

## Pengembangan Web Admin Aplikasi Pulsa Dengan *Reactjs* Menggunakan Metode *Rapid Application Development*

Heri Suroyo<sup>a</sup>, M. Dery Dermawan<sup>b</sup>, Alek Wijaya<sup>c</sup>, Jemakmun<sup>d</sup>

<sup>a</sup> Teknik Informatika, Fakultas Sains Teknologi, Universitas Bina Darma, E-mail: [herisuroyo@binadarma.ac.id](mailto:herisuroyo@binadarma.ac.id)

<sup>b</sup> Teknik Informatika, Fakultas Sains Teknologi, Universitas Bina Darma, E-mail: [deryeternalco@gmail.com](mailto:deryeternalco@gmail.com)

<sup>c</sup> Teknik Informatika, Fakultas Sains Teknologi, Universitas Bina Darma, E-mail: [alek\\_wj@binadarma.ac.id](mailto:alek_wj@binadarma.ac.id)

<sup>d</sup> Teknik Informatika, Fakultas Sains Teknologi, Universitas Bina Darma, E-mail: [jemakmun@binadarma.ac.id](mailto:jemakmun@binadarma.ac.id)

Submitted: 16-03-2024, Reviewed: 26-03-2024, Accepted 28-04-2024  
<https://doi.org/10.47233/jteksis.v6i2.1337>

### Abstract

*Credit has developed into a basic social need in the digital era, therefore an application is needed that can help in delivering this service. The credit admin application, which is a web-based application that can be accessed from anywhere, can be a solution in offering this service. The aim of this research is to develop a web-based credit admin application using ReactJS Framework and carry out API testing on the applications that have been created. A software development strategy known as Rapid Application Development was used in this research. By focusing on fast turnaround times and adaptability to change, this method can speed up the development process. Test findings carried out for this research include API testing and test cases. API test results include tests using the get, post, and put methods. In the user menu the get API method has an average time of 6.8ms and uses 294 B of memory, then in the post method it takes an average of 1630ms or 1.6 seconds and uses 294 B of memory, and in the put method it takes an average of 27.5ms and uses 298 B memory. Then, in testing the test case for the application, all tests on each feature were successful.*

**Keywords:** Web Application, Credit Admin, ReactJs, Rapid Application Development Method

### Abstrak

Pulsa telah berkembang menjadi kebutuhan sosial yang mendasar di era digital, oleh sebab itu diperlukan suatu aplikasi yang dapat membantu dalam penyampaian layanan tersebut. Aplikasi admin pulsa yang merupakan aplikasi berbasis web yang dapat diakses dari mana saja dapat menjadi solusi dalam menawarkan layanan tersebut. Tujuan dalam penelitian ini ialah pengembangan aplikasi admin pulsa berbasis web menggunakan framework ReactJS serta melakukan pengujian API terhadap aplikasi yang telah dibuat. Strategi pengembangan perangkat lunak yang dikenal sebagai Rapid Application Development digunakan dalam penelitian ini. Dengan berfokus pada waktu penyelesaian yang cepat dan kemampuan beradaptasi terhadap perubahan, metode ini dapat mempercepat proses pengembangan. Temuan pengujian yang dilakukan untuk penelitian ini meliputi pengujian API dan test case. Hasil pengujian API meliputi pengujian menggunakan metode get, post, dan put. Dalam menu pengguna metode get API memiliki rata-rata waktu 6,8ms dan memakai 294 B memori, kemudian pada metode post memakan waktu rata-rata 1630ms atau 1.6 detik dan menggunakan 294 B memori, dan pada metode put menghabiskan waktu rata-rata 27,5ms dan memakai 298 B memori. Kemudian dalam pengujian test case terhadap aplikasi semua pengujian pada tiap fitur berhasil.

**Kata kunci:** Aplikasi Web, Admin Pulsa, ReactJs, Metode Rapid Application Development

*This work is licensed under Creative Commons Attribution License 4.0 CC-BY International license*



### PENDAHULUAN

Kemajuan pesat teknologi informasi dan komunikasi telah mengubah cara orang berkomunikasi dan menjalankan bisnis sehari-hari. Sektor telekomunikasi merupakan salah satu sektor yang terkena dampaknya, terutama dalam hal pengisian pulsa atau top-up pulsa. Banyaknya jenis teknologi yang ada saat ini membuat hidup menjadi lebih mudah dan bermanfaat bagi masyarakat seiring dengan kemajuan teknologi. Salah satu jenis kemajuan teknologi adalah aplikasi. [1]. Dengan berkembangnya teknologi dalam bidang informasi dan komunikasi membuat manusia terus berinovasi dalam memenuhi kebutuhannya. [2] Pemanfaatan teknologi akan menghubungkan suatu pembicaraan seperti halnya ponsel yang merupakan gambaran keterkaitan antara komunikasi dan teknologi. [3].

Dengan adanya perkembangan ini, komunikasi memiliki peranan penting dalam kehidupan sebagai sarana informasi. [4]

Di era teknologi informasi, bisnis atau perdagangan, komunikasi, dan jaringan kini beralih dari metode konvensional ke metode digital. [5] Salah satu usaha perdagangan yang kini berkembang pesat di seluruh wilayah Indonesia adalah penjualan pulsa, baik isi ulang fisik maupun elektrik. Negara Indonesia mempunyai kebiasaan hidup konsumtif dalam menggunakan pulsa isi ulang. [3]. Akan sulit untuk bersaing di dunia komersial tanpa teknologi informasi, itulah sebabnya penggunaannya semakin meningkat. Hal ini harus diperhitungkan agar bisnis tetap kompetitif, efektif, dan efisien, yang pada gilirannya akan merangsang pertumbuhan penjualan dan

memberikan layanan terbaik kepada klien melalui penerapan dan pemanfaatan teknologi informasi.[6]. Namun demikian, masih terdapat beberapa permasalahan yang perlu diselesaikan terkait dengan penggunaan aplikasi pulsa. Keterbatasan fungsi, efektivitas dan user interface pada sistem administrasi aplikasi pulsa menjadi salah satu permasalahan yang sering dihadapi. Dalam hal ini, penggunaan teknologi web dengan pendekatan pengembangan aplikasi yang cepat (Rapid Application Development) dapat menjadi solusi yang efektif.

