

Pengembangan Aplikasi E-Commerce berbasis Android sebagai Sarana Jual Beli Barang Bekas menggunakan Metode Waterfall

Anastasia Kezia Beatrix Mayestika^{1*}, Eko Sedyono²
^{1,2}Jurusan Teknik Informatika, FTI UKSW, Salatiga, Indonesia
e-mail: ¹672019208@student.uksw.edu, ²eko@uksw.edu

Abstract

People often find it difficult to cope with the accumulation of household waste that includes furniture, clothes and other items. This accumulation of household waste does not only affect one house but also affects neighboring houses. Piles of unused items continue to accumulate in homes and other places, resulting in problems caused by inconvenience due to the large number of new items. With these problems, an application for buying and selling used goods was developed for people who want to sell or buy used goods. This research uses the Waterfall method as a software development method. This application is made by integrating the responsibilities of sellers and buyers simultaneously, resulting in one account that can buy and sell at the same time. With this application, it is hoped that the community can be helped in finding used products or goods that are still suitable for use.

Keywords: Android Application, E-Commerce, MVVM, Waterfall Method

Abstrak

Masyarakat sering kali bergumul dengan penumpukan sampah rumah tangga yang meliputi perabotan, pakaian, dan barang-barang lainnya. Penumpukan sampah rumah tangga ini tidak hanya berdampak pada satu rumah saja, tetapi juga memberikan efek bagi rumah-rumah di lingkungan sekitar. Tumpukan barang yang sudah tidak terpakai terus menumpuk di dalam rumah dan di tempat lain, sehingga terdapat masalah yang ditimbulkan, yaitu ketidaknyamanan akibat banyaknya benda yang tidak terpakai. Dengan adanya permasalahan tersebut, dikembangkanlah aplikasi jual beli barang bekas yang ditujukan untuk masyarakat yang ingin menjual atau membeli barang bekas. Penelitian ini menggunakan metode Waterfall sebagai metode pengembangan perangkat lunak. Aplikasi ini dibuat dengan mengintegrasikan tanggung jawab penjual dan pembeli secara bersamaan, sehingga dapat menghasilkan satu akun yang dapat membeli dan menjual dalam waktu yang bersamaan. Dengan adanya aplikasi ini, diharapkan masyarakat dapat terbantu dalam menemukan produk atau barang bekas yang masih layak pakai.

Kata kunci: Aplikasi Android, E-Commerce, MVVM, Metode Waterfall

1. PENDAHULUAN

Masyarakat kerap bergumul dengan penumpukkan sampah rumah tangga, termasuk perabotan, pakaian, dan barang-barang lainnya. Penumpukkan sampah rumah tangga ini tidak hanya menjadi masalah pada satu rumah saja tetapi juga memberikan efek bagi rumah-rumah di lingkungan sekitar. Tumpukkan barang yang sudah tidak terpakai terus menumpuk di dalam rumah dan ditempat lain, masalah yang ditimbulkan yaitu ketidaknyamanan akibat banyaknya barang yang tidak terpakai. Ada banyak barang yang masih layak pakai, namun masyarakat lebih memilih untuk tidak memanfaatkannya. Pasar atau tempat jual beli barang bekas terbentuk bagi mereka yang ingin memanfaatkannya. Tentu saja, pasar barang bekas ini ada di mana-mana, di setiap daerah di Indonesia atau di belahan dunia. Pasar ini dikenal dengan sebutan pasar loak.



Pasar loak merupakan pasar yang menjual berbagai jenis barang bekas. Di pasar loak kita ditawarkan berbagai jenis barang yang dijual oleh pedagang kecil hingga pedagang besar. Pedagang kecil hingga pedagang besar ini ditentukan oleh sejumlah faktor, seperti ukuran produk yang dijual, lokasi tempat berdagang, dan metode penjualan. Barang bekas yang dijual pun barang yang masih layak pakai dan dijual dalam keadaan yang baik. Karena penjualan barang bekas merupakan industri yang dapat menghasilkan keuntungan yang signifikan, penjualan barang bekas memperoleh popularitas selama lebih dari satu dekade[1].

Dengan adanya pasar loak, masyarakat bisa mengatasi kebutuhan rumah tangga yang semakin mahal ini. Sehingga masyarakat kelas menengah ke bawah mampu membeli kebutuhan sandang dan barang-barang rumah tangga lainnya. Di lain sisi, orang kalangan menengah ke atas kesulitan dalam menyalurkan barang kepunyaannya yang sudah tidak terpakai tetapi masih layak. Untuk itu diperlukan suatu wadah di mana barang-barang bekas tersebut dapat dijual dengan harga yang terjangkau dan pastinya dapat dengan mudah dijangkau oleh masyarakat luas. Keadaan tersebut sangat memikat kalangan menengah ke bawah atau menengah ke atas karena banyak barang bekas yang masih dalam kondisi prima dengan harga terjangkau.

