

Pemanfaatan Aplikasi Mobile Menggunakan Framework Flutter Pada Toko Outfitsecondbrand

Utilizing Mobile App Using Flutter Framework for Outfitsecondbrand Store

Tri Rudiant Winanto¹, Handy Kusuma², Ponti Harianto³

^{1,2,3}STMIK Pontianak; Jl. Merdeka No. 372 Pontianak, (0561) 735555/ (0561) 737777

Jurusan Teknik Informatika, STMIK Pontianak, Pontianak

e-mail: ¹triwinantopmk@gmail.com, ²handy.kusuma@stmikpontianak.ac.id,

³ponti.harianto@stmikpontianak.ac.id

Abstrak

Pemakaian teknologi mobile sangat efektif dalam kegiatan berbelanja karena dapat memberikan kemudahan, kenyamanan, mobilitas, dan fleksibilitas. Toko Outfitsecondbrand memasarkan produknya melalui media social, penjualan melalui media sosial ini memiliki banyaknya saingan bisnis pada media sosial dan pembaruan informasi produk yang tidak efektif. Maka dari itu, toko Outfitsecondbrand memerlukan aplikasi mobile sendiri untuk meningkatkan penjualan dan efektifitas. Aplikasi mobile yang akan dibangun menggunakan framework flutter. Bentuk penelitian yang digunakan adalah studi kasus dengan metode penelitian Design Science Research, teknik pengumpulan data berupa observasi dan wawancara. Metode pengembangan yang digunakan Extreme Programming, metode pengujian yang digunakan ialah black-box. Hasil akhir dari penelitian terdapat dua sistem yaitu aplikasi mobile digunakan user dan website digunakan oleh admin. Selain itu dapat mengintegrasikan API Rajaongkir untuk pengecekan ongkos kirim dan API Firebase Cloud Message untuk notifikasi. Dengan toko online dapat menjangkau pemasaran yang lebih luas, pengelolaan data toko lebih mudah, serta memungkinkan untuk bertransaksi tanpa batasan waktu dan lokasi sehingga dengan adanya aplikasi toko online sendiri permasalahan yang ada dapat teratasi. Untuk menyempurnakan toko online ini, dapat ditambahkan fitur baru seperti chat, dan ulasan produk.

Kata kunci— Mobile, Flutter, DSR, Toko Online

Abstract

The use of mobile technology is very effective in shopping activities because it can provide convenience, comfort, mobility and flexibility. Outfitsecond stores market their products through social media, sales through social media have many business competitors and ineffective product information updates. Therefore, Outfitsecond stores need their own mobile app to increase sales and effectiveness. Mobile applications are built using the Flutter framework. The form of research is a case study, the research method is Design Science Research and data collection techniques are in the form of observation and interviews. The development method is Extreme Programming, the testing method is black-box. The result is that there are two systems, namely the mobile application used by the user and the website used by the admin. Can integrate Rajaongkir API for checking postage and Firebase Cloud Message API for notifications. With online stores, you can reach broader marketing, store data management is easier, and it is possible to transact without time and location limitations so that with the online store application itself, existing problems can be resolved. To enhance this online store, new features can be added such as chat and product reviews.

Keywords— Mobile, Flutter, DSR, Online Store

1. PENDAHULUAN

Teknologi informasi sudah merubah konsep diberbagai bidang seperti bisnis, hiburan, dunia pendidikan, pemerintahan dan lain sebagainya, perkembangan teknologi informasi dapat menjadi sarana pembantu yang tepat, cepat, dan aman [1]. Pemilihan menggunakan *mobile* daripada *website* juga karena sekarang layanan yang tersedia dalam *mobile* telah memungkinkan komunikasi, kolaborasi dan koordinasi melalui perangkat *mobile* [2], salah satu bidang bisnis yang dijalankan dengan bantuan teknologi *mobile* adalah toko *online*. Pemakaian teknologi *mobile* sangat efektif dalam kegiatan berbelanja karena dengan teknologi ini dapat memberikan kemudahan dalam pencarian, kenyamanan pasca pembelian, mobilitas dan fleksibilitas [3].

Keberadaan bisnis toko *online* memudahkan konsumen atau pelanggan dapat melakukan transaksi bisnis tanpa batasan ruang dan waktu [4]. Dalam penjualan secara *online* toko Outfitsecondbrand memasarkan produknya melalui media sosial, penjualan yang di lakukan melalui media sosial memiliki banyak saingan bisnis, membuat pemilik toko tidak bisa melayani pelanggan dengan cepat. Untuk memperbarui informasi produk pemilik toko harus memperbarui informasi satu-persatu di tiap media sosial dan melakukan promosi kepada calon pembeli berpotensi tidak tepat sasaran. Dalam melakukan perancangan aplikasi *mobile* khusus nya android ini dibutuhkan bahasa pemrograman seperti *Java*. Hal ini dapat menyulitkan dalam membangun aplikasi karena harus membuat tampilan depan secara manual, untuk memudahkan dalam membangun aplikasi dibutuhkan pendekatan *framework*. Keuntungan utama dari pendekatan ini, selain menghemat waktu juga kehandalan dan memungkinkan untuk membuat sebuah aplikasi *mobile* lebih interaktif [5], salah satunya dengan menggunakan *framework flutter*.

