

Implementasi Mobile Apps Untuk Toko Highpom Menggunakan Framework Flutter

Implementation of Mobile Apps for Highpom Stores Using the Flutter Framework

Aldo Mahendra¹, Hendri Yanto², Wasti Nugia³, Saraswati⁴

^{1,2,3,4}STMIK Pontianak; Jl. Merdeka Barat No. 372 Pontianak Telp. (0561) 735555

Jurusan Teknik Informatika, STMIK Pontianak, Pontianak

e-mail: ¹aldomahendra2699@gmail.com, ²hendriyanto@stmikpontianak.ac.id,

³wasti.nurgia@stmikpontianak.ac.id, ⁴saraswati@stmikpontianak.ac.id

Abstrak

Penelitian ini dilandasi karena kemudahan yang diberikan oleh Mobile Apps banyak perusahaan yang memutuskan untuk menggunakan teknologi Mobile Apps untuk menjalankan kegiatan bisnisnya, diantara banyaknya Framework untuk membangun suatu mobile apps, Framework Flutter menjadi pilihan yang tepat karena membantu pengembang dalam menyesuaikan kebutuhannya seperti mendesain tampilan depan yang lebih modern pada aplikasi yang memiliki SDK open-source. Penelitian ini menggunakan metode penelitian Experimental dan metode perancangan Ekstreme Programming yang dapat mempermudah dalam proses pengembangan aplikasi mobile apps yang dihasilkan,. Hasil pada penelitian ini yaitu menghasilkan suatu MobileApps Toko Online pada Toko Highpom menggunakan framework flutter dengan menerapkan fitur flutter seperti hot reload untuk proses development yang cepat dan material design yang berfungsi untuk menerapkan UI/UX yang menarik. MobileApps Toko Online dengan framework flutter ini diharapkan dapat memiliki kontribusi bagi Toko Highpom dalam memberikan kemudahan dalam melakukan pemesanan produk secara online. Kesimpulan yang dihasilkan adalah mobile apps pada Toko Highpom ini telah berhasil dikembangkan dengan menggunakan Framework Flutter yang mampu mengelola aktivitas toko secara online dimana toko online ini dapat mempermudah penjualan untuk melakukan update data pada produk menggunakan teknologi webservice yaitu API pada halaman admin pemilik toko dan sudah menerapkan payment gateway midtrans sebagai metode pembayaran secara online. Untuk Mengoptimalkan aplikasi ini sebaiknya menambahkan fitur fitur seperti log acitivity pengiriman barang untuk mengetahui lokasi barang terkini saat dikirim sangat diperlukan untuk meningkatkan kepercayaan konsumen saat membeli produk dari toko.

Kata kunci : E-Commerce, Framework Flutter, Mobile Apps, Hot Reload,

Abstract

This research is based on the convenience provided by Mobile Apps, many companies decide to use Mobile Apps technology to carry out their business activities, among the many Frameworks for building mobile apps, the Flutter Framework is the right choice because it helps developers adjust their needs, such as designing a front view that more modern on apps that have open-source SDKs. This study uses the Experimental research method and the Extreme Programming design method which can simplify the process of developing the resulting mobile apps. The results of this study are to produce an Online Store MobileApps at Highpom Store using the Flutter framework by implementing flutter features such as hot reload for a fast development process and material design that functions to implement attractive UI/UX. The MobileApps Toko

Online with the Flutter framework is expected to have a contribution to the Highpom Store in providing convenience in ordering products online. The resulting conclusion is that the mobile apps at the Highpom Store have been successfully developed using the Flutter Framework which is able to manage store activities online where this online shop can make it easier for sales to update data on products using webservice technology, namely the API on the shop owner's admin page and has implemented payment gateway midtrans as an online payment method. To optimize this application, it is better to add features such as logs of goods delivery activities to find out the current location of goods when they are sent, which is very necessary to increase consumer confidence when buying products from stores..

Keywords : *E-Commerce, Framework Flutter, Mobile Apps, Hot Reload, Exstreme Progamming*

1. PENDAHULUAN

Hampir semua perusahaan menggunakan *MobileApps* karena dalam penggunaannya lebih fleksibel[1]. Karena kemudahan yang diberikan oleh *Mobile Apps* banyak perusahaan yang memutuskan untuk menggunakan teknologi *Mobile Apps* untuk menjalankan kegiatan bisnisnya[2]. Berdasarkan penelitian yang dilakukan Hoosuite mencatat bahwa pengguna internet di Indonesia sebanyak 202,5 juta orang per januari 2021 sementara jaringan mobile aktif mencapai 345,3 juta, atau 125,6% dari total populasi, angka ini lebih tinggi lantaran ada penduduk yang menggunakan internet lebih dari satu gawai saat beraktivitas di internet[3].

Dari laporan tersebut cukup menjadi bukti bahwa peluang penggunaan aplikasi *mobile* juga cenderung memiliki persentase tinggi. Namun, dalam membangun aplikasi *mobile* khususnya android, dibutuhkan bahasa pemrogramman seperti Java. Hal ini dapat menyulitkan programmer dalam membangun aplikasi berbasis *android* karena harus membuat tampilan depan aplikasi secara manual, untuk memudahkan *programmer* dalam membangun aplikasi dibutuhkan *Framework*. Diantara banyaknya *Framework*, *Flutter* menjadi pilihan yang tepat karena membantu dalam menyesuaikan kebutuhannya seperti mendesain tampilan depan yang lebih modern pada aplikasi yang memiliki *SDK open-source* untuk mengembangkan aplikasi seluler berkinerja tinggi dan lebih andal untuk sistem operasi seperti *iOS* dan *Android* [4].