Saat ini semua lapisan masyarakat sudah menggunakan smartphone untuk mendukung kegiatan sehari-hari. Hanya melalui smartphone orang bisa melakukan pemesanan, transaksi online, dan memesan ojek. Meningkatnya penggunaan smartphone dipengaruhi oleh perkembangan internet yang cepat. Berdasarkan temuan terbaru survei Asosiasi Penyelenggara Jasa Internet (APJII), sekitar 210 juta orang Indonesia menggunakan internet pada tahun 2022 (CNN, 2022), dengan sebagian besar menggunakan smartphone. Oleh karena itu, dibuatlah aplikasi e-commerce berbasis mobile yang dikhususkan untuk jual beli barang bekas. Aplikasi tersebut dapat diakses melalui smartphone. Karena smartphone telah menjadi gaya hidup sebagian besar masyarakat. Smartphone secara bertahap menggantikan komputer (desktop) untuk tugas sehari-hari, termasuk kegiatan bisnis. Menurut data dari StatCounter GlobalStats penjualan sistem operasi Android secara proporsional melampaui semua sistem operasi lain dengan persentase 43,06 %, diikuti oleh Windows dengan 29,39 %, dan iOS dengan 17,35 % untuk periode September 2021 hingga September 2022.

2. METODOLOGI PENELITIAN

2.1. Tinjauan Pustaka

Berdasarkan penelitian dengan judul Rancang Bangun Aplikasi E-Commerce Smartphone Second dengan Metode Waterfall yang dilakukan oleh Wahyu Eko Susanto, dkk mengenai perancangan aplikasi jual beli ponsel bekas berbasis android untuk menyelesaikan permasalahan ketika konsumen ingin membeli ponsel bekas tanpa harus bertemu langsung dengan penjual. Tentu saja konsumen juga akan dibekali informasi mengenai kualitas ponsel dan juga kemudahan konsumen dalam bertransaksi tanpa batas ruang dan waktu [2]. Pada penelitian tersebut metode waterfall dipilih sebagai metode pengembangan aplikasi karena metode ini menggunakan pendekatan sekuensial mulai dari analisis, desain,



pengkodean, pengujian, dan tahap pendukung. Sehingga untuk melanjutkan ke tahap selanjutnya tahap sebelumnya harus diselesaikan terlebih dahulu agar tahapan selanjutnya dapat berjalan dengan baik.

Pada penelitian berikutnya yang berjudul Perancangan Aplikasi E-Commerce pada Goresan Pena Studio Berbasis Android oleh Ayu Putri Iqklima dan Sekreningsih Nita mengenai perancangan aplikasi e-commerce pada Goresan Pena Studio Berbasis Android yang bertujuan untuk menarik pelanggan dan meningkatkan omset bagi pemilik usaha dengan hadirnya kemudahan pelanggan untuk pemesanan jasa pada Goresan Tinta Studio tanpa harus membayar biaya operasional [3]. Pada proses penelitian ini juga menggunakan metode waterfall sebagai metode perancangan aplikasi. Setelah melalui tahapan pengembangan dengan metode waterfall ini, terlihat hasil pada sistem lama penggunaannya masih bersifat offline dan hanya melalui media sosial sebagai perantara. Kemudian sistem baru akan memudahkan untuk mengiklankan bisnis yang sudah ada. Pelanggan dapat dengan mudah mendapatkan akses penuh dan fasilitas online yang memadai.

Sedangkan pada acuan penelitian berikutnya yang berjudul Perancangan Aplikasi E-Commerce Berbasis Android pada UD Hoky Celluler Shop oleh Adam Surya Wijaya dan Johannes Fernandes Andry mengenai pembuatan aplikasi e-commerce untuk menjual smartpone dan aksesoris seperti kabel USB, powerbank, dan sebagainya. Tujuan dibentuknya aplikasi ini untuk memperluas jangkauan bisnis, sehingga dapat memperoleh pelanggan baru dan memuaskan pelanggan karena dengan menggunakan aplikasi ini orang-orang dapat membeli barang yang dibutuhkan secara online atau dari rumah. Selain itu agar proses bisnis UD Hoky Celluler Shop lebih efektif dan efisien karena pelayanan yang diberikan lebih cepat dari sebelumnya dan juga aplikasi ini secara otomatis mencatat laporan setiap transaksi [4]. Metode waterfall merupakan salah satu teknik pengembangan perangkat lunak yang digunakan dalam penelitian ini karena UD Hoky Celluler ingin benar-benar dari awal dalam menganalisis setiap kebutuhan yang dimulai dari identifikasi masalah, kebutuhan aplikasi, eksternal/pelanggan. Hasil akhir dari penelitian tersebut adalah terbentuknya aplikasi yang dapat berjalan dengan baik sesuai kebutuhan pengguna dan diharapkan dapat meningkatkan penjualan. Berdasarkan penelitian tersebut dapat diambil kesimpulan bahwa metode waterfall sebagai metode pengembangan aplikasi benar-benar dapat menghasilkan aplikasi yang sesuai dengan kebutuhan user. Karena dalam prosesnya dilakukan analisis kebutuhan yang sangat mendalam.