Framework flutter merupakan *user interface framework* dan *software development kit open source* berbasis *Widget Based Apps*, dimana semua komponen adalah *widget* [6]. *Framework flutter* mendukung *multiplatform* menggunakan *single code* dengan bahasapemrograman *Dart* untuk *platform* apapun yang meningkatkan efisiensi dalam pengembangan [7]. Alasan menggunakan *framework flutter* karena *flutter* sudah menghilangkan *bridge* dan *OEMplatform* untuk dapat berkomunikasi langsung dengan *CPU* dan *GPU*, tujuannya adalah memudahkan pengembang melakukan perubahan aplikasi sesering mungkin, memudahkan penggunaan *platforms API* melakukan sinkronisasi sistem pesan. Cara kerja *flutter* yaitumenggabungkan beberapa *widget* pada *platform (Canvas/events)* dan memanfaatkan *platformchannels* untuk mengakses layanan *smartphone* (lokasi, audio, kamera, *bluetooth*, sensor, dll) [8]. Dalam penelitian sebelumnya, menggunakan konsep *e-commerce* menggunakan *framework flutter* [9] terbukti dapat diterapkan dari segi sistem membantu penjual dalam mempromosikan barangnya secara *online*, memudahkan dalam membeli barang tanpa dibatasiwaktu dan tempat, dapat memberikan informasi yang tepat kepada pembeli. Penelitian lainya, pada toko *online* menggunakan *framework flutter* [10] menyatakan bahwa dengan aplikasi toko *online* banyak manfaat dari segi waktu, pemasaran, dan segi kemudahan dalam berbelanja seperti melakukan transaksi pemesanan dan penjualan barang, dapat membangun *user interface* nya lebihmoderen. Penelitian lain nya, pada aplikasi penjualan tiket kelas pelatihan menggunakan *framework flutter* [11] maka dapat disimpulkan bahwa aplikasi yang di buat ini telah memenuhstandar ketentuan untuk digunakan dan dapat membantu memudahkan pengguna dalam mencari kegiatan pembelajaran yang diinginkan tanpa dibatasi waktu dan tempat, dari ketiga penelitiansebelum terdapat kekurangan yaitu tidak ada fitur *chat* melalui aplikasi atau pun terhubungdengan aplikasi *chat* lain nya seperti whatsapp, tidak ada fitur notifikasi dan fitur ongkos kirim. Berdasarkan paparan latar belakang dan mengacu pada penelitian sebelumnya, maka akan dibangun sebuah aplikasi *mobile* pada toko Outfitsecondbrand, untuk mempermudah pembeli dan pemilik toko dalam bertansaksi, mengelola data, pengecekan data, serta menambahkan beberapa fitur yang tidak ada pada penelitian sebelum nya seperti fitur notifikasi, fitur ongkos kirim, dan

dapat terhubung ke whatsapp admin sebagai usaha meningkatkan penjualan. Dalam proses perancangan toko *online* akan menerapkan *framework flutter*, dengan tujuan agar mempermudah serta mempercepat proses pembuatan aplikasi *mobile* toko *online*.

2. METODE PENELITIAN

Bentuk penelitian yang digunakan dalam penelitian ini adalah studi kasus. Studi kasus merupakan bentuk penelitian yang bertujuan untuk mencari secara intensif dari latar belakang dan keadaan atau kejadian sekarang terhadap toko Outfitsecondbrand yang dijadikan untuk sebuah kasus dengan menggunakan cara-cara yang sistematis yang dilakukan dengan melihat, mengamati apa yang akan dibutuhkan oleh objek penelitian.

Metode yang digunakan dalam penelitian ini adalah *design science research* (DSR). DSR juga merupakan kerangka prosedur yang digunakan untuk mempermudah penelitian di bidang teknologi informasi yang digunakan sebagai proses pemahaman serta mengulas untuk mengenali dan mengevaluasi hasil penelitian. Tahapan-tahapan yang dilakukan dalam penelitian *design science research methodology* ialah *problem identification and motivation, objective of the solution, design and development, demonstration, evaluation* dan *communication* [12].

Metode pengumpulan data merupakan bagian yang sangat penting disebuah penelitian yang dilakukan. Dengan demikian pengumpulan data harus dilakukan dengan teknik yang dapat memastikan data yang diperoleh sesuai dan akurat serta berasal dari sumber yang benar dan dapat dipercaya. Jenis data yang digunakan dalam penelitian ini ialah data primer dan data sekunder. Data primer merupakan data yang diperoleh secara langsung dari objek yang diteliti. Data primer diperoleh dengan cara wawancara dan observasi, wawancara dilakukan langsung kepada pemilik toko Outfitsecondbrand untuk memenuhi kebutuhan penelitian sedangkan observasi dilakukan dengan cara mengamati secara langsung proses proses bisnis pada toko Outfitsecondbrand. Data sekunder merupakan data yang berhubungan secara langsung dengan penelitian yang dilaksanakan data yang diperoleh merupakan data pendapatan, data penjualan, dan barang yang paling laku. Data lain yang diperlukan diperoleh melalui pencarian di internet mengenai artikel-artikel, jurnal, dan hasil dari penelitian sebelumnya yang dapat digunakan oleh peneliti sebagai bahan perbandingan dengan penelitian yang dilakukan.

Dalam perancangan pada penelitian ini penulis menggunakan metode *extrem programing*. *Extreme programing* merupakan sebuah model pengembangan prenakat lunak yang menyederhanakan tahapan-tahapan dalam proses pengembangan. *Extrem programing* tidak hanya berfokus pada penulisan baris kode namun meliputi keseluruhan area pengembangan. Tujuan dari metode pengembangan ini ialah meminimalisir biaya yang akan diperlukan jika terdapat perubahan dalam proses pengembangan. Tahapan-tahapan dalam metode pengembangan *extreme programing* ialah *planing, design, coding, testing* [13].

Pada penelitian ini untuk mempermudah pemahaman dalam perancangan digunakan alat bantu yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan Pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari system perangkat lunak [14].

Dalam perancangan suatu aplikasi *mobile* toko *online* untuk menjadinkannya siap digunakan dan sesuai dengan apa yang diharapkan, maka diperlukan suatu pengujian. Dalam penelitian ini metode pengujian yang digunakan ialah *black-box testing* dengan teknik *equivalence partitioning*. Metode *black-box testing* merupakan strategi pengujian yang memfokuskan pada fungsional dan spesifikasi suatu perangkat lunak. Sedangkan teknik *equivalence partitioning* dalam metode *black-box testing* merupakan teknik yang melakukan pengujian berdasarkan inputan setiap *form* yang terdapat pada sistem [15]. Dengan penerapan teknik *equivalence partitioning* maka akan didapati *test case* yang lebih spesifik dan dapat mengenali cacat sistem berdasarkan kasus uji yang sedang diuji.

3. HASIL DAN PEMBAHASAN

Tahapan awal pada metode penelitian yang digunakan ialah melakukan identifikasi masalah yang terjadi, hal ini dilakukan dengan melakukan pengumpulan data dengan metode wawancara dan observasi ke toko Outfitsecondbrand. Hasil yang didapatkan dari wawancara dan observasi ialah toko Outfitsecondbrand memerlukan sistem yang dapat melakukan pencatatan secara otomatis, dapat melakukan pengelolaan data barang, melakukan pemesanan secara *online*, pencarian produk, menampilkan informasi barang secara lengkap, dapat mengirimkan notifikasi.

