

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Hasil Pengumpulan Data

Hasil pengumpulan data pada penelitian ini diperoleh dari instrumen pengumpulan data, yaitu melakukan observasi dan wawancara kepada pihak yang terkait di *Corry Wedding* Kraksaan. Berikut pemaparan atas hasil dari pengumpulan data yang dilakukan.

##### 4.1.1 Hasil Observasi

Berdasarkan observasi penelitian pada tanggal 20 maret 2021 yang dilakukan dengan melakukan pengamatan secara langsung Pada *Corry Wedding* yaitu mengamati beberapa kegiatan sistem berjalan yang berhubungan dengan penelitian, seperti proses pemesanan, pencatatan transaksi, dan promosi *wedding*, pada observasi ini menghasilkan bahwa pada pengolahan data, sistem pemesanan dan promosi jasa WO pada *Corry Wedding Organizer* masih bisa dikatakan belum begitu efektif dan efisien. Pemesanan jasa WO dilakukan menggunakan aplikasi *WhatsApp* dan juga Klien harus ke tempat WO untuk melakukan pemesanan. Dalam pencatatan transaksi masih dilakukan secara manual yaitu menggunakan buku. Sementara itu promosi *Wedding Organizer* masih dilakukan melalui media sosial. sedangkan permasalahan yang dialami oleh konsumen yaitu kurangnya informasi seperti harga, alamat tempat penyewaan, rute, jarak, serta belum tersedianya proses penyewaan perlengkapan *Wedding Organizer* di Kabupaten Probolinggo secara online dan mereka masih menghubungi WO satu-persatu dengan mencari terlebih dahulu informasi kesesuaian harga dan ketersediaan tanggal.

##### 4.1.2 Hasil Wawancara

Adapun hasil wawancara yang dilaksanakan pada tanggal 20 maret 2021 di *Corry Wedding Organizer* kepada asisten *Corry Wedding* adalah sebagai berikut :

- 
- a. *Corry Wedding Organizer* didirikan pada tahun 2018.
  - b. Alur penyewaan pada *Corry Wedding* ialah pelanggan datang ke tempat vendor dan admin akan memberikan brosur paket pernikahan dan menjelaskan paket tersebut, selanjutnya admin memberikan formulir pemesanan untuk diisi oleh pelanggan, setelah mengisi formulir pelanggan menyerahkan formulir tersebut kepada admin, jika sudah lengkap admin akan mencatat data pemesanan di buku, dan memberikan kwitansi kepada pelanggan.
  - c. Pelayanan yang ditawarkan ialah paket pernikahan, make up, pakaian/gaun, dekorasi, dan catering.
  - d. Metode pembayaran bisa menggunakan cash atau transfer, pembayaran dilakukan pada awal pemesanan yaitu membayar DP terlebih dahulu selanjutnya membayar lunas setelah selesai acara.
  - e. Pencatatan transaksi masih menggunakan buku.
  - f. Kendala yang sering terjadi ialah data pelanggan yang belum terkomputerisasi, media promosi hanya dilakukan melalui media sosial.

#### 4.2 Hasil Analisis dan Desain

Data yang diperoleh dari hasil observasi pada *Corry Wedding Organizer* dapat digunakan sebagai referensi dalam penyusunan aplikasi *e-marketplace wedding organizer* berbasis android agar lebih akurat. Data-data tersebut tentunya sangat membantu dalam pembuatan analisis sistem dan desain sistem pada penelitian ini. Berikut pemaparan hasil perbandingan antara sistem lama dan sistem baru serta desain sistem yang akan berjalan.

##### 4.2.1 Analisis Sistem Lama

Analisis sistem lama merupakan hasil analisis setelah penelitian terhadap sistem dan kinerja yang selama ini berjalan. Adapun hasil analisis sistem lama yang berjalan dalam proses penyewaan *Wedding Organizer* ialah pelanggan datang ke tempat Vendor, kemudian Admin memberikan formulir pemesanan kepada pelanggan, pelanggan kemudian mengisi formulir tersebut seperti nama, alamat, tanggal pernikahan dan paket apa saja yang akan diambil serta membayar DP sesuai dengan

perjanjian, selanjutnya pelanggan menyerahkan persyaratan dan formulir tersebut kepada admin, setelah itu admin menerima dan mengecek kelengkapan jika sudah lengkap admin mencatat data pelanggan di buku yang sering kali terjadi kesalahan penulisan, kemudian admin memberikan kwitansi kepada pelanggan.