Berdasarkan penelitian terdahulu yang pernah dilakukan terkait Pengembangan Aplikasi E-Smes Berbasis Android dengan Waterfall Method Sebagai Solusi Pemasaran dan Pengelolaan Usaha UMKM oleh Tri Kurniawati, dkk membahas mengenai pengembangan aplikasi untuk membantu dalam pengelolaan usaha mitra yang dimulai dari data catatan keuangan, data produk, dan laporan keuangan yang berbentuk neraca, hingga data perhitungan laba/rugi. Aplikasi ini juga menyediakan fitur jual beli yang tergabung dengan aplikasi pertama. Produsen dapat menawarkan barang mereka yang tersedia di aplikasi kedua ini. Aplikasi ini juga dapat terhubung dengan platform e-commerce yang sudah ada



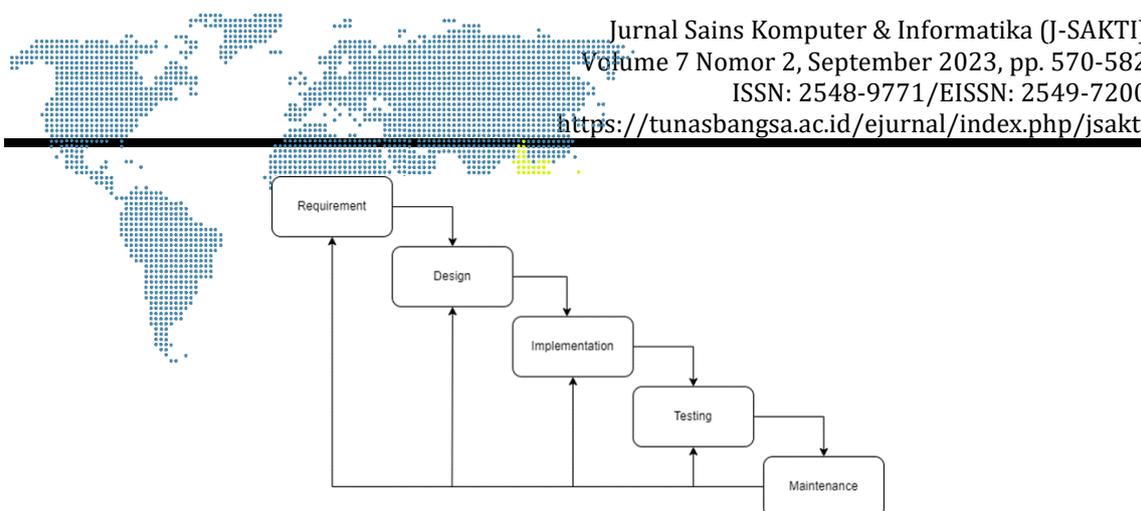
seperti Shopee dan Tokopedia untuk meningkatkan pemasaran online dan meningkatkan kepercayaan konsumen [5]. Karena tahapan proses disusun secara linier dan metodis, metode waterfall dipilih untuk pendekatan pengembangan aplikasi penelitian ini. Dimana proses awal dimulai dengan spesifikasi kebutuhan pengguna dan berkembang melalui proses perencanaan, pemodelan, pembangunan, dan penyebaran, dengan hasil akhir berupa pembuatan perangkat lunak ke dalam sistem yang terstruktur.

Pada acuan penelitian berikutnya yang berjudul Implementasi Metode Waterfall pada Aplikasi Marketplace Laundry Berbasis Android oleh Dini Silva Purnia, dkk membahas mengenai penerapan metode waterfall dalam pembuatan aplikasi laundry untuk memudahkan pelaku usaha layanan laundry serta konsumen yang bisa diakses kapanpun dan dimanapun. Apalagi bisnis laundry yang sudah menurun karena banyaknya persaingan sesama pengusaha jasa laundry. Sehingga dibuatlah perancangan sistem marketplace laundry berbasis android untuk memudahkan para pelaku usaha untuk mempromosikan jasa laundry-nya [6]. Pada penelitian ini metode waterfall digunakan sebagai metode perancangan aplikasi karena metode ini memiliki rangkaian alur kerja yang jelas dan terukur, selain itu biaya yang dikeluarkan lebih sedikit karena klien tidak dapat mencampuri urusan dari tim pengembangan aplikasi.

Berdasarkan penelitian terdahulu yang pernah dilakukan terkait implementasi metode pengembangan waterfall dalam perancangan aplikasi berbasis android. Maka penulis akan menggunakan metode pengembangan waterfall dalam mengembangkan aplikasi e-commerce berbasis android sebagai sarana jual beli barang bekas dikarenakan metode waterfall diyakini mampu menciptakan aplikasi yang mampu memenuhi semua kebutuhan pengguna. Dalam menggunakan metode ini, proses yang dibutuhkan diyakini cukup efektif untuk kondisi saat karena menghemat waktu dan sumber daya. Aplikasi ini akan dibuat dengan bahasa pemrograman kotlin dan menggunakan android studio sebagai IDE. Diharapkan penelitian ini dapat mengatasi permasalahan sampah rumah tangga, yaitu berupa barang-barang yang sudah tidak terpakai tetapi masih layak pakai, serta diharapkan memberikan kemudahan bagi penjual (pemilik barang bekas) dalam mempromosikan barangnya.

2.2. Tahapan Penelitian

Metode yang digunakan dalam proses penelitian ini adalah Waterfall yang merupakan salah satu teknik pengembangan sistem/aplikasi yang sering digunakan perangkat lunak. Metode menekankan tahapan dan juga urutan yang sistematis dan berurutan. Metode waterfall sendiri merupakan metode yang dirancang. Kelebihan dari penggunaan metode ini dalam pengembangan aplikasi/perangkat lunak adalah kualitas dari perangkat lunak yang dihasilkan akan baik karena tahapan yang dilakukan secara bertahap[7]. Tahapan penelitian metode pengembangan waterfall dalam pengembangan aplikasi berbasis android dapat dilihat pada Gambar 1.