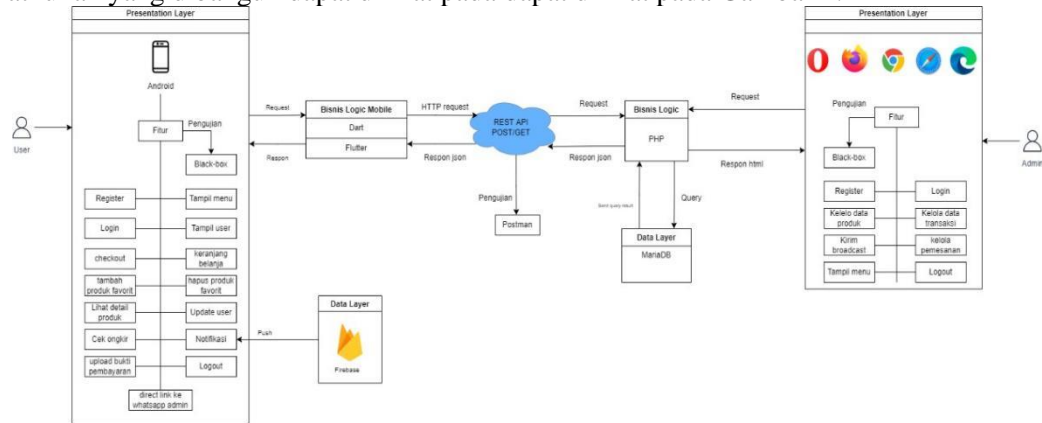
Tahap selanjutnya dilakukan identifikasi objek dari solusi permasalahan. Hasil yang didapati dari proses identifikasi ialah sistem akan dibagi menjadi dua *actor* yaitu admin dan pembeli. Admin dapat melakukan pengecekan setiap transaksi yang dilakukan, melakukan pengelolaan barang, mengecek status pembayaran, mengirimkan notifikasi. Pembeli dapat melakukan transaksi dimanapun, melakukan pencarian barang, dan melihat informasi barang secara lengkap.

Tahapan selanjutnya merupakan tahapan perencanaan dan pengembangan. Pada tahapan ini dimulai proses pengembangan aplikasi mobile dengan metode *extrem programming*. Tahapan pertama dari metode *extrem programming* ialah perencanaan, dilakukan dengan tujuan untuk memperoleh informasi terkait kebutuhan sistem yang akan dikembangkan. Berdasarkan hasil wawancara dan observasi yang dilakukan terkait kebutuhan sistem yang diperlukan oleh setiap pihak yang terlibat dalam sistem ini. Diperoleh kebutuhan sistem yang dijabarkan pada Tabel 1.

Tabel 1 Kebutuhan fungsional

No.	Pengguna	Kebutuhan Sistem	Fungsional Sistem
1.	Admin	Kelola data transaksi	Dapat melakukan pencatatan setiap transaksi yang dilakukan, merangkap tiap transaksi, mencari data-data transaksi, dan dapat menambahkan transaksi di dalam web admin jika transaksi yang dilakukan di luar aplikasi
2.	Admin	Kelola data produk	Dapat mengelola produk, seperti menambah, mengupdate, menghapus, dan pencarian produk.
3.	Admin	Mengirimkan notifikasi	Dapat mengirim notifikasi berupa promosi, penambahan produk baru, dan update status pemesanan.
4.	Pembeli	Melihat detail produk	Dapat menampilkan detail informasi produk, pada saat pembeli memilih produk.
5.	Pembeli	Melakukan pembelian secara online	Dapat memproses pembelian serta mencatat informasi pembelian, pengiriman, dan pembayaran, secara otomatis.
6.	Pembeli	Menambahkan produk ke dalam keranjang	Pembeli dapat menambahkan produk-produk yang akan di beli ke dalam keranjang sebelum melakukan checkout.
7.	Pembeli	Mengkonfirmasi pembayaran	Dapat mengkonfirmasi pembayaran setelah melakukan checkout melalui whatsapp admin atau langsung me upload bukti pembayaran melalui aplikasi.

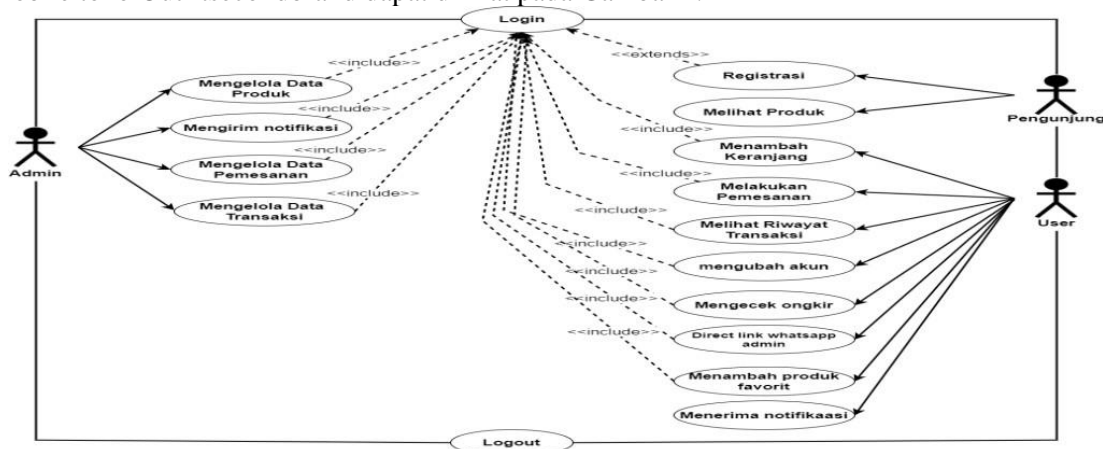
Tahapan selanjutnya dari metode *extrem programming* ialah perancangan. Pada tahapan ini dilakukan perancangan yang meliputi penempatan elemen-elemen interaksi serta pemodelan sebuah sistem berdasarkan hasil analisa kebutuhan sistem yang didapatkan. Berikut arsitektur perangkat lunak yang dibangun dapat dilihat pada dapat dilihat pada Gambar 1.