ReactJS, sebagai salah satu framework JavaScript yang populer, memberikan kemudahan dalam membangun antarmuka pengguna yang responsif dan interaktif. Library Javascript yang disebut React JS berguna untuk membuat antarmuka pengguna. Karena didasarkan pada komponen yang interaktif, stateful, dan dapat digunakan kembali, React JS memudahkan pengembang untuk membuat aplikasi web.[1]. Dengan menggabungkan teknologi ReactJS dan pendekatan RAD, dapat diharapkan bahwa pengembangan web admin aplikasi pulsa dapat dilakukan dengan lebih efisien dan hasil yang berkualitas. Library atau framewrok JavaScript mana yang kemudian digunakan dalam pengembangan situs web merupakan topik perdebatan di antara banyak CTO (Chief Technology Officers) pada perusahaan dan para pengembang software lainnya. Dua library dan framewrok yang paling banyak digunakan adalah Angular JS dan React JS. Masing-masing mempunyai kelebihan dan kekurangannya masing-masing. Namun, React JS memiliki keunggulan untuk pengembangan startup dibandingkan Angular JS. Hal ini disebabkan karena, React JS adalah library, sedangkan Angular JS adalah framework. Penggunaan library memungkinkan fleksibilitas yang lebih besar dibandingkan menggunakan framework. Disamping itu, React JS merupakan berbasis komponen dan dapat dipakai kembali jika diperlukan, yang membantu mempercepat pengembangan startup dan memberikan peluang bagi bisnis untuk mendominasi pasar mereka.[7]

Beberapa penelitian yang telah membahas mengenai pengembangan aplikasi manajemen pulsa berbasis web diantaranya penelitian yang telah dilakukan oleh Saputra dkk, dalam jurnal berjudul *Pembuatan Aplikasi Manajemen Konter Pulsa Dalam Meningkatkan Efektivitas Penjualan Dengan Metode Agile Model Studi Kasus Yuda Cell*. Didalam penelitian ini peneliti melakukan pengembangan aplikasi menggunakan metode Agile model serta menggunakan php yang menjadi bahasa pemograman. Temuan dari penelitian ini adalah sebuah aplikasi manajemen pulsa bernama Yuda Cell[8]. Selanjutnya penelitian yang dilakukan oleh

Ferdiansyah dkk dengan penelitian yang berjudul *Sistem Perancangan Penjualan Pulsa Pada Counter AF Cell Depok*. Pada penelitian ini peneliti menggunakan metode waterfall dalam perancangannya, dan menggunakan bahasa pemograman java dalam pengkodeannya. Berdasarkan temuan penelitian, sistem aplikasi ini dirancang bersifat sementara, sehingga penggunaannya dibatasi hanya pada mereka yang memiliki izin akses dan memanfaatkan kinerja Counter AF Cell.[9]. Penelitian selanjutnya yaitu, penelitian yang merancang sebuah sistem informasi atau aplikasi penjualan berbasis web pada MF-Store, dalam penelitian ini peneliti memakai bahasa pemograman php serta menggunakan metode waterfall. Kemudian terdapat penelitian yang telah dilaksanakan oleh Rohim dan Virgan yang melakukan perancangan sistem informasi penjualan pulsa elektronik berbasis web[10]. Dan penelitian terakhir yang berjudul *Sistem Informasi Penjualan Paket Berbasis WEB(Studi Kasus : ChariNet)*, dalam penelitian ini peneliti menggunakan metode waterfall dalam pengembangannya, serta menggunakan Adobe Dreamweaver CS4 dalam perancangan webnya, dan menggunakan bahasa pemograman php dalam pengkodeannya[11].

Penelitian ini bertujuan untuk mengembangkan web admin aplikasi pulsa menggunakan library JavaScript ReactJS dengan menerapkan metode Rapid Application Development yang belum diterapkan sebelumnya, serta pada aplikasi yang dikembangkan akan dilakukan pengujian terhadap API aplikasi. Dalam penelitian ini, akan dilakukan analisis kebutuhan, perancangan antarmuka pengguna, serta pengujian dari aplikasi pulsa.

## METODE PENELITIAN

Metode penelitian yang digunakan adalah Pendekatan Rapid Application Development (RAD). RAD adalah metode siklus hidup yang bertujuan untuk memberikan hasil yang jauh lebih cepat dan berkualitas lebih tinggi dibandingkan dengan yang diperoleh melalui siklus konvensional.[12] RAD juga menghasilkan suatu sistem yang dapat memenuhi kebutuhan dari para pemakai sehingga dapat mempersingkat waktu untuk pengembangan ulang setelah tahapan implementasi.[13] Rangkaian langkah-langkah pengembangan sebuah aplikasi dalam metode RAD adalah perencanaan kebutuhan, desain atau perencanaan sistem, proses pengembangan dan feedback dan Implementasi.

### 2.1. Perencanaan Kebutuhan

Pada tahap awal ini pengumpulan data dilakukan melalui tinjauan pustaka atau studi literatur. serta mengidentifikasi masalah yang didapat dari

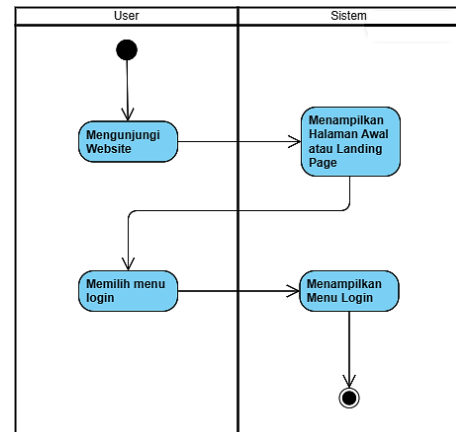
pengguna yang bertujuan untuk mengetahui tujuan serta informasi yang dibutuhkan dalam pengembangan aplikasi. Untuk mendukung dan memperkuat teori-teori yang sudah ada sebelumnya dan memberikan informasi yang cukup untuk menyelesaikan penelitian ini, studi literatur dilakukan dengan membaca dan memahami buku-buku referensi, tesis, jurnal, dan media lain yang terhubung dengan pengembangan web admin pulsa menggunakan ReactJs dan pendekatan RAD.