Gambar 1. Metode Pengembangan Waterfall

Dijelaskan sebagai berikut, tahapan penelitian pada gambar 1:

a) Requirement

Pada tahap ini semua persyaratan yang akan digunakan dalam pembuatan aplikasi akan dianalisis pada tahap *requirement*. Seorang pengembang harus menyadari dan memahami kebutuhan informasi konsumen sebelum memulai pengembangan perangkat lunak. Pada tahap analisis ini dilakukan penelitian studi pustaka dan observasi secara tidak langsung untuk mendapatkan data melalui internet yang disesuaikan dengan tema penulisan.

b) Design

Dalam tahapan ini semua informasi kebutuhan yang sudah dilakukan pada tahap *requirement* akan dianalisis yang kemudian akan diimplementasikan ke dalam desain pengembangan. Tujuan dari tahapan desain ini adalah untuk membuat gambaran yang jelas tentang apa yang harus dilakukan.

c) Implementation

Data yang sudah dianalisis atau dirancang akan dituliskan menggunakan bahasa pemrograman agar dapat dimengerti oleh *user* sesuai dengan kebutuhan desain yang sudah dibuat.

d) Testing

Dalam tahap testing program yang sudah dibuat atau dikembangkan akan di uji coba untuk mengurangi kesalahan (*error*) dan menjamin bahwa keluaran (*output*) sesuai dengan yang diharapkan.

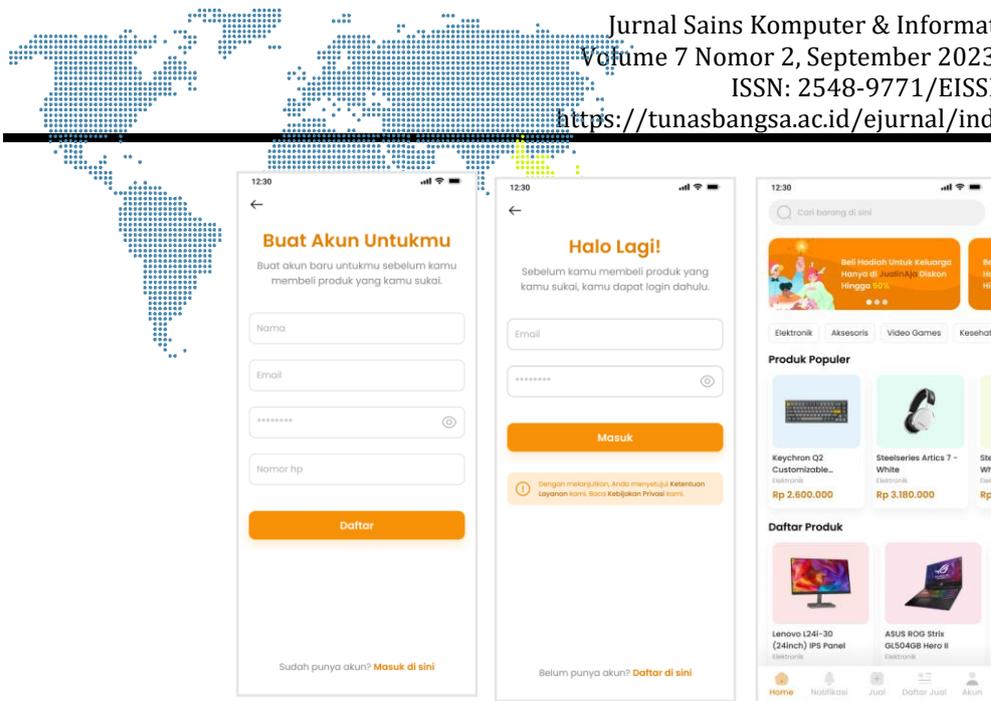
e) Maintenance

Dalam tahap *maintenance*, aplikasi yang sudah jadi akan diupayakan dipelihara secara aktif untuk mengatasi setiap kesalahan atau *bugs* yang mungkin akan ditemukan dalam program dan untuk meningkatkan pengalaman pengguna.