Gambar 1 Arsitektur Sistem Toko Online Outfitsecondbrand

Pada Gambar 1 merupakan arsitektur pada aplikasi toko *online* yang dibangun. Arsitektur ini menggambarkan bagaimana mekanisme interaksi *mobile user* dan web admin. Dimulai dari user mengakses *presentation layer* untuk menjalankan semua fitur maka *presentation layer* akan mengirim *request* ke *bisnis logic mobile* untuk menjalankan semua fitur yang ada. Apabila *user* akan mengirim atau menampilkan data dari database yang juga di gunakan admin maka akan melakukan permintaan atau *request* ke server, data tersebut akan dikirim ke *web server* yang dibangun. *REST API* akan melakukan proses permintaan ke database yang kemudian hasil dari proses permintaan atau request akan dikembalikan lagi ke pengguna dalam bentuk *JSON* dan diuraikan oleh aplikasi sehingga data hasil proses bisa digunakan dan ditampilkan. Sedangkan admin untuk mengakses *website* melalui *presentation layer*, untuk menjalankan semua fitur yang di sediakan maka *presentation layer* akan mengirim *request* ke *bisnis logic*, untuk menampilkan atau mengirim data ke database yang juga terhubung ke *mobile user* maka akan melakukan permintaan *query* untuk memproses data pada database. Hasil dari *query* yang dilakukan berupa *result send* data berdasarkan *request* admin, setelah itu hasil *query* akan dikirimkan kembali ke *bisnis logic* tersebut ke *presentation layer* berupa dokumen *HTML*.

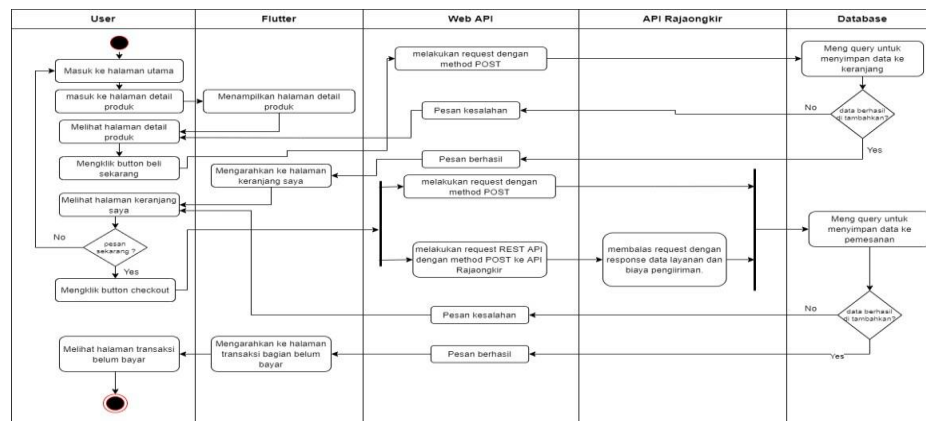
Diagram *unified modeling language* (UML) yang digunakan dalam pemodelan ialah *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*. *Use case diagram* aplikasi mobile toko Outfitsecondbrand dapat dilihat pada Gambar 2.



Gambar 2 Use Case Diagram Aplikasi Mobile Toko Outfitsecondbrand Berdasarkan

Gambar 2 pada aplikasi *mobile* ini di dalam diagram terdapat tiga aktor yang berperan yaitu *admin*, *user*, dan pengunjung. Pada *use case diagram* *admin* harus melakukan *login* terlebih dahulu untuk dapat mengelola data produk, data pemesanan, dan data transaksi. Aktor *user* harus melakukan *login* untuk dapat menambah keranjang, melakukan pemesanan, melihat riwayat transaksi, mengubah akun, menambah produk favorit, dan melakukan cek ongkir. Sedangkan aktor pengunjung hanya dapat melakukan melihat produk dan registrasi.

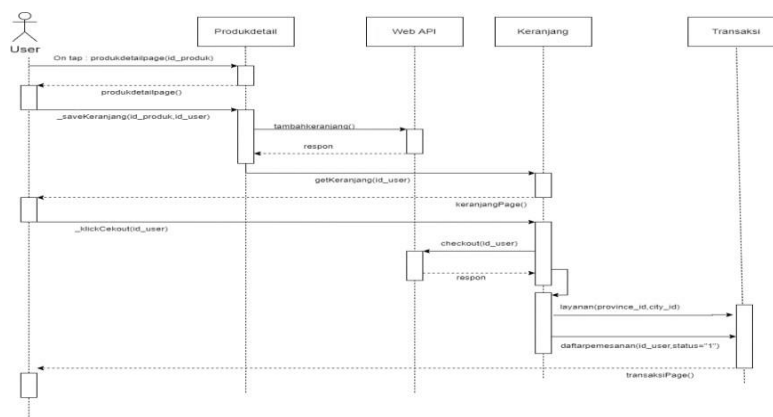
Activity diagram menggambarkan sebuah alur aktivitas dalam sebuah sistem yang dirancang, bagaimana masing-masing alur pada kondisi awal, keputusan yang mungkin terjadi, dan bagaimana alur bisa berakhir. Hasil *activity diagram* yang telah dirancang dapat dilihat pada Gambar 3.



Gambar 3 Activity Diagram Membuat Pesanan

Pada Gambar 3 proses dimulai pada saat *user* masuk ke halaman detail produk kemudian *user* meng-klik tombol beli sekarang, *user* akan diarahkan ke halaman keranjang saya, pada halaman keranjang saya *user* dapat mengubah jumlah produk dan juga menghapus produk dari list keranjang. *User* meng-klik tombol checkout maka ada dua proses yang terjadi secara bersamaan yaitu merequest ke *API* Rajaongkir dengan mengirim id provinsi dan id kota yang ada pada data *user* untuk mendapatkan layanan dan biaya pengiriman, kemudian data tersebut akan disimpan ke database. Jika data gagal di simpan maka sistem akan menampilkan pesan kesalahan, jika data berhasil di simpan maka sistem menampilkan pesan berhasil kemudian *user* akan diarahkan ke halaman transaksi bagian belum dibayar.