## 2.2. Desain Sistem

Pada tahap ini melakukan penyelesaian proses desain dan melakukan perbaikan jika masih terdapat perbedaan desain antara pengguna dan analis[14]. Pada tahap ini, tujuannya adalah untuk menyelesaikan proses desain dan membuat perubahan jika pengguna dan analis tidak setuju dengan desain tersebut. Dikarenakan pada tahap ini pengguna dapat langsung merespons jika ada masalah dengan desain, tingkat keterlibatan mereka sangat penting bagi keberhasilan proyek. Setelah mengetahui kebutuhan dalam pengembangan aplikasi web admin pulsa kemudian peneliti membuat desain *activity diagram*. Diagram aktivitas dalam perangkat lunak merepresentasikan alur kerja sistem, aktivitas, proses bisnis, atau menu pada sebuah aplikasi.[15] Diagram Activity juga digunakan untuk mengelompokkan alur tampilan terhadap sistem tersebut.[16]

### a) Diagram aktivitas *Login*

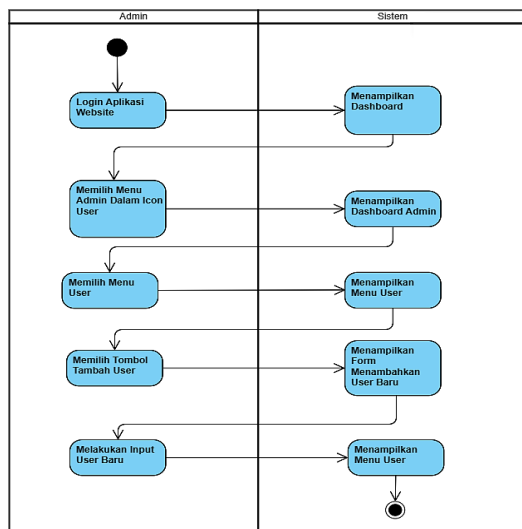
Didalam Diagram aktivitas *login* ini dijelaskan hal pertama yang harus dilakukan oleh pengguna adalah mengunjungi aplikasi website jika pengguna ingin menggunakan aplikasi ini, setelah itu sistem akan menampilkan halaman pertama yang akan tampil yaitu *landing page*, selanjutnya pengguna bisa memilih tombol login yang ada pada halaman tersebut sehingga sistem dapat mengarahkan pengguna ke halaman login aplikasi website. Didalam halaman login pengguna akan memasukkan email dan password setelah sistem memverifikasi login yang dilakukan, selanjutnya sistem akan mengarahkan pengguna ke menu dashboard.



Gambar 1 Diagram aktivitas Login

### b) Diagram aktivitas Memasukan Pengguna Baru

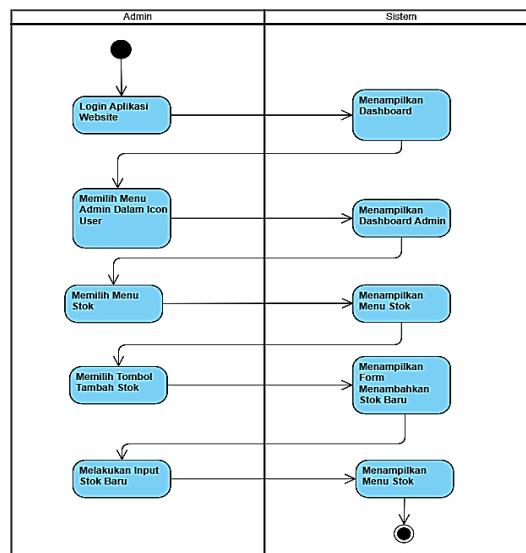
Pada Diagram aktivitas ini pertama admin melakukan *login* pada aplikasi website, kemudian admin akan diarahkan ke menu dashboard pengguna kemudian admin memilih menu admin yang terdapat pada *icon* pengguna, setelah memilih *icon* pengguna sistem akan memunculkan menu *icon* yang dimana didalamnya terdapat menu admin, setelah memilih menu admin sistem akan mengarahkan admin ke dashboard admin, menu ini hanya dapat diakses oleh admin saja. Setelah didalam mode admin selanjutnya admin memilih menu user, kemudian admin memilih tombol tambah pengguna, sistem akan menunjukkan form yang harus diisi oleh admin dengan data profil pengguna baru yang akan mendaftar mulai dari nama pengguna, email serta password pengguna. Fitur menambahkan pengguna baru hanya dapat diakses oleh admin saja.



Gambar 2 Diagram aktivitas Memasukan Pengguna Baru

c) Diagram aktivitas Memasukan Stok Baru

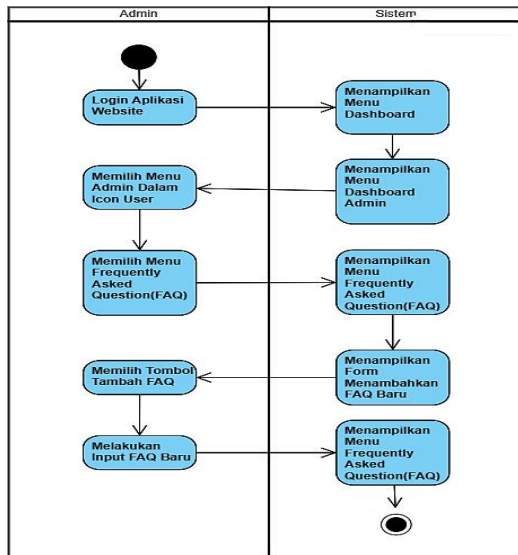
Pada Diagram aktivitas ini pertama-tama admin melakukan *login* pada aplikasi website, kemudian admin akan diarahkan ke menu dashboard pengguna setelah itu admin memilih menu admin pada *icon* pengguna, kemudian sistem akan menempatkan admin ke halaman dashboard admin selanjutnya admin memilih menu stok, didalam menu stok admin dapat memilih stok pulsa atau paket data, kemudian admin memilih tombol tambah stok, sesuai dengan menu pilihan sebelumnya sistem akan menampilkan form stok mana yang dipilih oleh admin selanjutnya admin mengisi form penambahan stok baru tersebut. Fitur penambahan stok pulsa maupun paket data hanya dapat dilakukan oleh admin saja, sehingga user biasa tidak akan bisa merubah atau mengelola stok yang tersedia didalam aplikasi.



