-16T02:45:38.000000Z", "updated_at": "2021-01-13T02:06:42.000000Z", "status": "Active" </pre>
--	--

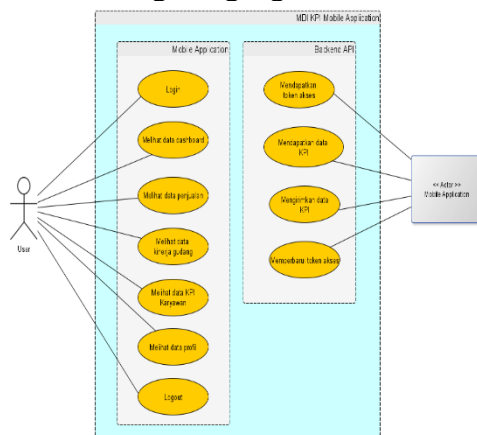
Rancangan Skenario Penggunaan API

Rancangan ini menggambarkan alur penggunaan aplikasi mulai dari pengguna login sampai dengan pengguna mengakses sumber data yang telah diolah di aplikasi *mobile* [6].



Gbr 2. Skenario aplikasi

Unified Modeling Language



Gbr 3. Use Case Diagram

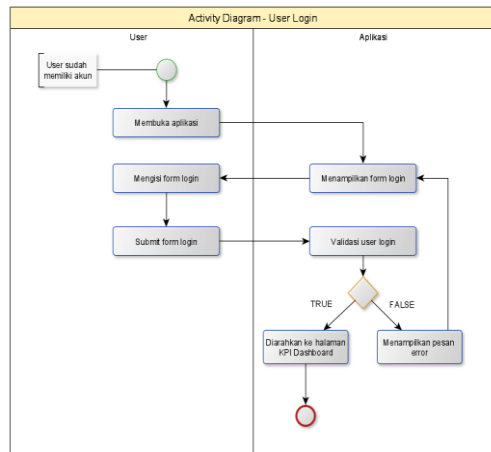
Berikut adalah deskripsi dari *use case diagram* dari gambar 3 sebagai berikut :

Table VII. Deskripsi *use case diagram*

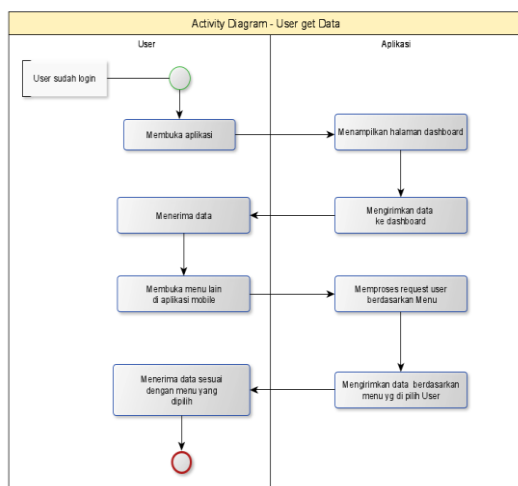
1.	Use case	Login
	Actor	User
	Deskripsi	User melakukan <i>login</i> dengan cara memasukkan <i>email</i> dan <i>password</i> agar dapat mengakses data di dalam aplikasi
2.	Use case	Melihat data dashboard

	Actor	User
	Deskripsi	User yang telah berhasil <i>login</i> dapat melihat data di dalam menu <i>dashboard</i>
3.	Use case	Melihat data penjualan
	Actor	User
	Deskripsi	User yang telah berhasil <i>login</i> dapat melihat data di dalam menu <i>sales/penjualan</i>
4.	Use case	Melihat data kinerja gudang
	Actor	User
	Deskripsi	User yang telah berhasil <i>login</i> dapat melihat data di dalam menu <i>warehouse/gudang</i>
5.	Usecase	Melihat data KPI Karyawan
	Actor	User
	Deskripsi	User yang telah berhasil <i>login</i> dapat melihat data di dalam menu <i>staff / karyawan</i>
6.	Usecase	Melihat data profil
	Actor	User
	Deskripsi	User yang telah berhasil <i>login</i> dapat melihat data di dalam menu <i>profil</i>
7.	Usecase	Logout
	Actor	User
	Deskripsi	User dapat melakukan <i>logout</i> setelah menekan tombol di menu <i>profil</i>
8.	Usecase	Mendapatkan token akses
	Actor	Mobile Application
	Deskripsi	Mobile Application mendapatkan token akses yang didapat dari API sebagai akses untuk mengizinkan user yang berhasil melakukan <i>login</i>
9.	Usecase	Mendapatkan data KPI
	Actor	Mobile Application
	Deskripsi	Mobile Application mendapatkan data KPI yang telah dikirim dari API jika token akses <i>valid</i>
10.	Usecase	Mengirimkan data KPI
	Actor	Mobile Application
	Deskripsi	Mobile Application mengirimkan data KPI ke bagian tampilan aplikasi agar bisa dilihat oleh user