Alasan menggunakan *framework flutter* adalah karena *Framework Flutter* merupakan *User Interface Framework* dan *Software development kit open source* berbasis *Widget Based Apps*, dimana semua komponen dalam perangkat lunak ini adalah *widget* yang berfungsi sebagai blok pembangun untuk membentuk antarmuka pengguna aplikasi [5]. *Flutter* mempunyai dua macam *widget* untuk pengembang aplikasi, yaitu *Material Design* dan *Cupertino* [6]. *Material Design* adalah bahasa *design* yang dibuat oleh Google, design ini sama dengan design yang dipakai pada *Android*. *Cupertino* atau dengan sebutan lain gaya *iOS* adalah bahasa design yang dipakai oleh *iOS*. *Flutter* mempunyai lebih banyak *widget Material Design* daripada *widget Cupertino* pada OS perangkat yang berbeda, *widget* bisa dipakai secara *crossplatform* [7].

Dalam penerapan toko online pada Toko Highpom tentunya dapat memiliki potensi pembeli jauh lebih luas. Dalam penerapan toko online, hubungan personalisasi dengan para pelanggan menjadi lebih harmonis, komunikatif, dan lebih interaktif. Kegiatan dalam melakukan promosi juga tanpa membutuhkan biaya besar dengan sasaran yang lebih jelas, mekanisme pemesanan dan sistem pendistribusian yang cepat dalam suatu jaringan global, serta memiliki sejumlah alternatif dalam proses pembayaran, diantaranya melalui sistem transfer [8].

Hal ini tentunya dapat membantu meningkatkan penjualan dan pelanggan mempunyai konsisten pada Toko Highpom karena praktis dan kemudahan dalam mengakses informasi pada toko. Oleh karena itu, *mobile Apps* khususnya *Android* ini dirancang menggunakan *Framework Flutter* agar dapat digunakan di berbagai platform seperti *Android* dan memiliki tampilan desain

yang modern. Dengan begitu, pelanggan akan nyaman dalam penggunaan aplikasi dan lebih mudah mengakses toko melalui mobile Apps untuk melakukan aktifitas bisnis. Banyaknya fitur pada toko online menjadikan Framework Flutter dalam merancang mobile Apps ini, dimana programmer hanya perlu menguasai bahasa pemrograman Dart untuk mengimplementasikan Framework Flutter dalam merancang toko online ini yang dapat berjalan di mobile Apps khususnya Android.

Penelitian terdahulu yang juga relevan dengan toko online menggunakan *Framework Flutter* pada PT. Putra Sumber Abadi mengatakan bahwa disimpulkan secara fungsional sistem sudah dapat menghasilkan hasil yang diharapkan dengan Framework Flutter walaupun belum sempurna, tetapi aplikasi yang dihasilkan memiliki tampilan yang cukup menarik, dan dapat mempercepat pengolahan data [9].

Berdasarkan paparan latar belakang dan penelitian sebelumnya, maka dalam proses perancangan toko *online* ini akan mengimplementasikan *Framework Flutter* pada Toko Highpom yang menggunakan bahasa pemrograman *Dart* dengan Framework Flutter programmer lebih cepat dalam mengembangkan aplikasi karena sudah terdapat *material design* untuk membangun tampilan *front-end* yang menarik.

2. METODE PENELITIAN

Metode penelitian yang digunakan oleh penulis pada penelitian ini adalah metode eksperimental. Pemilihan metode ini tujuannya untuk mendapatkan pendekatan komputasi yang berfokus pada manusia sebuah aplikasi *Mobile Apps* yang diinginkan oleh pengguna tersebut dengan variable yang telah ditetapkan pada studi pendahuluan.

Metode eksperimental dikembangkan untuk mengukur perilaku Pengguna dalam upaya untuk menentukan strategi secara empiris yang dapat digunakan manusia secara efektif [10] Strategi secara empiris yang dimaksud adalah dengan melakukan pengetasan pada sebuah kode program sehingga diperoleh parameter (pengukuran) yang merepresentasikan kualitas perangkat lunak tersebut, proses tersebut meliputi requirement, analisis, desain, implementasi dan testing, serta evaluasi. Dengan demikian perangkat lunak yang dihasilkan akan memiliki produk perangkat lunak yang efisien, murah, memenuhi kebutuhan dan kepuasan user [11]

Metode eksperimental memiliki langkah langkah sebagai berikut :

1. Perencanaan , proses perencanaan sangat penting untuk meningkatkan proses keberhasilan dalam penelitian.
2. Desain, setelah melakukan perencanaan tahap selanjutnya adalah melakukan desain visualisasi terhadap perencanaan yang sudah dibuat dalam susunan garis, bentuk, ukuran, dan juga nilai yang dibuat menurut prinsip prinsip desain agar memiliki nilai yang bisa digunakan.
3. Peforma, setelah melakukan tahapan desain tahap selanjutnya adalah melakukan peforma suatu tolak ukur untuk menilai kualitas dari sumber daya yang dihasilkan.
4. Analisis data, setelah melakukan peforma tahap selanjutla adalah melakukan analisis data yaitu seperti pengujian terhadap produk yang dihasilkan untuk mendapatkan hasil yang terbaik.
5. Evaluasi, setelah melakukan berbagai macam tahapan selanjutnya adalah melakukan evaluasi produk dari tahapan tahapan yang sudah dilakukan.

3. HASIL DAN PEMBAHASAN

3.1 Perencanaan Penelitian Eksperimental

Pada tahap perencanaan, penelitian yang digunakan dalam membangun sistem menggunakan metode analisis dan pengembangan yaitu metode (*Extreme Programming*), pengembangan dengan menggunakan metode ini memiliki beberapa tahapan yang terdiri dari perencanaan, desain, coding, serta testing. Peneliti menganalisis setiap kebutuhan yang akan

digunakan dalam membangun perangkat lunak. Adapun kebutuhan ini dimulai dari mencari masalah-masalah yang dapat diangkat dalam penelitian, kemudian mencari referensi-referensi dari penelitian terdahulu untuk mendukung penelitian yang dibahas. Setelah seluruh informasi yang didapat terpenuhi, maka tahapan selanjutnya adalah menganalisis setiap kebutuhan.