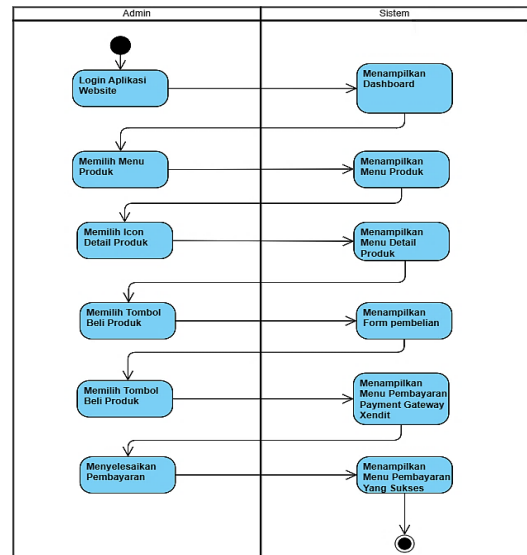
Gambar 3 Diagram aktivitas Memasukan Stok Baru

d) Diagram aktivitas Memasukan *Frequently Asked Question*(FAQ)

Pada Diagram aktivitas FAQ pertama-tama admin melakukan *login* pada aplikasi website, kemudian admin akan diarahkan ke menu dashboard pengguna setelah itu admin memilih menu admin pada *icon* pengguna, kemudian sistem akan menempatkan admin ke halaman dashboard admin, selanjutnya admin memilih menu kelola *Frequently Asked Question*(FAQ), didalam menu FAQ kemudian admin menekan tombol tambah item, sistem selanjutnya akan menampilkan form untuk menambah item FAQ, selanjutnya admin mengisi form penambahan item baru tersebut. Fitur penambahan *Frequently Asked Question*(FAQ) hanya dapat dilakukan oleh admin saja.



Gambar 4 Diagram aktifitas *Input FAQ*



Gambar 5 Diagram aktivitas Transaksi

#### e) Diagram aktivitas Transaksi

Pada Diagram aktivitas transaksi pertama-tama pengguna melakukan login pada aplikasi website, kemudian pengguna akan diarahkan ke menu dashboard setelah itu pengguna memilih menu stok, didalam menu stok pengguna memilih *icon* detail stok produk, didalam menu detail produk pengguna dapat melakukan transaksi sesuai dengan produk yang dipilih dengan memilih tombol beli produk, setelah itu sistem akan menampilkan form pembelian produk, setelah mengisi dan memilih tombol beli produk pengguna akan diarahkan ke menu pembayaran, setelah melakukan pembayaran sistem akan menampilkan status orderan yang telah sukses.

#### 2.3 Proses pengembangan dan feedback

Dalam tahap ini rancangan sistem aplikasi yang telah didesain dan telah mendapat persetujuan akan dikembangkan menjadi aplikasi dalam versi beta hingga tahap akhir yaitu versi final. Dalam tahap ini, pemrogram harus melaksanakan tugas pengembangan, berintegrasi dengan komponen lain, dan terus mempertimbangkan umpan balik dari pengguna atau pihak klien. Apabila semua tahapan sesuai rencana, prosesnya dapat dilanjutkan ke tahapan selanjutnya, sementara itu jika aplikasi yang dikembangkan belum dapat memenuhi kebutuhan, programmer diharuskan untuk kembali ke tahapan sebelumnya. [17]

#### 2.4 Implementasi

Pada tahap ini desain sistem aplikasi yang telah dibuat serta dilakukan proses feedback pada tahap sebelumnya akan dilakukan pengujian. Pengujian pertama ialah melakukan pengujian terhadap API dengan menggunakan aplikasi postman, kemudian aplikasi juga akan dilakukan pengujian dengan *test case*. Pengujian bertujuan agar aplikasi telah siap untuk diberikan kepada para pengguna.

#### HASIL DAN PEMBAHASAN

Pada saat pengembangan aplikasi penulis mendapati beberapa masalah dalam melakukan pengambilan data dari API(Application Programming Interface), saat melakukan fetching API dari *backend* dibutuhkan sebuah token yang berfungsi sebagai *authorization* yang akan mencegah pengguna yang tidak memiliki hak akses untuk melakukan pengambilan data dari API, sebagai contoh user tidak dapat melakukan penambahan produk sedangkan admin dapat menambahkan produk, maka dari itu dibuatlah



sebuah fungsi berupa Axios Interceptors. Axios Interceptors memberi Anda kemampuan untuk mencegah permintaan atau tanggapan sebelum mereka dikirim atau diterima. Fungsi ini berguna untuk menambahkan header, mencatat data, dan mengubah data permintaan atau respons.[18] Dengan menambahkan header berupa bearer token pada saat pengambilan data dari API, aplikasi dapat memilah pengguna mana yang mengakses API dengan mengecek setiap request yang telah memiliki bearer token jika token tersebut merupakan token dari pengguna maka akan diberikan hak akses sebagai pengguna, jika merupakan token admin maka akan diberikan full akses dalam aplikasi.

Kemudian permasalahan muncul kembali saat user telah memilih produk dan akan melakukan pembayaran dibutuhkan metode pembayaran yang fleksibel dan aman, penulis menggunakan metode pembayaran berupa payment gateway xendit, Xendit adalah payment gateway yang mempermudah pembayaran bagi berbagai jenis bisnis di Indonesia dan Filipina.[19] Metode pembayaran ini lebih fleksibel dikarenakan xendit menyediakan banyak opsi pembayaran mulai dari e-wallet sampai pembayaran lewat bank dan transaksi yang dilakukan lebih aman dikarenakan data transaksi telah terintegrasi dengan payment gateway, dibandingkan dengan beberapa penelitian yang dijadikan rujukan seperti pada penelitian yg dilakukan oleh A. Rohim and M. Virgan berjudul “Perancangan sistem informasi penjualan pulsa elektronik berbasis web” pada penelitian ini peneliti belum mengimplementasikan payment gateway dan hanya menampilkan hasil transaksi berupa data log pembayaran. Kemudian pada penelitian berjudul “Perancangan Sistem Informasi Penjualan MF-STORE”, menu pembayaran belum menggunakan payment gateway dan berupa form inputan produk yang dibeli.