3. HASIL DAN PEMBAHASAN

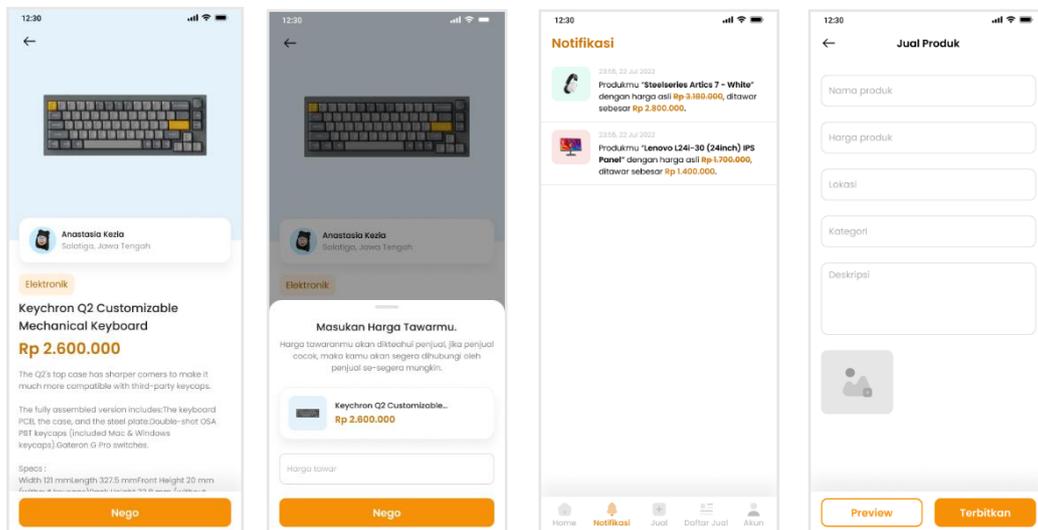
3.1. Implementasi Sistem

Pengembangan aplikasi e-commerce berbasis android sebagai sarana jual beli barang bekas dengan nama aplikasi “JualinAja” telah selesai dibuat setelah melalui proses penelitian yang melibatkan proses pengembangan dan pengujian. Berikut merupakan berbagai halaman dari aplikasi ini.



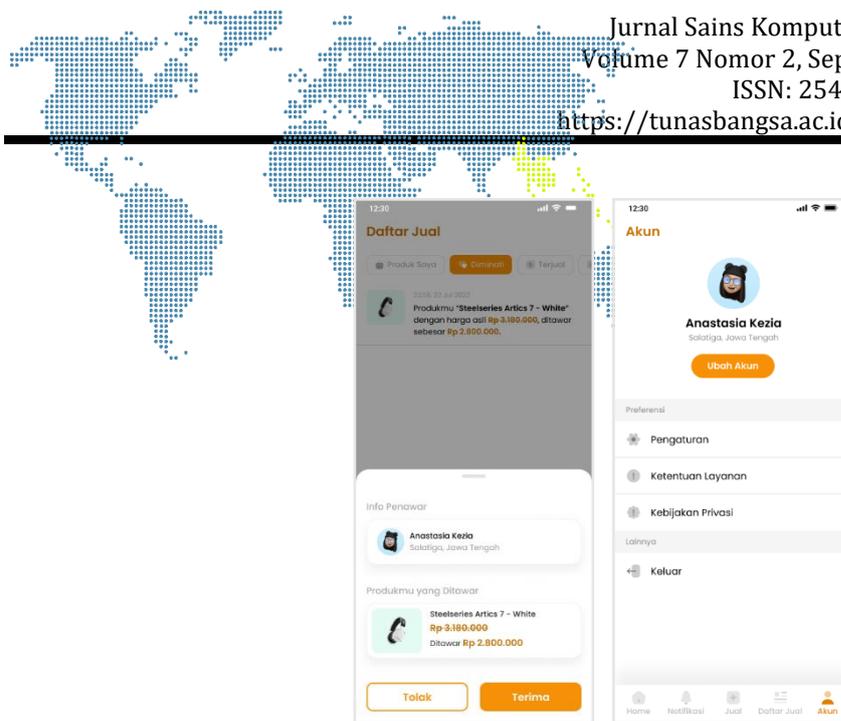
Gambar 2. Tampilan Daftar, Login, Homepage

Setelah merancang tampilan melalui *wireframe* dengan warna hitam dan putih [8]. Maka akan diimplementasikan seperti Gambar 2 yang merupakan tampilan daftar dan masuk yang digunakan pengguna untuk masuk ke dalam aplikasi. Setelah berhasil masuk, maka pengguna akan masuk ke halaman *homepage* untuk melihat semua produk yang dijual.



Gambar 3. Tampilan Detail Produk, Nego, Notifikasi, Jual Produk

Gambar 3 merupakan tampilan detail produk dan nego yang dapat digunakan pengguna untuk melihat detail dan menawar produk. Kemudian tampilan notifikasi untuk melihat tawaran dari pembeli, dan tampilan jual produk untuk menerbitkan produk yang akan dijual.

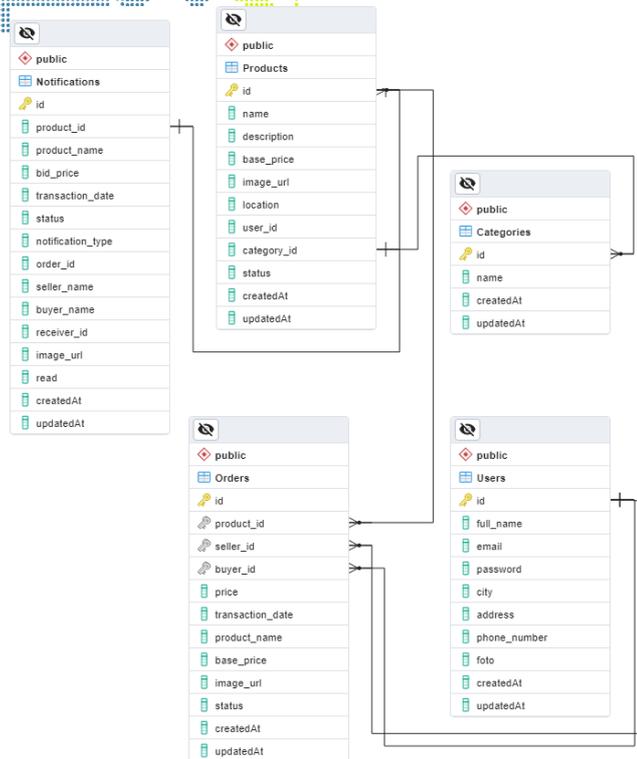


Gambar 4. Tampilan Merima/Menolak Tawaran, Akun

Gambar 4 merupakan tampilan untuk penjual menerima atau menolak tawaran dari pengguna dan tampilan akun untuk memperbaharui identitas pribadi. Aplikasi ini dibuat dengan menggunakan arsitektur MVVM (Model-View-ViewModel), yaitu arsitektur yang memisahkan model (data), view (tampilan), dan ViewModel yang berperan sebagai penghubung antara model dan tampilan [9]. MVVM memiliki beberapa keuntungan, seperti meningkatkan testability, modifiability, dan performance [10].