Table 1 Table Planning

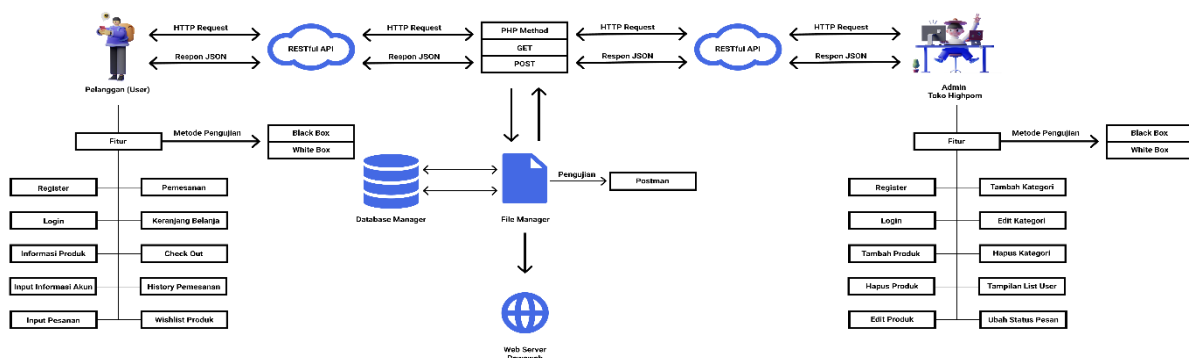
NO	Masalah	Solusi	Fitur Pada <i>Mobile Apps</i>
1	Belum Tersedia Sistem Untuk Mengetahui Ketersediaan Produk	Menyediakan Sistem Admin Untuk Mengelola Produk	Menu Kelola Produk
2	Mencatat Transaksi yang masih manual di buku bulanan.	Menyediakan Sistem Admin Untuk Mengelola Transaksi	Menu Kelola Transaksi
3	Melakukan Penjualan Online Di Sosial Media	Menyediakan <i>Mobile Apps Delivery Order</i> .	Menu Home Pages.
4	Melakukan Proses Pembayaran Yang masih dengan sistem COD	Menyediakan Payment Gateway Untuk Proses Pembayaran	Menu Payment Gateway

3.2 Desain Eksperimental

Pada tahapan desain, semua kebutuhan yang terdiri dari pengguna, sistem, perangkat keras dan lunak diuraikan serta digunakan sebagai dasar untuk merancang sebuah desain perangkat lunak yang dibangun. Adapun desain sistem perangkat lunak yang dibangun meliputi perancangan pemodelan arsitektur perangkat lunak yang menggunakan UML dalam bentuk diagram yaitu *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dengan basis data yang dirancang menggunakan *Entity Relationship Diagram* dan mendesain tampilan *interface* menggunakan Figma

3.2.1 Arsitektur Sistem

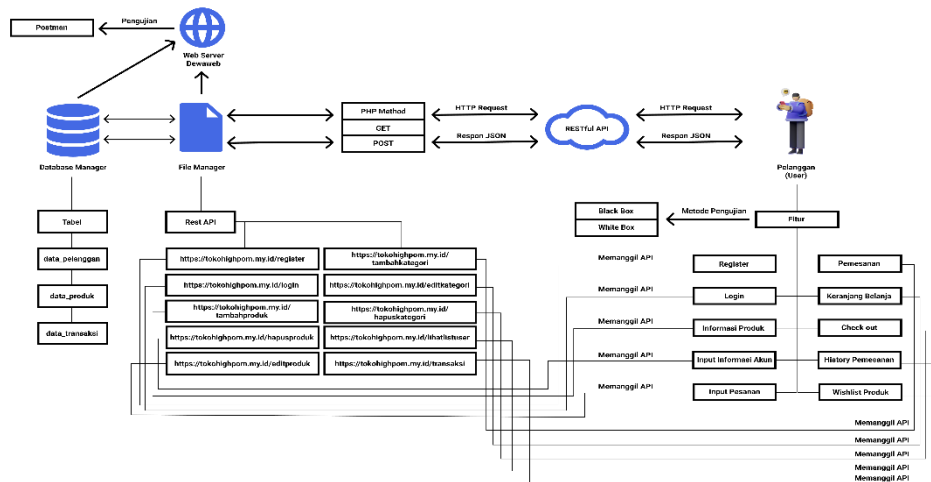
Aplikasi mobile yang di bangun ini terintegrasi oleh *API (Application Programming Interface)* yang menggunakan metode *RESTful web service*. Pertukaran data antara sistem dan *REST API* dengan format JSON untuk pengambilan dan pengiriman data melalui *web service*. Perancangan arsitektur sistem ini dimaksudkan untuk memberikan gambaran dari kebutuhan pengguna dan sistem.



Gambar 2 Arsitektur Sistem

Aplikasi yang sudah terinstall pada perangkat pengguna nantinya dapat melakukan pertukaran data dengan *REST API* dengan format JSON yang terkonfigurasi dengan database toko online Highpom. Ketika pengguna meminta suatu perintah maka *API* dengan format JSON

memproses kemudian meneruskan ke database dan mengembalikan lagi informasinya untuk diterima oleh *Client*. Tentunya database yang ada pada toko online Highpom sudah diinput oleh admin toko tersebut. Pada *dashbord* admin dapat mengetahui informasi data yang telah diinput oleh pengguna melalui *API* dengan format JSON.