Pada bagian ini peneliti akan membagi menjadi 2 bagian, hasil berupa hasil implementasi desain sistem yang telah dibuat dan pada bagian pembahasan akan membahas pengujian yang dilakukan terhadap aplikasi. Aplikasi yang dikembangkan bernama Hola Pulsa aplikasi ini merupakan aplikasi yang berfungsi untuk manajemen penjualan pulsa dan paket data, dengan menggunakan payment gateway aplikasi memudahkan penggunanya untuk bertansaksi dengan berbagai macam opsi pembayaran yang telah disediakan.

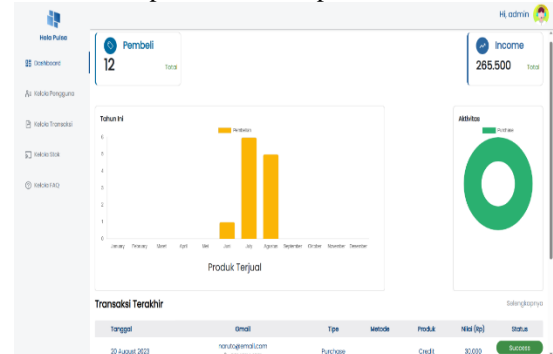
### 3.1 Hasil Desain Sistem

Pada bagian hasil desain sistem ini peneliti akan membahas 3 tampilan halaman pada aplikasi yaitu dashboard admin, dashboard user dan menu produk didalam menu produk akan dibagi dalam 3 bagian

yaitu menu produk, form pembelian produk serta menu pembayaran.

#### A. Dashboard Admin

Berikut merupakan hasil tampilan dashboard admin

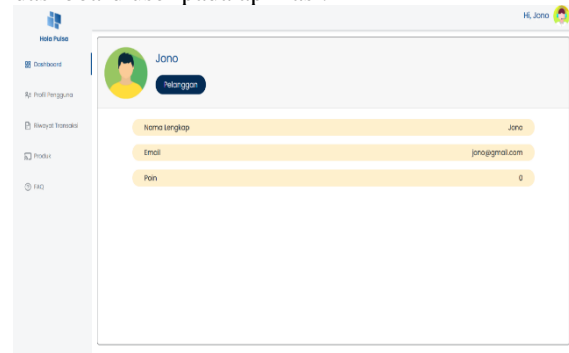


Gambar 6 Dashboard Admin

Dari gambar Menu dashboard admin hanya dapat diakses oleh admin, menu ini memiliki informasi terkini didalam aplikasi, disebelah kiri terdapat *sidebar* yang berfungsi sebagai menu navigasi tiap menu pada aplikasi web. Kemudian *navbar* pada bagian atas yang berisikan nama dan *icon* pengguna yang jika dipilih akan menampilkan menu *icon*, menu *icon* berisikan menu mode admin, menu *user page* serta *logout*. Terdapat card dibagian atas yang berisikan jumlah penjualan dan total pemasukan, kemudian terdapat grafik *bar* yang menunjukkan penjualan pada tiap bulan yang terjadi pada tahun ini, disebelahnya terdapat *diagram* donat yang berisikan informasi tipe pembelian yang dilakukan oleh pengguna. Dibagian bawah terdapat tabel yang menunjukkan pembelian terbaru yang dilakukan oleh pengguna.

#### B. Dashboard User

Berikut merupakan hasil dari tampilan halaman dashboard user pada aplikasi.

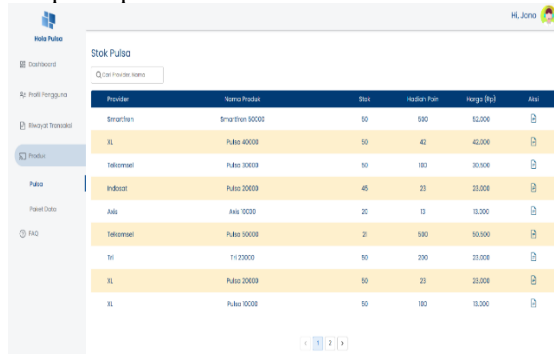


Gambar 7 Dashboard User

Halaman ini merupakan halaman yang akan tampil pertama kali saat seorang user ataupun admin memasuki aplikasi. Menu ini berisikan informasi mengenai pengguna mulai dari nama lengkap, alamat email dan poin yang dimiliki oleh pengguna.

#### C. Menu Produk

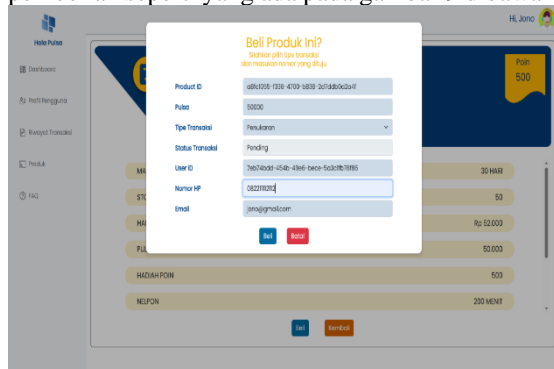
Berikut merupakan tampilan dari menu produk yang ada pada aplikasi



Produk	Nama Produk	Stok	Notifikasi	Harga (Rp)	Aksi
Smartfren	Smartfren 50000	50	500	52.000	
XL	Pulsa 40000	50	40	42.000	
Telkomsel	Pulsa 50000	50	100	50.000	
Indosat	Pulsa 20000	45	20	23.000	
Axiata	Axiata 10000	20	10	13.000	
Telkomsel	Pulsa 10000	20	500	10.000	
Tri	Tri 20000	50	200	21.000	
XL	Pulsa 20000	50	20	23.000	
XL	Pulsa 10000	50	100	13.000	