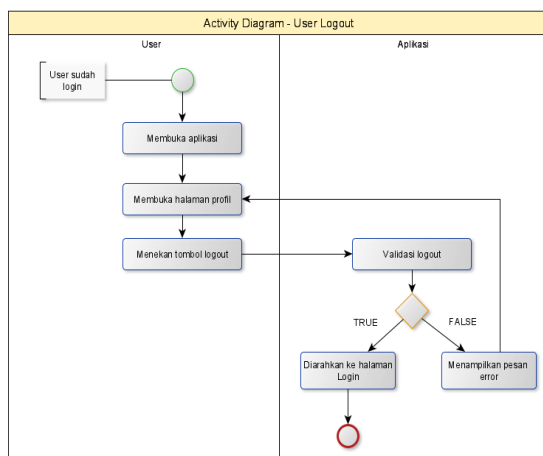
11	Usecase	Memperbarui token akses
	Actor	Mobile Application
	Deskripsi	Mobile Application dapat memperbarui token akses jika statusnya sudah expired



Gbr 4. Activity diagram user login



Gbr 5. Activity diagram user get data



Gbr 6. Activity diagram user logout

3. HASIL Spesifikasi

Aplikasi ini dibuat menggunakan bahasa pemrograman *JavaScript* didukung dengan *framework React Native* agar dapat mempercepat proses pembuatan aplikasi. Sesuai dengan analisa dan perancangan sebelumnya, bahwa dalam aplikasi ini terdapat beberapa halaman berisi tampilan dengan fungsi masing-masing.

Dalam pembuatan aplikasi ini menggunakan perangkat keras dengan spesifikasi sebagai berikut :

- Laptop model *Razer Blade 15*
- Prosesor Laptop *Intel Core i7 gen 8th*
- Storage Laptop *SSD 1 TB*
- RAM Laptop *16 GB*
- VGA Card *Nvidia Geforce RTX 2070*
- Smartphone *Samsung a52s 5G*
- Prosesor *smartphone Snapdragon 778G 5G*
- RAM *smartphone 8 GB*
- Storage *smartphone 128 GB*

Tahap implementasi adalah tahap dimana aplikasi telah dibuat dan siap untuk digunakan. Implementasi adalah tahap lanjutan setelah melakukan analisa dan perancangan aplikasi. Dalam tahap ini dapat diketahui apakah aplikasi telah sesuai atau tidak sesuai dengan apa yang sudah dirancang sebelumnya.

Implementasi Perangkat Keras

Dalam pengembangan aplikasi *KPI Mobile* ini dibutuhkan beberapa syarat minimum baik dari segi perangkat keras maupun perangkat lunak agar perancangan aplikasi berjalan dengan baik. Tujuan tahap implementasi ini yaitu untuk memastikan perangkat keras maupun perangkat lunak dapat berjalan dengan baik sesuai dengan yang diharapkan.

Table VIII. Implementasi Perangkat Keras

No	Perangkat Keras	Spesifikasi
1.	Smartphone	Samsung a52s 5G
2.	Processor smartphone	Snapdragon 778G 5G
3.	RAM smartphone	8 GB
4.	Storage smartphone	128 GB
5.	Laptop	Razer Blade 15
6.	RAM laptop	16 GB
7.	Processor laptop	Intel Core i7 gen 8 th
8.	Storage laptop	1 TB
9.	VGA Card laptop	Nvidia Geforce RTX 2070

Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk implementasi aplikasi adalah sebagai berikut :

Table IX. Implementasi Perangkat Lunak

No	Perangkat Lunak	Keterangan
1.	Android 13	Sistem operasi <i>smartphone</i>
2.	Android Studio	Aplikasi pendukung
3.	Windows 10 64bit	Sistem operasi laptop
4.	JavaScript	Bahasa Pemrograman
5.	React Native	Framework pendukung
6.	Java SDK	Aplikasi pendukung
7.	Android Kit	Aplikasi pendukung
8.	Android Virtual Device	Aplikasi pendukung
9.	NodeJS	Aplikasi Pendukung
10.	Postman	Aplikasi pendukung
11.	Visual Studio Code	Aplikasi editor bahasa pemrograman
12.	React Paper, React Native Chart, Tailwind CSS, & React Navigation	Library pendukung

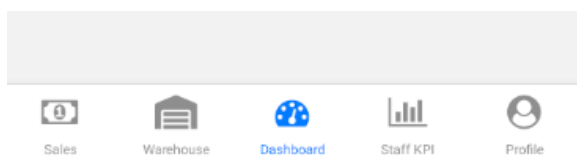
Implementasi Antarmuka

Implementasi antarmuka memiliki tujuan untuk membuat hasil aplikasi berupa tampilan dapat dilihat dan digunakan oleh user.