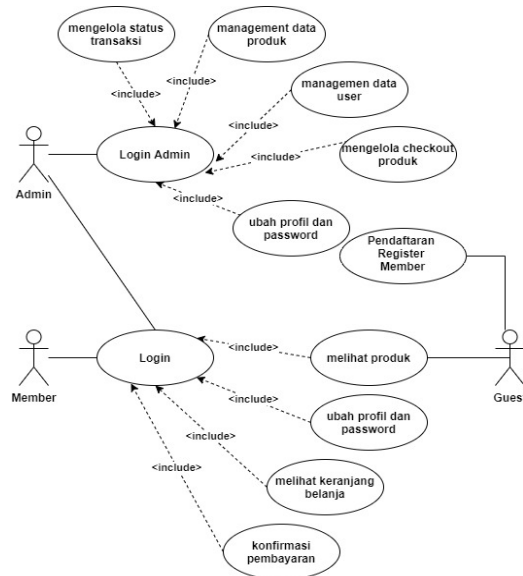


Gambar 3 Arsitektur Sistem Pengguna

Setelah berhasil melakukan *login*, pengunjung dapat membuka kategori produk atau melihat produk rekomendasi yang ditawarkan, pengunjung dapat langsung memilih produk yang diinginkan dan langsung mememesannya atau menandai produk favorit, sehingga produk tersebut bisa dilihat kapanpun tanpa mencarinya kembali untuk melakukan pemesanan dilain waktu. Setelah memasan pengunjung bisa melihat rincian pemesanan mereka. Selanjutnya apabila pemesanan menyetujui produk pesanannya, pengunjung bisa melakukan transaksi pembayaran melalui nomor rekening yang tertara. Kemudian, setelah sukses melakukan pembayaran Admin akan meverifikasi pembayaran tersebut lalu produk akan diproses untuk pengiriman ke pemesan.

3.3 Use Case Diagram

Adapun pemodelan *Use Case diagram* digunakan untuk menjelaskan sistem perancangan dan mengetahui fungsi apa saja yang ada pada *mobile apps* toko online ini. Aktor mewakili peran orang atau sistem yang lain ketika berkomunikasi dengan *use case*. Aktor yang berperan pada perangkat lunak toko online ini adalah :

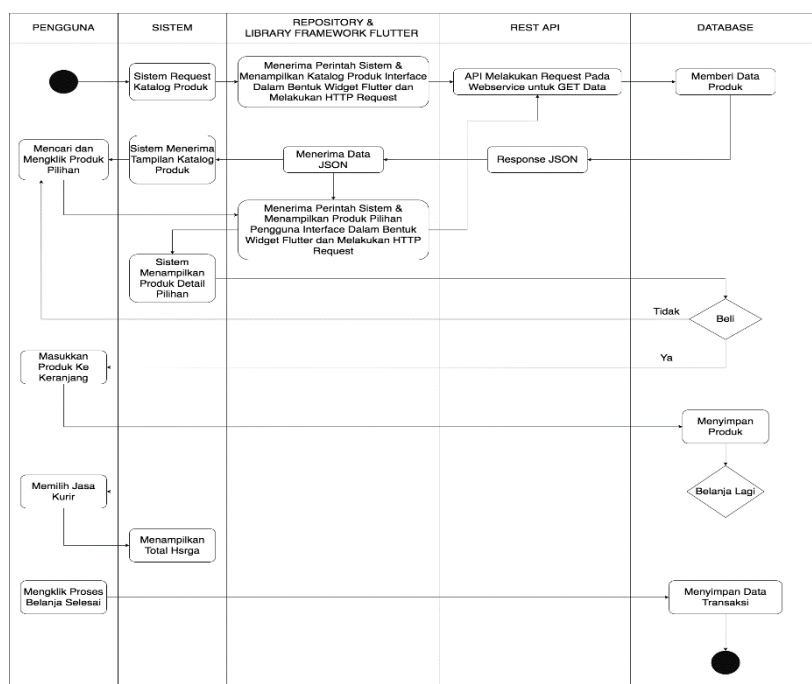


Gambar 4 Use Case Diagram Admin, Guest dan Member

Berdasarkan gambar 4, terdapat 3 aktor yang berperan sebagai pengguna toko *online* yaitu *guest*, *member* dan *admin*. Dimana pada *member* dan *guest* memiliki kesamaan yaitu dapat melihat produk yang dijual. Adapun tabel-tabel deskripsi yaitu:

3.4 Activity Diagram

Activity diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang. *Activity* merupakan state *diagram* khusus, di mana bagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu *Activity diagram* tidak menggambarkan behaviour internal sebuah sistem dan interaksi antar subsistem secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Adapun *Activity diagram* dalam perancangan dan pengembangan perangkat lunak yang dibuat yaitu :