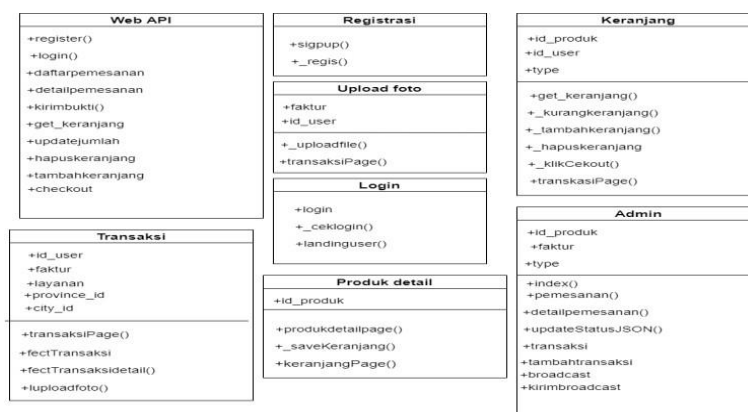
Diagram sequence merupakan salah satu *diagram* yang menjelaskan bagaimana suatu operasi itu dilakukan, pesan apa yang dikirim dan kapan pelaksanaannya. Hasil *sequence diagram* yang telah dirancang dapat dilihat pada Gambar 4.



Gambar 4 Sequence Diagram Pemesanan

Pada Gambar 4 proses ini di mulai pada saat *user* masuk ke menu produk detail, *class* produk detail memanggil fungsi *produkdetailpage()*, untuk menambah pemesanan *class* produkdetail memanggil fungsi *_saveKeranjang(id_produk,id_user)*, untuk mengirim data ke database *class Web API* memanggil fungsi *tambahkeranjang*, setelah berhasil di tambahkan *class* produkdetail memanggil fungsi *keranjangPage()* untuk menampilkan halaman keranjang, untuk melanjutkan pemesanan *class* keranjangPage memanggil fungsi *_klikCekout(id_user)* untuk meminta *request* ke *class Web API*, setelah *request* di terima *class WebAPI* memanggil fungsi *checkout(id_user)* kemudian hasil respon di teruskan ke kelas keranjang untuk menjalankan dua fungsi ke *class* transaksi yaitu *layanan(province_id,city_id)* untuk mendapatkan layanan dan biaya pengiriman, fungsi *daftarpemesanan(id_user,status="1")* untuk menampilkan data pemesanan yang belum di bayar,kemudian *class* transaksi memanggil dan fungsi *transaksiPage()* untuk menampilkan halaman transaksi.

Class diagram adalah salah satu jenis diagram berbentuk struktur pada model UML. Diagram ini menggambarkan struktur, atribut, kelas, hubungan dan metode dengan sangat jelas dari setiap objeknya. Hasil *class diagram* yang telah dirancang dapat dilihat pada Gambar 5.



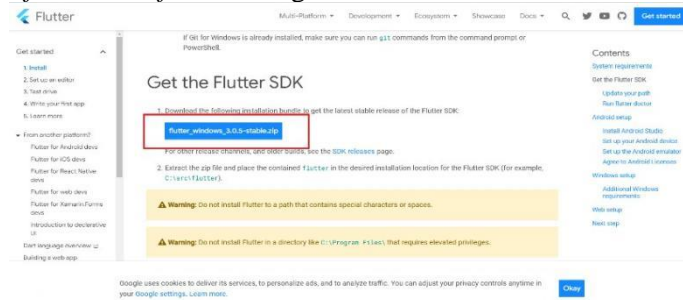
Gambar 5 Class Diagram

Berdasarkan Gambar 5 terdapat *class diagram* yang menghubungkan suatu *class* dengan *class* yang lainnya serta memperlihatkan hubungan antara relasi-relasi.

Tahapan selanjutnya dari metode *extreme programming* yaitu pengkodean. Pengkodean merupakan tahap penerapan hasil rancangan yang telah ada dengan melakukan penulisan baris kode program web menggunakan bahasa pemrograman *HTML, PHP, CSS*, sedangkan untuk kode program aplikasi *mobile* menggunakan bahasa pemrograman *dart* dan menggunakan *framework*

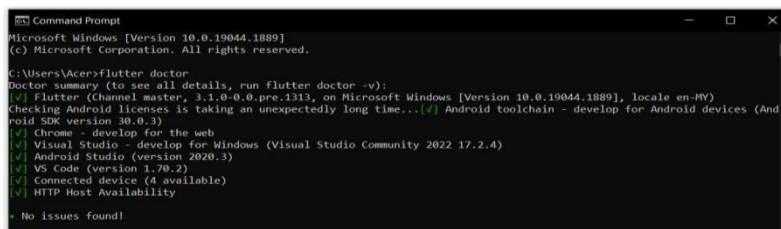
Pemanfaatan Aplikasi Mobile Menggunakan Framework Flutter Pada Toko Outfitsecondbrand

flutter, digunakan juga *MySQL* sebagai basis data. Ada beberapa konfigurasi yang dilakukan untuk menggunakan *framework flutter* sebagai berikut :



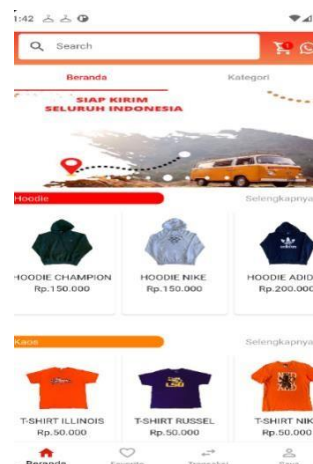
Gambar 6 Install Flutter SDK

Pada Gambar 6 ditampilkan *website* resmi *flutter* kemudian *download Flutter SDK*, setelah di *download* ekstrak *file zip* dan tempatkan *flutter* yang ada di lokasi instalasi yang diinginkan untuk *Flutter SDK* (misalnya, *C:\src\flutter*). Setelah itu jalankan *flutter doctor* pastikan semua seperti pada gambar 7 di bawah ini