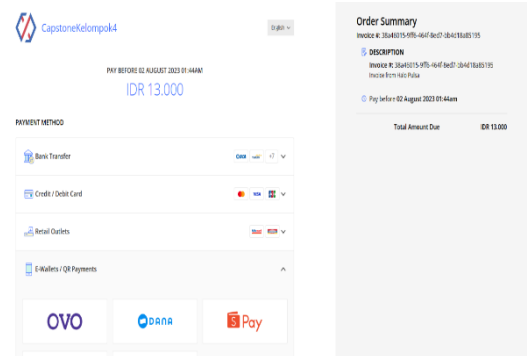
Gambar 8 Menu Produk

Didalam menu ini terdapat informasi mengenai produk apa saja yang tersedia pada aplikasi, pengguna dapat melakukan pembelian sebuah produk dengan memilih *icon file* pada kolom aksi, setelah memilih *icon file* tersebut maka pengguna akan diarahkan ke menu detail produk, didalam menu detail produk akan ditampilkan info detail mengenai produk serta didalamnya terdapat tombol beli yang bila dipilih maka akan menampilkan form pembelian seperti yang ada pada gambar 9 dibawah.



Gambar 9 Form Pembelian Pada Menu Detail Produk

Pada form pembelian yang ada pada gambar 9 diatas pengguna hanya perlu memasukan tipe transaksi dan nomor hp yang dituju, setelah itu pengguna memilih tombol beli yang ada dibawah selanjutnya sistem akan mengarahkan pengguna ke menu pembayaran. Didalam menu pembayaran akan memeperlihatkan opsi pembayaran seperti e-wallet, transfer bank, kartu kredit dll.



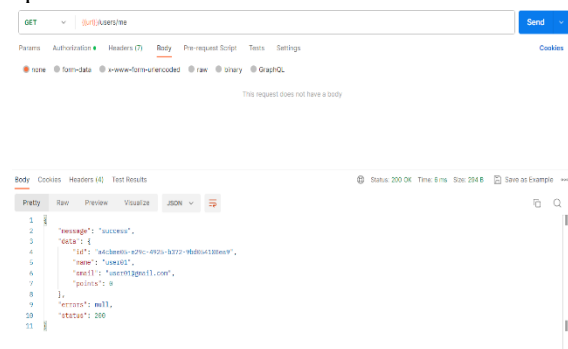
Gambar 10 Menu Pembayaran Payment Gateway Xendit

### 3.2 Pembahasan

Pembahasan akan membahas pengujian yang dilakukan terhadap aplikasi pengujian tersebut ialah pengujian API dengan aplikasi Postman. Rest API (Application Programming Interface) telah menjadi fondasi utama sebagai sarana komunikasi antara berbagai aplikasi.[20] Untuk menguji REST API, Postman adalah program yang bertindak sebagai klien REST. Pengembang API menggunakan Postman sebagai alat untuk menguji API yang baru dibuat. [21]. Pengujian akan dilakukan pada menu pengguna yang ada pada aplikasi menggunakan metode get, post dan put. Kemudian peneliti juga melakukan pengujian test case terhadap aplikasi. Test Case adalah sekelompok skenario yang dibuat oleh QA (Quality Assurance) sehingga sistem yang diuji dapat memenuhi persyaratan tertentu, memenuhi standar tertentu, dan berfungsi sebagaimana mestinya. [22]. Pengujian test case dilakukan dengan aplikasi katalon studio.

#### A. Metode get

Metode get berfungsi untuk mengambil informasi, pada pengujian metode get peneliti melakukan pengambilan informasi profil dari 10 pengguna yang ada di aplikasi, gambar 10 dibawah merupakan hasil dari pengujian metode get pada aplikasi Postman.



Gambar 11 Hasil Pengambilan Profil Pengguna dengan Metode Get Menggunakan Postman

Pada gambar diatas merupakan salah satu hasil pengujian API dengan metode get yang sukses dan berhasil menampilkan informasi profil pengguna

yang ada di menu pengguna. Didapatkan hasil dari pengujian tersebut memakan waktu 6ms dan memakai 294 B memori. Berikut hasil pengujian metode get pada 10 pengguna.

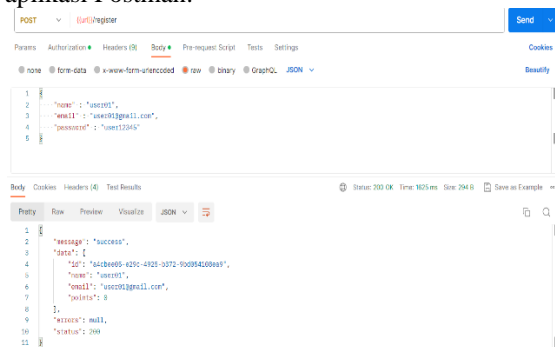
Tabel 1 Hasil Pengambilan Profil Pengguna dengan Metode Get Menggunakan Postman

User	Waktu(ms)	Memori	Status
User01	6ms	294 B	Success
User02	6ms	294 B	Success
User03	9ms	294 B	Success
User04	7ms	294 B	Success
User05	5ms	294 B	Success
User06	5ms	294 B	Success
User07	6ms	294 B	Success
User08	7ms	294 B	Success
User09	5ms	294 B	Success
User10	12ms	294 B	Success
Rata-rata	6,8ms	294 B	

Dari tabel diatas didapatkan hasil rata-rata dalam pengujian metode get untuk mengambil informasi profil pengguna pada menu pengguna memakan waktu 6,8ms dan menggunakan 294 B memori.

#### B. Metode post

Metode post berfungsi untuk mengirim informasi, pada pengujian API dengan metode post peneliti melakukan penambahan pengguna baru pada menu pengguna, gambar 11 dibawah merupakan hasil dari pengujian metode post pada aplikasi Postman.