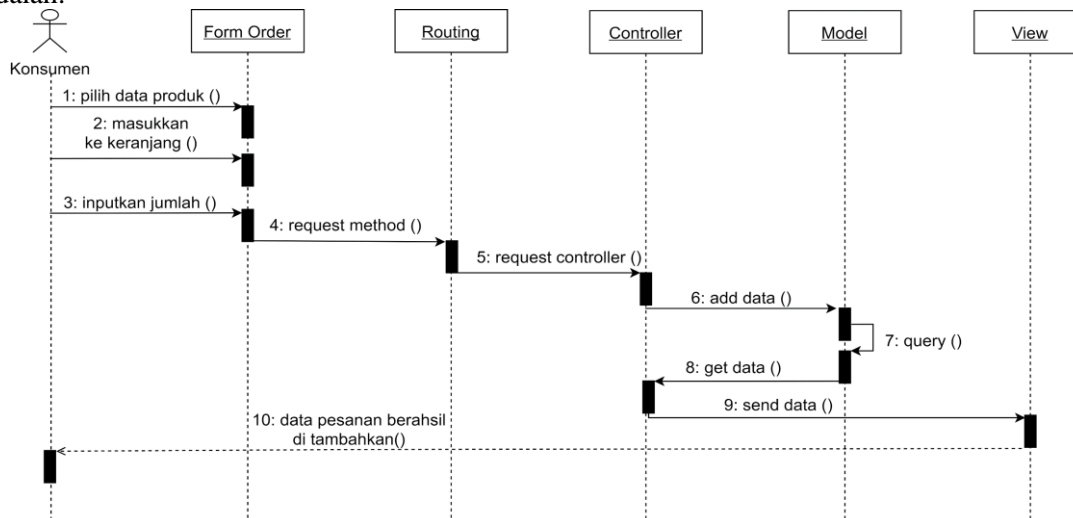


Gambar 5 Activity diagram Checkout

Pada gambar 5 member dapat langsung melakukan pencarian terhadap barang. Untuk dapat berbelanja ember harus login terlebih dahulu, kemudian masuk ke halaman home, pelanggan diberikan opsi produk dan cari produk, member memilih form lanjutkan belanja untuk memilih produk yang lain, setelah itu sistem akan melanjutkan ke halaman porduk detail untuk memilih barang. Setelah member memilih produk yang ingin dibeli, member dapat memilih jasa ongkir yang ingin digunakan. Setelah member memilih jasa pengiriman, member mengklik tombol selesai belanja. Sistem akan menyimpan data pemesanan pada database transaksi, dan akan menampilkan no invoice pemesanan untuk member

3.5 Sequence Diagram

Sequence diagram adalah *diagram* yang digunakan untuk menggambarkan interaksi antara objek didalam dan disekitar sistem. *sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek yang terkait). *sequence diagram* digunakan dalam menggambarkan rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah event dalam menghasilkan suatu output tertentu. Adapun *sequence diagram* dalam perancangan perangkat lunak yang dibuat adalah:

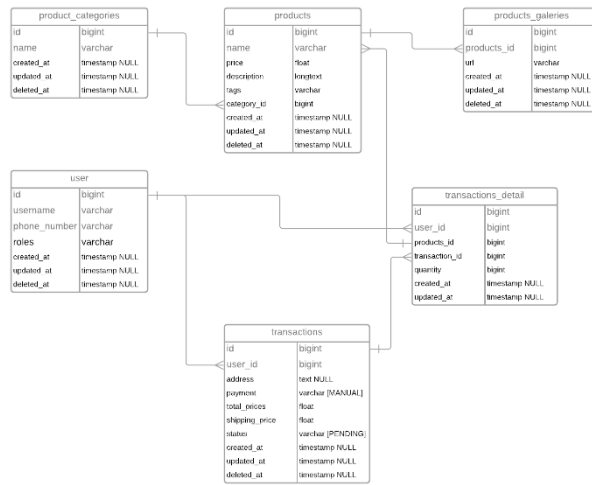


Gambar 6 Sequence Diagram Kelola Produk

Pada gambar 6 konsumen memilih produk dengan cara menampilkan detail produk dan menambahkan produk tersebut kedalam keranjang belanja serta menginputkan jumlah produk. Semua produk yang dipesan akan dimasukan kedalam keranjang belanja. Setiap produk yang ada dikeranjang belanja dapat dihapus atau ditambah.

3.6 Class Diagram

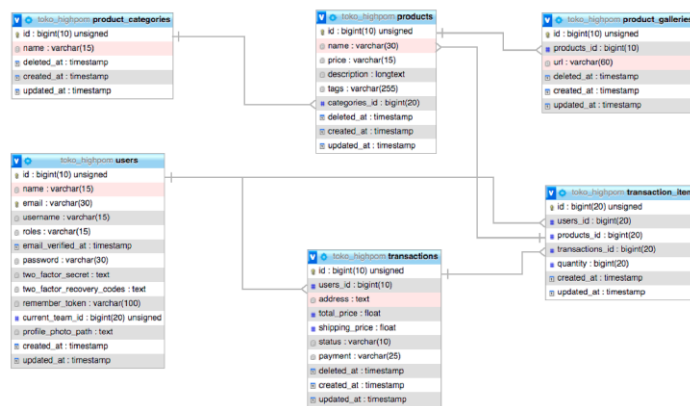
Adapun *class diagram* digunakan untuk menampilkan beberapa kelas serta paket-paket yang terdapat pada sistem *website* yang dibuat. Pemodelan *class diagram* yang dirancang dalam mengembangkan Toko Highpom adalah:



Gambar 7 Class Diagram

Pada gambar 7 merupakan Class Diagram yang berfungsi untuk perencanaan struktur program database yang akan digunakan pada perancangan aplikasi Mobile Apps ini.

3.7 Diagram Hubungan Entitas

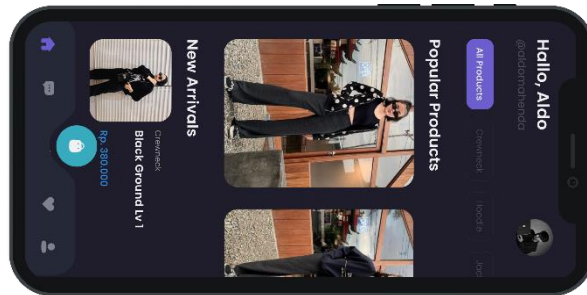


Gambar 8 Diagram Hubungan Entitas

Diagram hubungan entitas pada gambar 8 adalah perancangan model database yang menampilkan berbagai data yang memiliki hubungan dengan basis data yang nantinya akan dibuat dan saling terhubung.

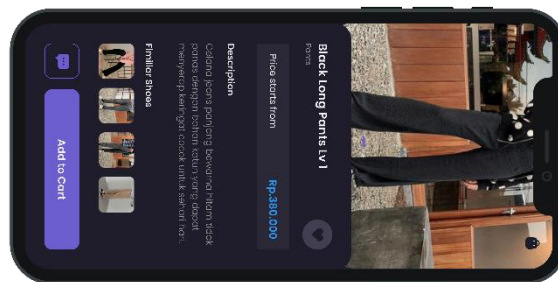
3.8 Interface Pada iOS (Iphone 12 Pro Max)

Berikut ini tampilan interface *mobile apps* pada saat di run menggunakan Visual Studio Code yang menggunakan emulator ios pada tampilan iPhone 12 Pro Max.