Gambar 7 Flutter Doctor

Setelah proses pengkodean selesai maka didapati hasil sebuah aplikasi *mobile* toko Outfitsecondbrand yang sudah dapat digunakan dalam memenuhi kebutuhan yang telah ditentukan sebelumnya. Hasil tampilan dari aplikasi mobile toko Outfitsecondbrand ialah sebagai berikut:



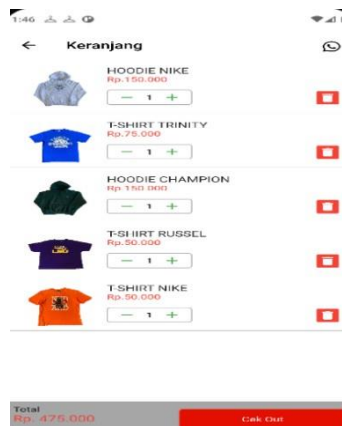
Gambar 8 Tampilan Beranda

Berdasarkan Gambar 8 merupakan tampilan halaman utama dari aplikasi toko *online* yang di dalamnya terdapat produk-produk yang dijual dan kategorinya, dimana halaman ini dapat diakses oleh *user*, dan pengunjung.



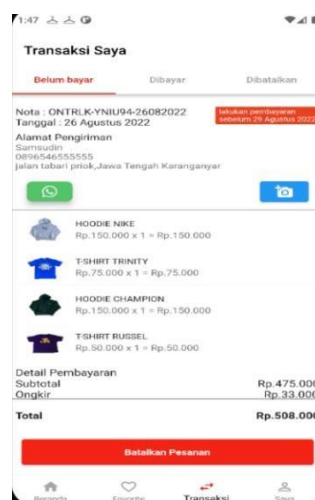
Gambar 9 Halaman Detail Barang

Pada Gambar 9 merupakan tampilan halaman detail barang yang menampilkan informasi barang berupa gambar barang, nama barang, harga barang, dan deskripsi barang, serta terdapat juga tombol untuk menambahkan produk ke keranjang dan favorit.



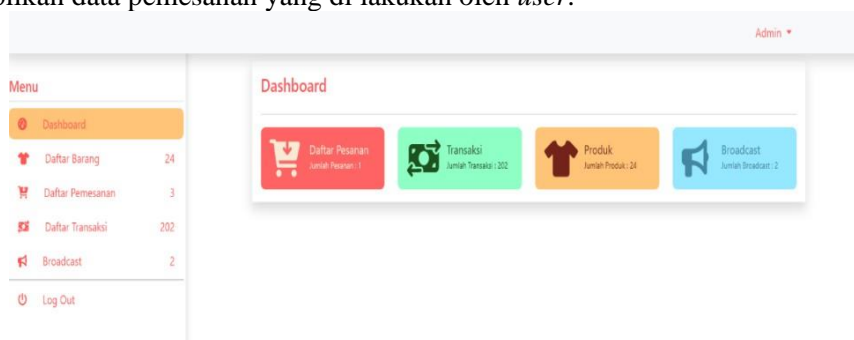
Gambar 10 Halaman Kernajang

Pada Gambar 10 merupakan halaman keranjang saya digunakan untuk menampilkan data yang di keranjang *user*.



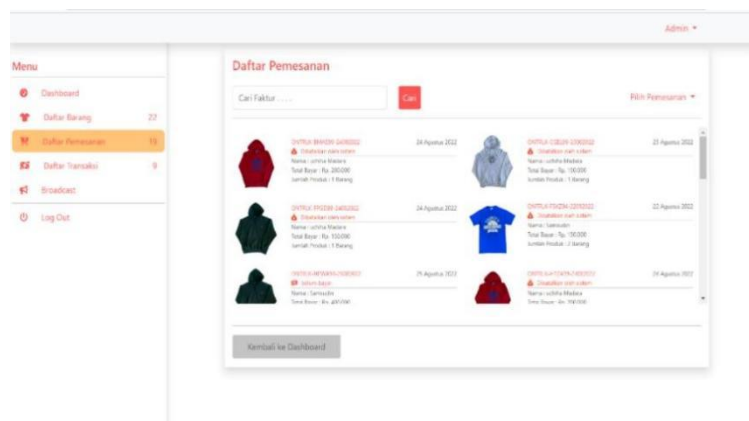
Gambar 11 Halaman Detail Pemesanan

Pada Gambar 11 merupakan tampilan halaman transaksi belum bayar, di mana halaman ini menampilkan data pemesanan yang di lakukan oleh *user*.



Gambar 12 Halaman Utama Admin

Pada gambar 12 merupakan tampilan halaman utama setelah admin melakukan *login* pada *website* admin.



Gambar 13 Halaman Kelola Pesanan

Pada gambar 13 merupakan tampilan mengelola data pesanan pada *website* admin, halaman ini hanya dapat diakses oleh admin. Pada halaman ini admin dapat melakukan pengelolaan pesanan seperti mengecek status pembayaran dan mengkonfirmasi pembayaran.

Setelah aplikasi *mobile* toko *online* berhasil dibuat maka akan dilanjutkan pada tahap pengujian. Tahapan pengujian dilakukan agar dapat memastikan semua fungsi dari aplikasi dapat berjalan sesuai kebutuhan. Metode pengujian yang digunakan pada penelitian ini adalah *black box* dengan teknik pengujian *equivalent partition*. Hasil pengujian dapat dilihat pada tabel berikut:

Tabel 2 Pengujian pemesanan barang

ID	Tahapan Pengujian	Hasil yang Diharapkan	Hasil Pengujian
----	-------------------	-----------------------	-----------------