MDIKPI APP

Log in

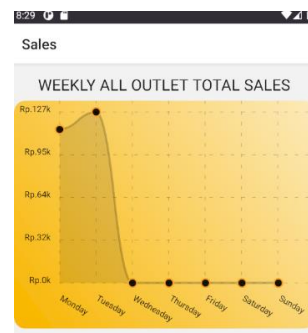
Gbr 7. Antarmuka Login



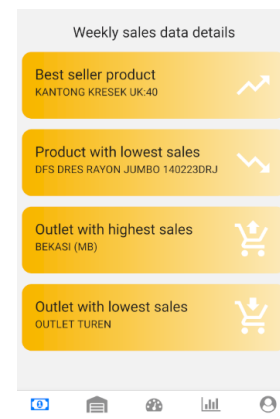
Gbr 8. Antarmuka Navigation Menu



Gbr 9. Antarmuka Dashboard



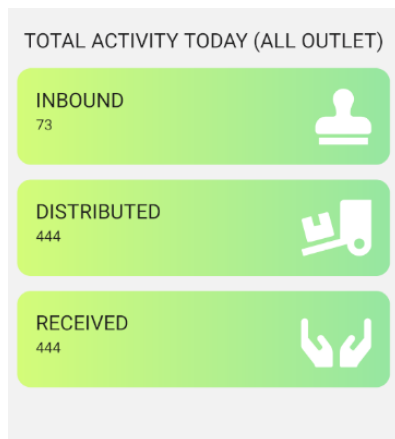
Gbr 10. Antarmuka Sales Chart



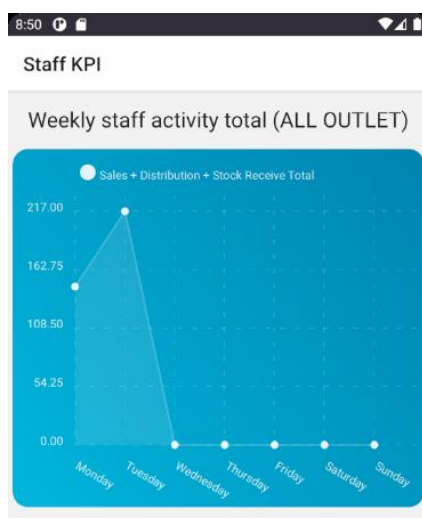
Gbr 11. Antarmuka Sales Detail



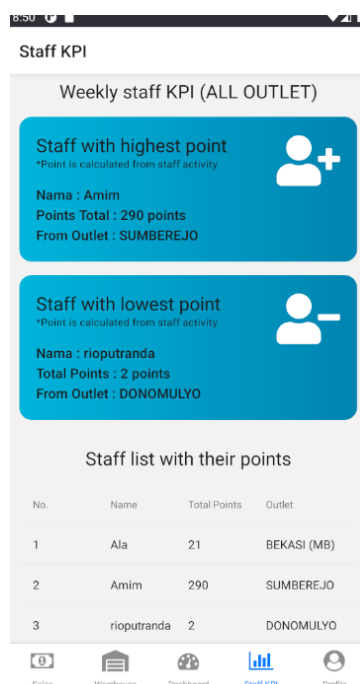
Gbr 12. Antarmuka Warehouse Chart



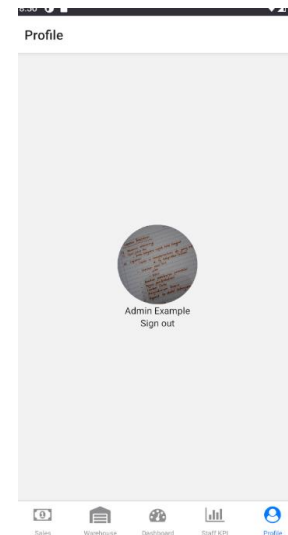
Gbr 13. Antarmuka Warehouse Detail



Gbr 14. Antarmuka Staff KPI Chart



Gbr 15. Antarmuka Staff KPI Detail



Gbr 16. Antarmuka Profile

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan mengenai perancangan aplikasi *mobile KPI Dashboard* pada PT. Mega Dagang Internasional, dapat diambil kesimpulan sebagai berikut :