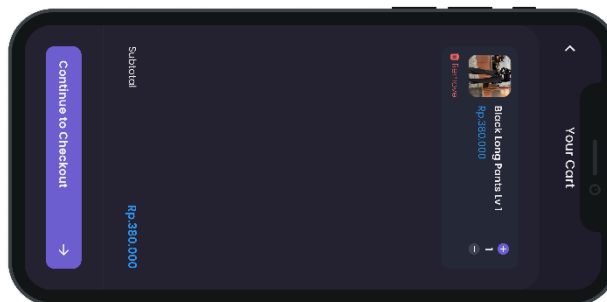
Gambar 9 Interface Form Beranda

Pada gambar 9 merupakan halaman form beranda yang berfungsi menampilkan card sebagai informasi tampilan produk serta harga dan beberapa produk lainnya seperti dibagian New Arrivals.



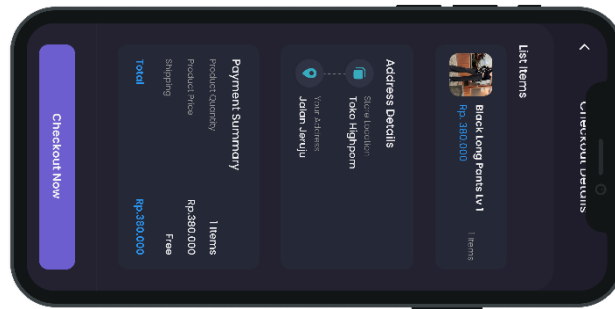
Gambar 10 Interface Detail Produk

Pada gambar 10 merupakan tampilan detail produk yang tersedia di aplikasi pada *Mobile Apps* Penjualan pada Toko Highpom berfungsi untuk mengetahui informasi detail produk dan melakukan *checkout*.



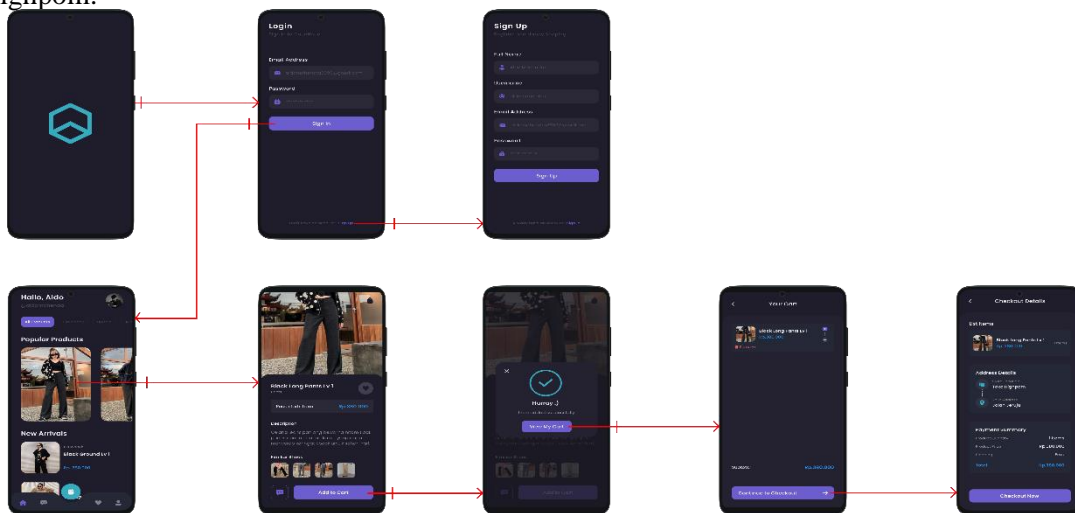
Gambar 11 Interface Cart Checkout

Pada gambar 11 merupakan tampilan cart checkout yang tersedia di aplikasi pada *Mobile Apps* Penjualan pada Toko Highpom berfungsi untuk mengetahui melakukan checkout produk yang tersedia pada Toko Highpom.



Gambar 12 Interface Checkout

Pada gambar 12 merupakan tampilan fitur checkout yang tersedia di aplikasi pada *Mobile Apps* Penjualan pada Toko Highpom berfungsi melakukan pembelian pada produk yang sudah dimasukkan ke dalam cart atau keranjang sehingga produk pemesanan akan diterima oleh Toko Highpom.

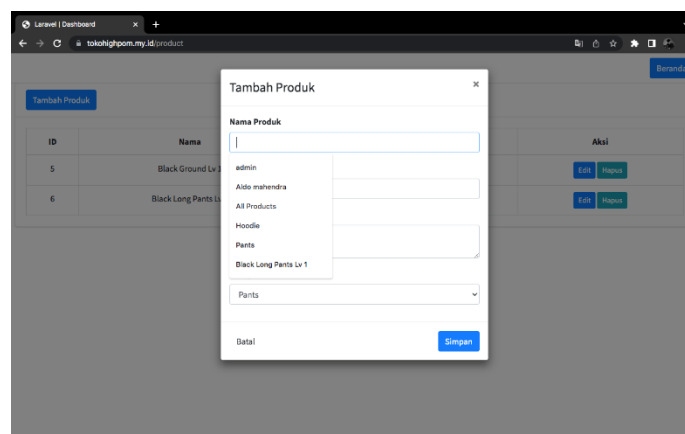


Gambar 13 Halaman Tambah Produk

Pada gambar 13 merupakan perancangan flow aplikasi sebelum melakukan slicing pada program aplikasi agar lebih jelas dan terarah.