A01	<ol style="list-style-type: none"> 1. Mengakses halaman produk detail 2. Mengklik tombol beli sekarang 3. <i>Login == true</i> 	Sistem menerima data, dan berhasil menambahkan produk ke keranjang lalu user di arahkan ke halaman keranjang saya	Sistem menerima data, dan berhasil menambahkan produk ke keranjang lalu user di arahkan ke halaman keranjang saya
A02	<ol style="list-style-type: none"> 1. Mengakses halaman produk detail 2. Mengklik tombol beli sekarang 3. <i>Login == false</i> 	Sistem menolak <i>request</i> , dan menampilkan pesan anda belum <i>login</i> .	Sistem menolak <i>request</i> , dan menampilkan pesan anda belum <i>login</i> .
A03	<ol style="list-style-type: none"> 1. Mengakses halaman keranjang 2. Klik tombol <i>checkout</i> 	Sistem menerima <i>request</i> , data produk di tambahkan ke pemesanan lalu <i>user</i> akan di arahkan ke halaman transaksi belum bayar.	Sistem menerima <i>request</i> , data produk di tambahkan ke pemesanan lalu <i>user</i> akan di arahkan ke halaman transaksi belum bayar.
A04	<ol style="list-style-type: none"> 1. Masuk ke halaman transaksi saya bagian belum bayar 2. Pilih Pemesanan untuk melihat detail pemesanan 3. Klik tombol batalkan pemesanan 	Sistem menerima <i>request</i> dan menghapus data pemesanan lalu muncul pesan “berhasil membatalkan pesanan”.	Sistem menerima <i>request</i> dan menghapus data pemesanan lalu muncul pesan “berhasil membatalkan pesanan”.
A05	<ol style="list-style-type: none"> 1. Masuk ke halaman transaksi saya bagian belum bayar 2. Pilih Pemesanan untuk melihat detail pemesanan 3. Klik tombol upload bukti pembayaran 4. Input foto : bukti.jpg 5. Klik kirim bukti pembayaran 	Sistem menerima request untuk menampilkan halaman detail transaksi, foto bukti berhasil di upload muncul pesan “pembayaran berhasil, menunggu konfirmasi admin” dan kembali ke halaman transaksi.	Sistem menerima request untuk menampilkan halaman detail transaksi, foto bukti berhasil di upload muncul pesan “pembayaran berhasil, menunggu konfirmasi admin” dan kembali ke halaman transaksi.

Pada Tabel 2 dilakukan skenario pengujian pemesanan barang, untuk memastikan bahwa sistem dapat memberikan respon yang sesuai. Pengujian yang dilakukan mendapati hasil bahwa sistem dapat melakukan pemesanan dengan inputan yang benar, dan sistem juga dapat menolak jika inputan tidak benar.

Tabel 3 Pengujian konfirmasi pembayaran

ID	Tahapan Pengujian	Hasil yang Diharapkan	Hasil Pengujian
B01	<ol style="list-style-type: none"> 1. Mengakses halaman pemesanan 2. Lihat detail pemesanan 3. Klik tombol <i>update</i> status pemesanan 4. Foto bukti pembayara == <i>true</i> 	Sistem menerima <i>request</i> , dan berhasil mengubah status pemesanan, <i>user</i> mendapatkan notifikasi.	Sistem menerima <i>request</i> , dan berhasil mengubah status pemesanan, <i>user</i> mendapatkan notifikasi.
B02	<ol style="list-style-type: none"> 1. Mengakses halaman pemesanan 2. Lihat detail pemesanan 3. Klik tombol <i>update</i> status pemesanan 4. Foto bukti pembayara == <i>false</i> 	Sistem menerima <i>request</i> , dan gagal mengubah status pemesanan, muncul pesan wajib ada foto bukti pembayaran.	Sistem menerima <i>request</i> , dan gagal mengubah status pemesanan, muncul pesan wajib ada foto bukti pembayaran.

Pada Tabel 3 dilakukan skenario pengujian konfirmasi pembayaran, untuk memastikan bahwa sistem dapat memberikan respon yang sesuai. Pengujian yang dilakukan mendapati hasil bahwa sistem dapat dapat melakukan *update* pesanan apabila pesanan tersebut memiliki bukti foto pembayaran dan sistem juga dapat menolak jika foto bukti pembayaran tidak ada.

Tabel 4 Hasil Pengujian

NO	ID	Kesimpulan
1	A01	Berhasil
2	A02	Berhasil
3	A03	Berhasil
4	A04	Berhasil
5	A05	Berhasil
6	B01	Berhasil
7	B02	Berhasil

Pada Tabel 4 merupakan kesimpulan hasil dari setiap *case test*. Dari tiap proses pengujian yang dilakukan didapati kesimpulan bahwa setiap *case test* berhasil dikerjakan oleh sistem dengan benar dan aplikasi mobile toko *online* berhasil berjalan sesuai dengan yang diharapkan.

Proses pengembangan aplikasi *mobile* telah selesai dilakukan, selanjutnya akan kembali pada tahapan *design science research* yaitu tahapan demonstrasi. Tahap ini penulis melakukan *hosting* terhadap *website* admin yang telah dibangun sebagai *web server* agar aplikasi *mobile* toko *online* dapat mengakses data secara *online* dan melakukan simulasi dari penggunaan *website* admin dan aplikasi *mobile* toko *online user* yang dapat mempermudah pihak toko dalam mengoperasikan penjualan produk. dapat dilihat pada Gambar 14.



Gambar 14 Kegiatan Simulasi

Setelah tahapan demonstrasi dilakukan didapati sebuah evaluasi dari pemilik toko, dimana diperlukan penambahan dari sisi *website* admin di segi fitur penambahan produk di mana di minta pada saat penambahan produk notifikasi akan otomatis di kirim berdasarkan data produk baru yang di tambahkan, dengan tujuan agar pemilik toko tidak perlu melakukan pekerjaan yang berulang-ulang setelah menambahkan produk. Setelah mendapati apa yang perlu tambahkan maka dilakukan kembali kepada proses penulisan baris kode. Adapun hasil penambahan dan baris kode yang ditambahkan dapat dilihat pada gambar 15.

```
$.ajax({
  url: base_url + "admin/addBarangJSON",
  type: "post",
  data: new_form,
  contentType: false,
  processData: false,
  dataType: "json",
  success: function (data) {
    console.log(data);
    var array = [];

    for (let i = 0; i < data.user.length; i++) {
      array.push(data.user[i].token);
    }

    console.log(array);

    var linkFoto = base_url + data.produk.foto_produk;
    var title = "Hai ada Produk Baru nih!!!!";
    var body = "${nama} dapat kalian dapatkan dengan harga sebesar ${harga}, kuy kejar a sebelum kehabisan!!";
    var image = linkFoto;

    notif(title, body, image, array);
    setTimeout(function() { window.location.replace(base_url + "admin/barang/"); }, 1000);
  }
});
```

Gambar 15 Kode Mengirimkan Notifikasi Bersamaan dengan Produk di Tambah

Setelah tahapan evaluasi dilakukan, selanjutnya dilakukan tahapan komunikasi. Pada tahapan ini dilakukan kembali simulasi kepada pemilik toko untuk menunjukkan hasil akhir setelah dilakukannya perubahan. Kegiatan simulasi dapat dilihat pada Gambar 16.