Gambar 12 Hasil Penambahan Pengguna dengan Metode Post Menggunakan Postman

Gambar diatas merupakan salah satu hasil pengujian API dengan metode post yang sukses dan berhasil menambah pengguna yang ada menu pengguna. Didapatkan hasil dari pengujian tersebut memakan waktu 1625ms dan memakai 294 B memori. Berikut hasil pengujian metode post menambahkan 10 pengguna baru pada menu pengguna.

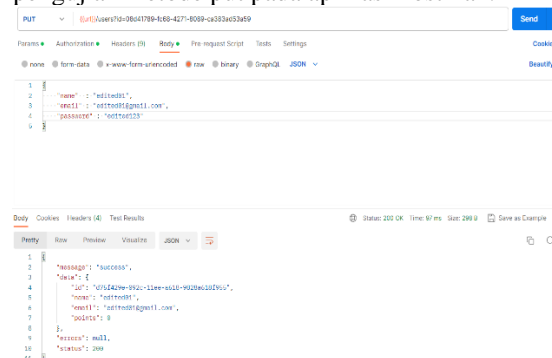
Tabel 2 Hasil Penambahan Pengguna dengan Metode Post Menggunakan Postman

User	Waktu(ms)	Memori	Status
User01	1625ms	294 B	Success
User02	1618ms	294 B	Success
User03	1630ms	294 B	Success
User04	1628ms	294 B	Success
User05	1601ms	294 B	Success
User06	1602ms	294 B	Success
User07	1616ms	294 B	Success
User08	1622ms	294 B	Success
User09	1620ms	294 B	Success
User10	1742ms	294 B	Success
Rata-rata	1630ms	294 B	

Dari tabel diatas didapatkan hasil rata-rata dalam pengujian metode post untuk menambah pengguna baru pada menu pengguna memakan waktu 1630ms atau 1.6 detik dan menggunakan 294 B memori.

#### C. Metode put

Metode put berfungsi untuk mengupdate informasi, pada pengujian API dengan metode put peneliti melakukan update informasi profil pada 10 pengguna, gambar 12 dibawah merupakan hasil dari pengujian metode put pada aplikasi Postman.



Gambar 13 Hasil Pengupdetan Informasi Pengguna dengan Metode Put Menggunakan Postman

Gambar diatas merupakan salah satu hasil pengujian API dengan metode put yang sukses dan berhasil mengupdate informasi mengenai pengguna yang ada pada aplikasi. Didapatkan hasil dari pengujian tersebut memakan waktu 97ms dan memakai 298 B memori. Berikut hasil pengujian metode put kepada 10 pengguna yang terdapat didalam menu pengguna.



Tabel 3 Hasil Pengupdetan Informasi Pengguna dengan Metode Put Menggunakan Postman

User	Waktu(ms)	Memori	Status
User01	97ms	298 B	Success
User02	18ms	298 B	Success
User03	21ms	298 B	Success
User04	7ms	298 B	Success
User05	22ms	298 B	Success
User06	21ms	298 B	Success
User07	46ms	298 B	Success
User08	17ms	298 B	Success
User09	11ms	298 B	Success
User10	15ms	298 B	Success
Rata-rata	27,5ms	298 B	

Dari tabel diatas didapatkan hasil rata-rata dalam pengujian metode put untuk mengupdate informasi pengguna pada menu pengguna memakan waktu 27,5ms dan menggunakan 298 B memori.

#### D. Pengujian Test Case Pada Aplikasi

Pada pengujian ini peneliti akan memperlihatkan hasil dari pengujian dengan skenario penambahan pengguna baru pada aplikasi yang dilakukan oleh admin, hasil pengujian dapat dilihat pada tabel dibawah.

Tabel 4 Hasil Pengujian Test Case

Deskripsi	Waktu	Status
Membuka Browser	1.506s	Passed
Mengklik tombol Login	0.567s	Passed
Menekan field email	0.494s	Passed
Memasukan email	0.684s	Passed
Menekan field password	0.331s	Passed
Memasukan password	0.723s	Passed
Mengklik tombol Masuk	0.327s	Passed
Login sukses	3.797s	Passed
Menekan ok pada notifikasi login sukses	0.333s	Passed
Masuk menu pengguna	0.676s	Passed
Masuk menu pelanggan	0.345s	Passed
Menekan tombol tambah pengguna	0.989s	Passed

Menekan field nama lengkap	0.452s	Passed
Memasukan nama lengkap	0.750s	Passed
Menekan field email	0.328s	Passed
Memasukan email	0.622s	Passed
Menekan field password	0.435s	Passed
Memasukan password	0.741s	Passed
Menekan field konfirmasi password	0.336s	Passed
Memasukan konfirmasi password	0.741s	Passed
Menekan tombol tambah pengguna	0.457s	Passed
Total waktu	15.828s	

Dari tabel diatas dijelaskan bahwa pertama admin mengakses web aplikasi, kemudian masuk kehalaman login dengan memilih menu login, setelah itu admin menginput email dan password pada form login, kemudian menekan tombol masuk untuk masuk ke menu dashboard, setelah didalam menu dashboard admin kemudian masuk ke menu pengguna dan menekan tombol tambah pengguna, selanjutnya admin memasukkan data pengguna baru dan menekan tombol tambah pengguna. Didapati hasil pengujian keseluruhan berhasil dan memakan total waktu 15.828s atau 16 detik.

## SIMPULAN

Setelah melakukan pengujian pada aplikasi peneliti menarik kesimpulan bahwa, Hasil dari pengujian test case pada aplikasi memberikan status passed yang berarti aplikasi telah berhasil dijalankan. Aplikasi yang dikembangkan dapat berjalan dengan lancar dengan hasil dari pengujian API menggunakan aplikasi Postman pada semua menu yang ada di aplikasi yang berstatus success.

## UCAPAN TERIMA KASIH

Saya mengucapkan banyak terima kasih kepada para dosen pembimbing saya, dan juga teman-teman yang telah meluangkan waktu untuk dapat membimbing saya dalam menyelesaikan pembuatan jurnal ini.