3.9 Interface Admin

Berikut ini tampilan halaman admin menggunakan Framework Laravel yang sudah dihosting dengan alamat <https://tokohighpom.my.id/> :



Gambar 14 Halaman Tambah Produk

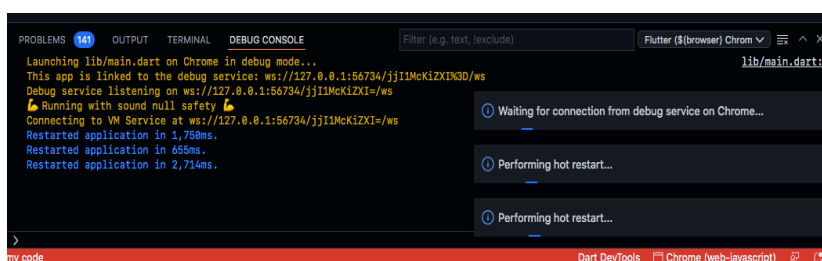
Gambar 14 merupakan pop up tambah produk yang muncul pada halaman produk yang berfungsi untuk admin dalam menambah produk dan informasi produk pada Toko Highpom yang akan di akses oleh pengguna dalam mengetahui informasi produk.

3.10 Analisis Data Eksperimental

Analisa data merupakan tahap eksperimen program dan pengujian yang menggunakan *Black Box Testing* untuk mengetahui hasil dari slicing program yang menggunakan *Framework Flutter* seperti kecepatan fitur yang ada pada *Framework Flutter* yaitu *Hot Reload*, berikut merupakan tahapan tahapannya :

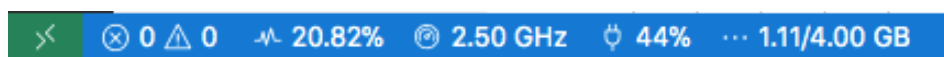
3.10.1 Eksperimen Program

Eksperimen program pada *Framework Flutter* merupakan tahapan untuk melakukan test performa suatu aplikasi yang di buat, dalam eksperimen program pada *Framework Flutter* ada beberapa tahapan yang dilakukan, masing – masing tahapan dilakukan secara berkala untuk melihat kecepatan akses suatu aplikasi, berikut adalah beberapa tahapan eksperimen pada *Framework Flutter*:



Gambar 15 Hot Reload Framework Flutter

Pada gambar 15 merupakan hasil *Hot Reload* yang dilakukan pada program aplikasi, hasil kecepatan *Hot Reload* yang dilakukan adalah 873ms.



Gambar 16 Hasil Performa Penggunaan Framework Flutter

Bisa dilihat seperti gambar 16 yang merupakan hasil pengujian CPU dengan aplikasi menggunakan *Framework Flutter* yang telah dibuat sebelumnya, dari hasil pengujian tersebut dinyatakan dapat berjalan dengan baik, karena selama detik setelah running test CPU yang terpakai 18.53%.

Table 2 Hasil Pengujian Framework Flutter

No	Pengujian yang dilakukan	Hasil	Waktu Akses	Ukuran File	Hasil yang diharapkan
1	Menjalankan Program Aplikasi	Berjalan baik	888ms	4.00GB	Sesuai yang diharapkan
2	Pengujian Kecepatan Request data	Berjalan Baik	2.50 GHz	4.00 GB	Sesuai yang diharapkan
3	Pengujian performa CPU	Berjalan baik	18.53%	4.00 GB	Sesuai yang diharapkan
4	Pengujian performa memori	Berjalan baik	1,3/ 4.00GB	4.00 GB	Sesuai yang diharapkan

Dari tabel 12 merupakan pengujian yan dilakukan terhadap *Framework Flutter* untuk mendapatkan parameter pengukuran dsari hasil pembangunan *Mobile Apps* pada Toko Highpom seperti kecepatan request data, peforma CPU, dan pengujian terhadap memori yang digunakan.

3.11 Pengujian Perangkat Lunak

Testing atau pengujian perangkat lunak adalah menguji semua elemen-elemen pada sistem perangkat lunak untuk mengetahui kesesuaian dari perancangan perancangan perangkat lunak yang dibangun

3.11.1 Black Box Testing

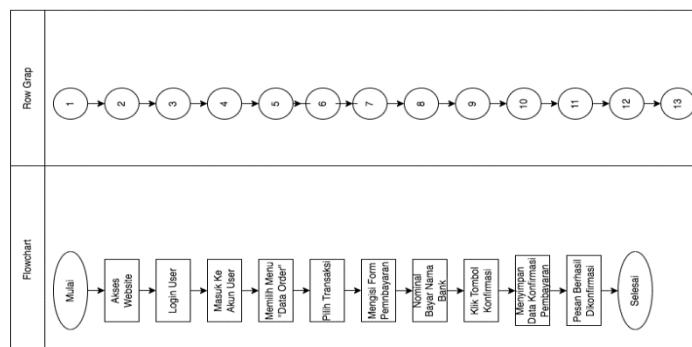
Pengujian ini dilaksanakan oleh user dengan metode pengujian menggunakan *Black-Box Testing*. Salah satu jenis pengujian *Black-Box Testing* adalah dengan menggunakan teknik *Equivalence Partitioning (EP)* yang penulis gunakan untuk menguji masukan serta membagi masukan kedalam kelompok-kelompok berdasarkan fungsinya. Sehingga didapatkan sebuah test case yang akurat. Adapun pengujian *Black-Box Testing* dari perancangan *mobile apps* yang dibuat yaitu:

Tabel 3 Tabel Pengujian *Black-Box* Pada Pengguna

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Melakukan pemesanan barang jumlah lebih dari 0	User memasukan jumlah barang lebih dari 0 pada setiap produk yang ingin dipesan	Pesanan berhasil ditambahkan ke <i>database</i>	Sesuai Harapan	<i>Valid</i>
2.	Melakukan pemesanan barang jumlah kurang dari 1	User memasukan jumlah produk kurang dari 1 yang ada di keranjang belanja dan melakukan pemesanan	Pesanan tidak berhasil dilakukan dan gagal menyimpan data ke <i>database</i>	Sesuai Harapan	<i>Valid</i>
3.	Melengkapi seluruh alamat pengiriman produk	User mengisi form pengiriman dan melakukan submit	Berhasil menambahkan alamat pengiriman	Sesuai Harapan	<i>Valid</i>
4.	Tidak mengisi salah satu kolom alamat pengiriman	User tidak mengisi salah satu form alamat pengiriman	Gagal menambahkan alamat pengiriman	Sesuai Harapan	<i>Valid</i>

Pada tabel 2 merupakan pengujian fungsi terhadap *Fronnd End* aplikasi pada pengguna untuk mengavaluasi dari aplikasi yang dibangun menggunakan *Framework Flutter*. Dari hasil pengujian terhadap navigasi pada aplikasi, semuanya berhasil untuk melakukan integrasi pengambilan data.