1. Agar aplikasi *Key Performance Indicator (KPI) Dashboard* berbasis *Android* dapat membantu perusahaan dalam memperoleh informasi KPI secara efisien, maka dalam proses perancangan aplikasi tersebut digunakanlah metode pengembangan aplikasi *Rapid Application Development (RAD)*. Penggunaan *framework React Native* juga berperan dalam menghasilkan aplikasi yang efisien karena dapat mempercepat proses pembuatan aplikasi serta memberikan kemudahan dalam menyajikan informasi KPI kepada *user* agar lebih mudah dipahami.
2. Implementasi *backend API* pada aplikasi *KPI mobile* ini yaitu menggunakan rancangan *request* dan *response* yang diolah menggunakan *framework React Native*. Rancangan *request* dan *response* tersebut berfungsi sebagai acuan dalam menentukan data apa saja yang akan ditampilkan dalam aplikasi *KPI mobile*. *Backend API* yang diterima diolah menggunakan *framework React Native* untuk menampilkan informasi KPI ke dalam tampilan aplikasi. Dengan penggunaan *framework React Native* dapat membuat proses implementasi *backend API* menjadi lebih aman, mudah, dan cepat.

3. Penerapan Metode *Rapid Application Development* (RAD) dalam proses perancangan aplikasi *KPI Mobile* ikut melibatkan *user* dalam prosesnya sehingga hasil yang didapatkan sesuai dengan yang diharapkan *user*. Metode RAD memiliki empat tahap yaitu menentukan *requirements*, membuat desain *prototype*, pengumpulan *feedback*, dan penyelesaian aplikasi. Dengan ikut andilnya *user* dalam semua tahap tersebut maka dapat membantu proses perancangan aplikasi dengan efisien dan cepat.

5. DAFTAR PUSTAKA

- [1] Setiawan, Rony. (2021, Oct.20). *Apa Itu React Native? Apa Kelebihan dan Kekurangannya?* [Online]. Available: <https://www.dicoding.com/blog/apa-itu-react-native/>.
- [2] Talenta, Insight. (2023, Jan.29). *Key Performance Indicators KPI Sebagai Contoh Alat Ukur Kinerja* [Online]. Available: <https://www.talenta.co/blog/insight-talenta/definisi-dan-cara-alat-ukur-kinerja-karyawan-dengan-kpi/>.
- [3] A, Faradilla. (2022, Dec.07). *Apa Itu API? Pengertian, Cara Kerja, dan Manfaat API* [Online]. Available: <https://www.hostinger.co.id/tutorial/api-adalah>.
- [4] Kendall, J.E. & Kendall, K.E. "Analisis dan Perancangan Sistem", no. 5, 2010
- [5] Akbar, Muhammad. (2018). *Pengembangan RESTFUL API Untuk Application Specific High Level Location Service* [Online]. Available FTP: [dSPACE.UIN.ac.id](ftp://dSPACE.UIN.ac.id/bitstream/handle/123456789/9836/RESTFUL_API_Untuk_Application_Specific_High_Level_Location_Service.pdf) Directory: [bitstream/handle/123456789/9836](ftp://dSPACE.UIN.ac.id/bitstream/handle/123456789/9836/RESTFUL_API_Untuk_Application_Specific_High_Level_Location_Service.pdf) File: [RESTFUL_API_Untuk_Application_Specific_High_Level_Location_Service.pdf](ftp://dSPACE.UIN.ac.id/bitstream/handle/123456789/9836/RESTFUL_API_Untuk_Application_Specific_High_Level_Location_Service.pdf)
- [6] Rifandi, Ade. (2017). *Membuat Aplikasi Informasi Lokasi Friend Finder Berbasis GPS Pada Sistem Operasi Android Menggunakan Navigasi Tercepat* [Online]. Available FTP: [https://eprints.umm.ac.id/](ftp://eprints.umm.ac.id/jiptumpp-gdl-aderifandi-48944-4-babiii.pdf) Directory: [35666/4](ftp://eprints.umm.ac.id/jiptumpp-gdl-aderifandi-48944-4-babiii.pdf) File: [jiptumpp-gdl-aderifandi-48944-4-babiii.pdf](ftp://eprints.umm.ac.id/jiptumpp-gdl-aderifandi-48944-4-babiii.pdf)