3.2. Rancangan Basis Data

Sebuah model untuk menggambarkan hubungan antara data dalam database disebut *Entity Relationship Diagram* atau ERD. Diagram ini memiliki tiga komponen yang digunakan yaitu entitas, atribut, dan relasi. Berikut ERD yang digunakan pada aplikasi ini:



Gambar 5. ERD Aplikasi

3.3. Kode Program Konfigurasi API

Pada kode program ini, hanya dipilih satu dari banyaknya kode program. Kode program yang diambil adalah `getProfile` untuk mengambil data profil akun.

Kode Program 1. Kode Program Konfigurasi API

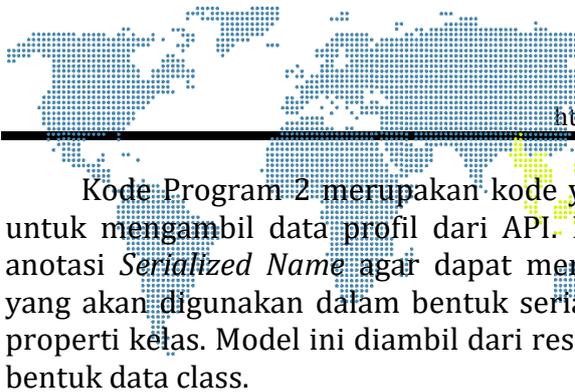
```
1 @GET("auth/user") session_start();
2 suspend fun getProfile(
3     @Header("Authorization") access_token: String
4 ): Response<GetProfileResponse>
```

Kode Program 1 merupakan kode yang digunakan untuk mengambil data dari API. Pada kode tersebut dipilihlah data "auth/user" untuk mengambil data profil yang ditandai dengan @GET.

3.4. Kode Program Pembuatan Model

Kode Program 2. Kode Program pembuatan Model

```
1 data class GetProfileResponse(
2     @SerializedName("id") var id : Int? = null,
3     @SerializedName("full_name") var fullName : String? = null,
4     @SerializedName("email") var email : String? = null,
5     @SerializedName("phone_number") var phoneNumber : String? = null,
6     @SerializedName("address") var address : String? = null,
7     @SerializedName("image_url") var imageUrl : String? = null,
8     @SerializedName("city") var city : String? = null,
9     @SerializedName("created_at") var createdAt : String? = null,
10    @SerializedName("updated_at") var updatedAt : String? = null
11 )
```



Kode Program 2 merupakan kode yang digunakan untuk pembuatan model untuk mengambil data profil dari API. Dalam pembuatan model ini, memakai anotasi *Serialized Name* agar dapat memberikan nama alternatif pada *property* yang akan digunakan dalam bentuk serialisasi, tanpa harus mengubah nama asli properti kelas. Model ini diambil dari respon API yang kemudian diubah ke dalam bentuk data class.

3.5. Kode Program Pembuatan View

Kode Program 3. Kode Program pemuatan View

```
1 <androidx.constraintlayout.widget.ConstraintLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
2 ...
3
4 <androidx.appcompat.widget.AppCompatEditText
5   android:id="@+id/et_pilih_kota"
6   style="@style/GeneralEditText"
7   android:layout_width="0dp"
8   android:layout_height="wrap_content"
9   android:entries="@array/daftar_kota"
10  android:hint="Kota"
11  android:inputType="text"
12  app:layout_constraintLeft_toLeftOf="parent"
13  app:layout_constraintRight_toRightOf="parent"
14  app:layout_constraintTop_toBottomOf="@+id/et_nama" />
15
16 <ImageView
17   android:id="@+id/ic_arrow"
18   android:layout_width="wrap_content"
19   android:layout_height="wrap_content"
20   android:layout_marginStart="16dp"
21   android:layout_marginTop="16dp"
22   android:src="@drawable/ic_baseline_arrow_back_24"
23   app:layout_constraintStart_toStartOf="parent"
24   app:layout_constraintTop_toTopOf="parent" />
25 ...
26   <ImageView
27     android:id="@+id/iv_profile"
28     android:layout_width="match_parent"
29     android:layout_height="match_parent"
30     android:scaleType="centerCrop" />
31
32 </RelativeLayout>
33
34 </com.google.android.material.card.MaterialCardView>
35
36 </androidx.constraintlayout.widget.ConstraintLayout>
```

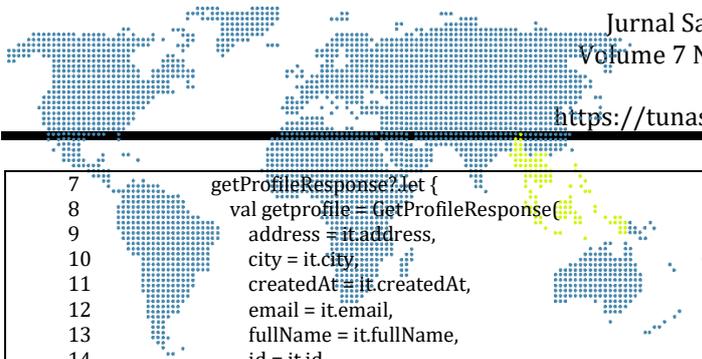
Kode Program 3 merupakan kode yang digunakan untuk pembuatan View, yaitu tampilan UI dari profil. View disini bertanggung jawab untuk menampilkan data dan meneruskan Tindakan pengguna ke ViewModel.