3.11.2 White Box Testing



Gambar 17 Pengujian (Testing) Konfirmasi Pembayaran

Berdasarkan gambar 17 adapun jalur pengujian yaitu :

$$V(G) E - N + 2$$

E = Jumlah Edge Flowgraph

N = Jumlah Simpul Flowgraph

Sehingga kompleksitas pada flowgraph data adalah :

$$V(G) = 12 - 13 + 2 = 1$$

$$\text{Jalur R1} = 1 - 2 - 3 - 4 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 13$$

4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan, maka kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut:

- a. Perangkat lunak penjualan berbasis *Mobile Apps* pada Toko Highpom ini telah berhasil dikembangkan dengan menggunakan Framework Flutter yang mampu mengelola aktivitas toko secara online.
- b. Dengan adanya sistem penjualan online (*e-commerce*) maka toko dapat memudahkan proses transaksi yang selama ini sifatnya konvensional menjadi lebih modern dengan transaksi *online* menggunakan *Payment Gateway* yaitu Midtrans.
- c. Dengan adanya toko online ini dapat mempermudah penjualan untuk melakukan update data pada produk menggunakan teknologi webservice yaitu API pada halaman admin pemilik toko.
- d. Kekurangan pada pengembangan *mobile apps* ini belum disertai kerja sama kurir untuk pengantaran barang, sehingga untuk barang yang dijual hanya wilayah yang dapat terjangkau oleh toko.

5. SARAN

Walaupun aplikasi berbasis mobile yang dirancang dapat memberikan solusi tentang permasalahan yang terjadi, namun sistem yang dirancang masih jauh dari sempurna. Maka dari itu penulis memberikan saran – saran sebagai berikut:

- a. Adanya kerja sama dengan kurir yang sudah ada seperti Gojek, Grab, JNT, dan JNE agar dapat mengirim produk lebih jauh untuk mendapatkan konsumen jauh lebih banyak.
- b. Website penjualan yang peneliti usulkan ini dapat diterapkan dengan mengimplementasikan teknologi yang sudah ada dan dapat membantu dalam melakukan penjualan dan mengoptimalkan pekerjaan, sebaiknya sistem penjualan yang dilakukan secara manual diperbarui dengan menerapkan sistem yang terkomputerisasi.

- c. Penambahan fitur seperti *log acitivity* pengiriman barang untuk mengetahui lokasi barang terkini saat dikirim sangat diperlukan untuk meningkatkan kepercayaan konsumen saat membeli produk dari toko.

DAFTAR PUSTAKA

- [1] Datareportal, (2022, Mei). Retrieved from www.datereportal.com: <https://datereportal.com/reports/digital-2021-indonesia>.
 - [2] Hootsuit, (2021, Mei). Retrieved from www.hootsuit.com: <https://hootsuit.com.l.com/reports/>.
 - [3] Brunette, E., 2019, *Android: Introducing Google's Mobile Development Platform. (3rd ed)*, The Pragmatic Bookshelf., New York.
 - [4] Tashildar, Aakanksha., 2020, Application Development Using Flutter, IRJEMTS, Vol. 2, No. 8, pp. 1262-1266.
 - [5] Seputra, K. A., dan Sandiasa, G., 2020, Rancang Bangun Sistem Informasi Satgas Gotong Royong (Si Garong) Desa Adat Berbasis Mobile, Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI, 9(3), 338-350.
 - [6] Tran, Thanh., 2020, Flutter Native Performance and Expressive UI/UX.
 - [7] Tjandra, S., dan Chandra, G. S., 2020, Pemanfaatan Flutter dan Electron Framework pada Aplikasi Inventori dan Pengaturan Pengiriman Barang, Journal of Information System, Graphics, Hospitality and Technology, 2(02), 76-81.
 - [8] Kosasi, Sandy., 2016, Perancangan Bisnis Toko Online untuk Memperluas Pangsa Pasar Aksesoris Produk Rohani." *Semnaseknomedia Online* vol.4, no.1 pp.2-3.
 - [9] Hendriawan, Muhammad, dkk. "PENGEMBANGAN APLIKASI E-COMMERCE DI PT. PUTRA SUMBER ABADI MENGGUNAKAN FLUTTER." *Journal of Information System, Informatics and Computing* 5.1 (2021): 69-88.
 - [10] Cash, Philip, Tino Stanković, and Mario Štorga. "Experimental design research." Cham: Springer International Publishing (2016).
 - [11] Widjaja, Yulia. "PENDEKATAN EMPIRIS DALAM REKAYASA PERANGKAT LUNAK." *Jurnal INFORMASI* Vol 4.1 (2017): 4. [12] Del Sole, Alessandro., 2021, *Visual Studio Code Distilled: Evolved Code Editing for Windows, MacOS, and Linux*. Jerman, Apress.
 - [13] Di Marzio, Jerome F.. Beginning., 2017, *Android Programming with Android Studio*. Amerika Serikat, John Wiley & Sons, Incorporated.
 - [14] Doglio, Fernando, Doglio, dan Corrigan., 2018, *Pengembangan REST API dengan Node.js js* . Apres.,
 - [15] Ernawati, Siti, dan Risa Wati., 2021, Aplikasi Quran Berbasis Android Pada Flutter Framework Dengan Menggunakan Model Fountain, *Jurnal Riset Informatika*, vol.3, no.2, pp.195-202.
- Introduction to Mobile App Development*. Inggris, O'Reilly Media, Incorporated.