## DAFTAR PUSTAKA

- [1] A. A. Kroons and C. Dewi, "Pengembangan Dashboard Trivy Berbasis Website Menggunakan React Js dan Golang," *J. Indones. Manaj. Inform. Dan Komun.*, vol. 4, no. 3, pp. 1037–1049, 2023.
- [2] R. Choirudin and A. Adil, "Implementasi Rest Api Web Service dalam Membangun Aplikasi Multiplatform untuk Usaha Jasa," *MATRIK J. Manaj. Tek. Inform. Dan Rekayasa Komput.*, vol. 18, no. 2, pp. 284–293, 2019.

- [3] H. Intan, "PERANCANGAN SISTEM PENJUALAN PULSA ELEKTRIK MENGGUNAKAN BAHASA PEMROGRAMAN C++ (BORLAND C++) DENGAN LOW LEVEL CODING," *Eprint Pancabudi*, 2021.
- [4] O. Fenardi and F. S. Lee, "Aplikasi Akademik Berbasis Website Menggunakan Metode Extreme Programming Pada SMAN1 Belinyu," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 4, pp. 440–447, 2023.
- [5] K. Khalid, I. S. Rozas, and D. Rolliawati, "Trends and Patterns of The Internet Use During School Holidays," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 6, no. 2, pp. 89–98, 2020.
- [6] M. Y. Putra and R. W. R. Lolly, "Sistem Aplikasi Penjualan Souvenir Berbasis Web Menggunakan Metode Rapid Application Development (RAD)," *Inf. Syst. Educ. Prof. J. Inf. Syst.*, vol. 5, no. 2, p. 151, Jun. 2021, doi: 10.51211/isbi.v5i2.1548.
- [7] L. Iswari and others, "Penerapan React JS Pada Pengembangan FrontEnd Aplikasi Startup Ubaform," *AUTOMATA*, vol. 2, no. 2, 2021.
- [8] A. F. Saputra, F. A. Panuntun, R. Ariansyah, and R. Djutalov, "PEMBUATAN APLIKASI MANAJEMEN KONTER PULSA DALAM MENINGKATKAN EFEKTIVITAS PENJUALAN DENGAN METODE AGILE MODEL Studi kasus YUDA CELL," *J. Res. Publ. Innov.*, vol. 1, no. 2, pp. 407–413, 2023.
- [9] V. Ferdiansyah, T. Afrizal, and N. Dwitianti, "Sistem Perancangan Penjualan Pulsa pada Counter AF Cell," *J. Ris. Dan Apl. Mhs. Inform. JRAMI*, vol. 3, no. 02, pp. 276–283, 2022.
- [10] A. Rohim and M. Virgan, "Sistem Informasi Penjualan Pulsa Prabayar Berbasis Web," *J. Ilm. MIKA AMIK Al Muslim*, vol. 5, no. 2, pp. 47–52, 2021.
- [11] M. Amin, K. Ihwan, and others, "SISTEM INFORMASI PENJUALAN PAKET BERBASIS WEB (Studi Kasus: ChariNET)," *JUTI UNISI*, vol. 6, no. 2, pp. 24–29, 2022.
- [12] J. Parhusip, A. Saputra, C. Handika, and I. P. Parhusip, "PENGEMBANGAN APLIKASI BIMBINGAN KONSELING PADA SMK NEGERI 1 MUARA TEWEH MENGGUNAKAN METODE RAPID APPLICATION DEVELOPMENT (RAD)," *J. Teknol. Inf. J. Keilmuan Dan Apl. Bid. Tek. Inform.*, vol. 16, no. 1, pp. 61–72, Jan. 2022, doi: 10.47111/jti.v16i1.3668.
- [13] S. Aswati, M. S. Ramadhan, A. U. Firmansyah, and K. Anwar, "Studi analisis model rapid application development dalam pengembangan sistem informasi," *MATRIK J. Manaj. Tek. Inform. Dan Rekayasa Komput.*, vol. 16, no. 2, pp. 20–27, 2017.
- [14] D. Hariyanto, R. Sastra, and F. E. P. E. P. Putri, "Implementasi Metode Rapid Application Development Pada Sistem Informasi Perpustakaan," *Jupit. J. Penelit. Ilmu Dan Tek. Komput.*, vol. 13, no. 1, pp. 110–117, 2021.
- [15] R. Aditya, V. H. Pranatawijaya, and P. B. A. A. Putra, "Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype," *J. Inf. Technol. Comput. Sci.*, vol. 1, no. 1, pp. 47–57, 2021.
- [16] K. Wakhidah, B. Budiman, and W. Winarti, "Rancangan Sistem Informasi Perpustakaan Berbasis Website Menggunakan Barcode Di Sekolah MA Raden Rahmat," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 61–68, 2023.
- [17] C. Rizal, M. Iqbal, R. R. Putra, and M. Israr, "RANCANG BANGUN SISTEM INFORMASI POSYANDU IBU DAN ANAK BERBASIS WEB," vol. 1, 2023.
- [18] N. Yadav, A. Kaur, and R. Kanji, "Comparative Analysis of API Integration using Code Vs No-code, low-code Platforms," *Jaypee Univ. Inf. Technol. Solan HP*, 2023.
- [19] M. N. J. Rozaq and U. Y. Oktiawati, "Implementasi Sistem Otomasi Order Mentoring Pada Marketplace Mentoring Platform Nusademy," *J. Appl. Comput. Sci. Technol.*, vol. 3, no. 2, pp. 199–207, 2022.
- [20] L. D. Andrianto and D. F. Suyatno, "Analisis Performa Load Testing Antara Mysql Dan Nosql Mongoddb Pada RestAPI Nodejs Menggunakan Postman," *J. Emerg. Inf. Syst. Bus. Intell. JEISBI*, vol. 5, no. 1, pp. 18–26, 2024.
- [21] M. K. Anwar and others, "Perancangan Database IoT Berbasis Cloud dengan Restful API," *Techno Com*, vol. 20, no. 2, pp. 268–279, 2021.
- [22] I. R. Dhaifullah, A. A. Salsabila, M. A. Yaqin, and others, "Survei Teknik Pengujian Software," *J. Autom. Comput. Inf. Syst.*, vol. 2, no. 1, pp. 31–38, 2022.