3.6. Kode Program Pembuatan ViewModel

Kode Program 4. Kode Program pemuatan ViewModel

```
1 fun getProfile() {
2   CoroutineScope(Dispatchers.IO).launch {
3     val response = repository.getProfile(repository.getToken())
4     withContext(Dispatchers.Main) {
5       if (response.isSuccessful) {
6         val getProfileResponse = response.body()

```



```
7     getProfileResponse?.let {  
8         val getprofile = GetProfileResponse(  
9             address = it.address,  
10            city = it.city,  
11            createdAt = it.createdAt,  
12            email = it.email,  
13            fullName = it.fullName,  
14            id = it.id,  
15            imageUrl = it.imageUrl,  
16            phoneNumber = it.phoneNumber,  
17            updatedAt = it.updatedAt  
18        )  
19        shouldShowUser.postValue(getprofile)  
20    }  
21    } else {  
22        showResponseError.postValue("Request get Profile Tidak Failed" + response.code())  
23    }  
24 }  
25 }  
26 }
```

memudahkan pengembangan aplikasi yang lebih sederhana.

3.7. Pengujian

Black Box merupakan metode yang digunakan untuk menguji pada aplikasi ini. Pengujian dengan menggunakan Black Box mendapatkan hasil seperti berikut:

Tabel 1. Tabel data pengujian Black Box

No	Skenario	Hasil yang diharapkan	Hasil pengujian	Status
1	Melihat tampilan OnBoarding	Pengguna dapat melihat tampilan OnBoarding saat pertama kali menginstall dan membuka aplikasi	Pengguna berhasil melihat tampilan OnBoarding saat pertama kali menginstall dan membuka aplikasi	<i>Valid</i>
2	Membuat akun baru dan melakukan login	Pengguna dapat masuk ke dalam aplikasi dengan akun yang sudah dibuat.	Pengguna berhasil masuk ke dalam aplikasi dengan akun yang sudah dibuat.	<i>Valid</i>
3	Mencari produk yang diinginkan	Pengguna dapat menemukan produk yang dicari	Pengguna berhasil menemukan produk yang dicari	<i>Valid</i>
4	Memilih kategori pada halaman home	Pengguna dapat melihat produk sesuai dengan kategori yang dipilih	Pengguna berhasil melihat produk sesuai dengan kategori yang dipilih	<i>Valid</i>
5	Memilih produk pada daftar produk di halaman Home	Pengguna dapat memilih produk yang berada pada daftar produk di halaman Home	Pengguna berhasil memilih produk yang berada pada daftar produk di halaman home	<i>Valid</i>
6	Melihat detail produk	Pengguna dapat melihat detail produk yang diinginkan	Pengguna berhasil melihat detail produk yang	<i>Valid</i>

No	Skenario	Hasil yang diharapkan	Hasil pengujian	Status
			diinginkan	
7	Menawar/menego harga produk	Pengguna dapat menawar/menego harga produk yang diinginkan	Pengguna berhasil menawar/menego harga produk yang diinginkan	<i>Valid</i>
8	Menerbitkan produk	Pengguna dapat menerbitkan produk	Pengguna berhasil menerbitkan produk yang sudah dibuat.	<i>Valid</i>
9	Melihat <i>preview</i> sebelum menerbitkan produk	Melihat <i>preview</i> sebelum menerbitkan produk	Pengguna berhasil melihat <i>preview</i> sebelum menerbitkan produk	<i>Valid</i>
10	Melihat produk yang sudah diterbitkan pada bagian Produk Saya di halaman Daftar Jual	Pengguna dapat melihat produk yang sudah diterbitkan pada bagian Produk Saya di halaman Daftar Jual	Pengguna berhasil melihat produk yang sudah diterbitkan pada bagian Produk Saya di halaman Daftar Jual	<i>Valid</i>
11	Memperbaharui identitas produk	Pengguna dapat memperbaharui identitas produk pada bagian Produk Saya melalui halaman Daftar Jual	Pengguna berhasil memperbaharui identitas produk pada bagian Produk Saya melalui halaman Daftar Jual	<i>Valid</i>
12	Menghapus produk	Pengguna dapat menghapus produk pada bagian Produk Saya melalui halaman Daftar Jual	Pengguna berhasil menghapus produk pada bagian Produk Saya melalui halaman Daftar Jual	<i>Valid</i>
13	Melihat tawaran yang diajukan	Pengguna dapat melihat tawaran yang diajukan pada bagian Diminati di halaman Daftar Jual	Pengguna berhasil melihat tawaran yang diajukan pada bagian Diminati di halaman Daftar Jual	<i>Valid</i>
14	Menerima atau tmenolak tawaran yang diajukan	Pengguna dapat menerima atau menolak tawaran yang diajukan	Pengguna berhasil menerima atau menolak tawaran yang diajukan	<i>Valid</i>
15	Menghubungi via aplikasi WhatsApp	Pengguna dapat menghubungi pembeli via aplikasi WhatsApp setelah menerima tawaran	Pengguna berhasil menghubungi pembeli via aplikasi WhatsApp setelah menerima tawaran	<i>Valid</i>
16	Melihat produk yang terjual	Pengguna dapat melihat daftar produk yang sudah terjual pada bagian Terjual di	Pengguna berhasil melihat daftar produk yang sudah terjual pada bagian	<i>Valid</i>