Gambar 16 Kegiatan Komunikasi

Tahapan komunikasi yang telah dilakukan, menghasilkan kesimpulan bahwa pengembangan dan evaluasi terhadap aplikasi *mobile* toko *online* Outfitsecondbrand memberikan sebuah hasil yang sesuai dengan apa yang diharapkan, dan secara *funksional* dapat digunakan dengan baik.

4. KESIMPULAN

Berdasarkan hasil kajian dan pembahasan dalam penelitian ini, Permasalahan yang dihadapi toko Outfitsecondbrand penjualan yang dilakukan di media sosial mengakibatkan banyak saingan bisnis, kesulitan dalam melayani calon pembeli, memberikan promosi yang kemungkinan tidak tepat sasaran, melakukan pekerjaan yang sama berulang ulang dan kesulitan dalam pendataan transaksi karena menggunakan pembukuan sehingga dapat terjadinya data transaksi yang tidak terdata dan rentan terhadap kerusakan dan kehilangan. Hasil dari penelitian ini ialah aplikasi toko *online* dengan menerapkan *framework flutter* yang *multiplatform* terdapat dua sistem yaitu *mobile* untuk *user* dan *website* untuk admin. Pada aplikasi *mobile* memiliki fitur *login*, registrasi, pemesanan, keranjang, dan notifikasi yang digunakan untuk promo-promo, terkonfirmasi nya pembayaran, dan produk terbaru. Sedangkan *website* memiliki fitur mengelola data produk, data pemesanan, data transaksi, dan dapat mengirim notifikasi. dengan toko *online* dapat menjangkau pemasaran yang lebih luas, pengelolaan data toko lebih mudah, serta memungkinkan untuk bertransaksi tanpa batasan waktu dan lokasi, sehingga dengan adanya aplikasi toko *online* sendiri permasalahan yang ada pada toko Outfitsecondbrand dapat teratasi.

5. SARAN

Aplikasi *mobile* yang dibangun sudah cukup memberikan solusi mengenai permasalahan yang terjadi, namun aplikasi *mobile* yang dibangun masih jauh terdapat belum sangat sempurna, karna itu terdapat saran-saran untuk proses pengembangan aplikasi selanjutnya, saran tersebut ialah menambahkan fitur *chat* dengan pihak toko Outfitsecondbrand sehingga untuk *chatting* tidak perlu lagi pindah ke whatsapp dan hanya fokus pada aplikasi *mobile* toko *online*, dan menyediakan fitur *review* berupa komentar langsung pada aplikasi *mobile* menggunakan *text*, foto, ataupun video.

DAFTAR PUSTAKA

- [1] D. Pangestu dan T. Wijaya, "Penerapan Web Service Untuk E-Commerce Makro Komputer Sintang," *Jurnal ENTER*, vol. 2, pp. 175-185, 2019.
 - [2] V. Gow, S. Kosasi, T. Wijaya, David dan R. Laipaka, "Penerapan Teknologi Restful Web Service Aplikasi Mobile Jasa Make Up," *InfoSys Journal*, vol. 6, no. 2, pp. 128-140, 2022.
 - [3] Gat, "Pengembangan Sistem Reservasi Hotel Berbasis Mobile dengan Framework JQuery Mobile," *Citec Journal*, vol. 3, pp. 27-36, 2016.
 - [4] Gat, "Pemanfaatan JQuery Mobile Untuk Merancang Sistem Informasi Wisata Kuliner (Studi Kasus:Kota Pontianak)," dalam *Konferensi Nasional Sistem & Informatika*, STMIK STIKOM Bali, 2015.
 - [5] S. Kosasi, "Perancangan Sistem E-Commerce Untuk Memperluas Pasar Produk Oleh- Oleh Khas Pontianak," dalam *SNASTIA*, Universitas Surabaya, 2015.
 - [6] K. Seputra dan G. Sandiasa, "Rancang Bangun Sistem Informasi Satgas Gotong Royong (Si Garong) Desa Adat Berbasis Mobile," *Jurnal Nasional Pendidikan Teknik Informatika*, vol. 9, no. 3, pp. 339-352, 2020.
 - [7] T. Thanh, *Flutter Native Performance and Ex- pressive UI/UX*, Finlandia:Metropolia University of Applied Sciences, 2020.
 - [8] L. Suhery dan Dilson, "Model dan Prototype Aplikasi Penerimaan Mahasiswa Baru (PMB) Berbasis Mobile Menggunakan Framework Flutter," *Jurnal SIMTIKA*, vol. 4, no. 2, pp. 29-39, 2021.
 - [9] M. Hendriawan, T. Budiman, V. Yasin dan A. Rini, "Pengembangan Aplikasi E- commerce di PT.Putra Sumber Abadi Menggunakan Flutter," *Journal of Information System, Informatics and Computing Website/URL:*, vol. 5, no. 1, pp. 69-88, 2021.
 - [10] Herlangga dan L. Ni, "Rancang Bangun Aplikasi E-commerce Berbasis Android dan Website Menggunakan Framework Flutter dan Laravel," *SmartEDU*, vol. 1, no. 1, pp. 31- 44, 2020.
 - [11] F. Krisnada dan R. Tanone, "Aplikasi Penjualan Tiket Kelas Pelatihan Berbasis Mobile menggunakan Flutter," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 5, no. 3, pp. 281-295, 2019.
 - [12] J. Brocke, M. Weber dan T. Grisold, *Design Science Research Cases*, Switzerland: Springer Nature Switzerland AG, 2021.
 - [13] R. Pressman dan B. Maxim, *Software Engineering A Practitioner's Approach*, New York: McGraw-Hill Education, 2020.
 - [14] A. Rosa dan M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*, Bandung: Informatika, 2018.
 - [15] T. Hidayat dan M. Muttaqin, "Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *JUTIS*, vol. 6, no. 1, pp. 25-19, 2018.
-