No	Skenario	Hasil yang diharapkan	Hasil pengujian	Status
		halaman Daftar Jual	Terjual di halaman Daftar Jual	
17	Mengubah status produk	Pengguna dapat mengubah status produk yaitu berhasil terjual atau membatalkan transaksi	Pengguna berhasil mengubah status produk yaitu berhasil terjual atau membatalkan transaksi	<i>Valid</i>
18	Mengubah atau memperbaharui profil akun	Pengguna dapat mengubah atau memperbaharui identitas akun	Pengguna berhasil mengubah atau memperbaharui identitas akun	<i>Valid</i>

Berdasarkan hasil pengujian yang sudah tertera pada Tabel 1, dapat dilihat dan disimpulkan bahwa aplikasi telah berjalan dengan baik dan lancar sesuai dengan hasil yang diharapkan. Sehingga secara fungsionalitas aplikasi sudah siap untuk digunakan.

4. SIMPULAN

Berdasarkan penelitian yang sudah dilakukan serta hasil pengujian yang didapat, dapat disimpulkan bahwa Pengembangan Aplikasi E-Commerce berbasis Android sebagai Sarana Jual Beli Barang Bekas menggunakan Metode Waterfall dapat dijadikan sarana untuk membantu masyarakat dalam menemukan produk atau barang bekas yang masih layak pakai. Meskipun aplikasi ini masih memiliki fitur yang terbatas, namun aplikasi ini dapat membantu memenuhi kebutuhan masyarakat yang masih ingin membeli produk atau barang bekas.

DAFTAR PUSTAKA

- [1] D. Kurniawan Gultom, D. Aulia, M. Rheza Febriansyah, R. Wahyudi, S. Hafiza Pasaribu, and Y. Rachma, "E-Commerce Berbasis Website Sebagai Sarana Jual Beli Barang Bekas Layak Pakai," *Proceeding Semin. Nas. Kewirausahaan*, vol. 2, no. 1, pp. 614–619, 2021, doi: 10.30596%2Fsnk.v2i1.8347.
- [2] W. Eko, S. Septiyan, and B. Saputro, "Rancang Bangun Aplikasi E-Commerce Smartphone Second Dengan Metode Waterfall," vol. 10, no. 4, pp. 54–60, 2018.
- [3] P. Iqlima and S. Nita, "Perancangan Aplikasi E-Commerce Pada Goresan Pena Studio Berbasis Android," *Seminar Nasional Teknologi Informasi dan Komunikasi*, pp. 97–107, 2021.
- [4] A. S. Wijaya and J. F. Andry, "Perancangan Aplikasi E-Commerce Berbasis Android Pada Ud Hoky Celluler Shop," *J. Teknoinfo*, vol. 15, no. 2, p. 97, 2021, doi: 10.33365/jti.v15i2.1065.
- [5] T. Kurniawati, M. Ritonga, J. E. Marna, Y. Artati, and W. P. Rahayu, "Pengembangan Aplikasi E-Smes Berbasis Android dengan Waterfall Method Sebagai Solusi Pemasaran dan Pengelolaan Usaha UMKM," *J. Ecogen*, vol. 4, no. 4, p. 594, 2021, doi: 10.24036/jmpe.v4i4.12414.



- [6] D. Silvi Purnia and Y. Herlina, "Implementasi Metode Watterfall pada Aplikasi Marketplace Laundry Berbasis Android," *Indones. J. Softw. Eng.*, vol. 7, no. 2, pp. 159–168, 2021.
- [7] M. Aida, "Sistem Informasi Laundry Menggunakan Metode Waterfall Berbasis Android Pada Simply Fresh Laundry," *J. Teknol. Inf.*, vol. 3, no. 2, p. 122, 2019, doi: 10.36294/jurti.v3i2.934.
- [8] G. Karnawan, "Implementasi User Experience Menggunakan Metode Design Thinking Pada Prototype Aplikasi Cleanstic," *J. Teknoinfo*, vol. 15, no. 1, p. 61, 2021, doi: 10.33365/jti.v15i1.540.
- [9] M. Arif, A. Musthafa, and D. Muriyatmoko, "Implementasi Pola Arsitektur Model-View-ViewModel (MVVM) pada Sistem Informasi Akademik Universitas Darussalam Gontor Berbasis Mobile," pp. 1–12, 2019.
- [10] Deni Murdiani and Muhamad Sobirin, "Perbandingan Metodologi Waterfall Dan Rad (Rapid Application Development) Dalam Pengembangan Sistem Informasi," *J. Inform. Teknol. dan Sains*, vol. 4, no. 4, pp. 302–306, 2022, doi: 10.51401/jinteks.v4i4.2008.