#### 4.2.2 Analisis Sistem Baru

Analisis sistem baru adalah hasil analisis yang berasal dari penelitian pada sistem lama sebagai pengembangan sistem baru yang akan menjadi solusi dan bermanfaat dalam penggunaannya. Adapun hasil analisis sistem baru pada penelitian ini adalah membangun Aplikasi *E-marketplace Wedding Organizer* yang dapat membantu dalam penyewaan *Wedding Organizer*, aplikasi ini diimplementasikan dengan bahasa pemrograman Android.

Aplikasi ini mampu membuat akun Vendor, membuat akun user, mengupload produk Vendor, menampilkan transaksi antara vendor dan user, membuat informasi promo bagi vendor.

#### 4.2.3 Desain Sistem

Tahap ini memaparkan formulasi dan pemilihan alternatif solusi. Berdasarkan permasalahan sistem pada *Corry Wedding*, maka dibutuhkan suatu perencanaan dalam pengembangan aplikasi sehingga mampu mempermudah kinerja dalam penyewaan *Wedding Organizer* sesuai dengan analisa yang berasal dari data yang didapat dan membutuhkan suatu Aplikasi *E-marketplace Wedding Organizer* dari proses pemantauan dan pencatatan manual menjadi terkomputerisasi.

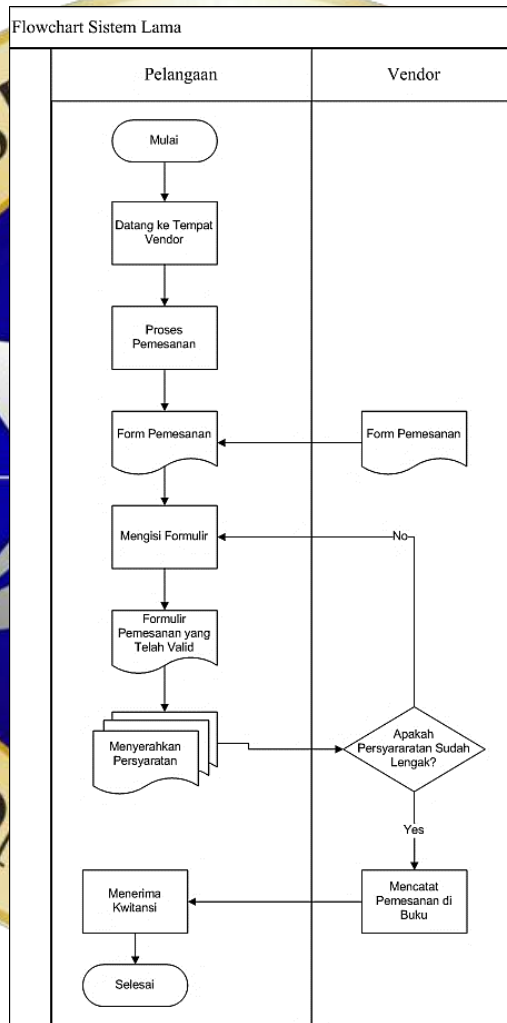
Adapun sistem yang terkomputerisasi memerlukan suatu desain perancangan sistem sebagai acuan dalam pembuatan aplikasi, agar proses pembuatan sistem menjadi lebih tertata dan sesuai dengan prosedur yang ada. Selain itu, kegunaan desain sistem yaitu untuk mengetahui dan memberikan gambaran sistem secara menyeluruh, baik sistem yang sedang digunakan ataupun yang akan diimplementasikan.

Sedangkan desain atau perancangan sistem yang digunakan

dalam proses pembuatan aplikasi *E-marketplace Wedding Organizer* terdiri dari *Flowchart*, *Data Flow Diagram (DFD)*, *Eternity Relationship Diagram (ERD)*, desain *Database* dan desain *Interfacenya*. Berikut ini penjabaran mengenai desain-desain tersebut :

a. *Flowchart* sistem lama

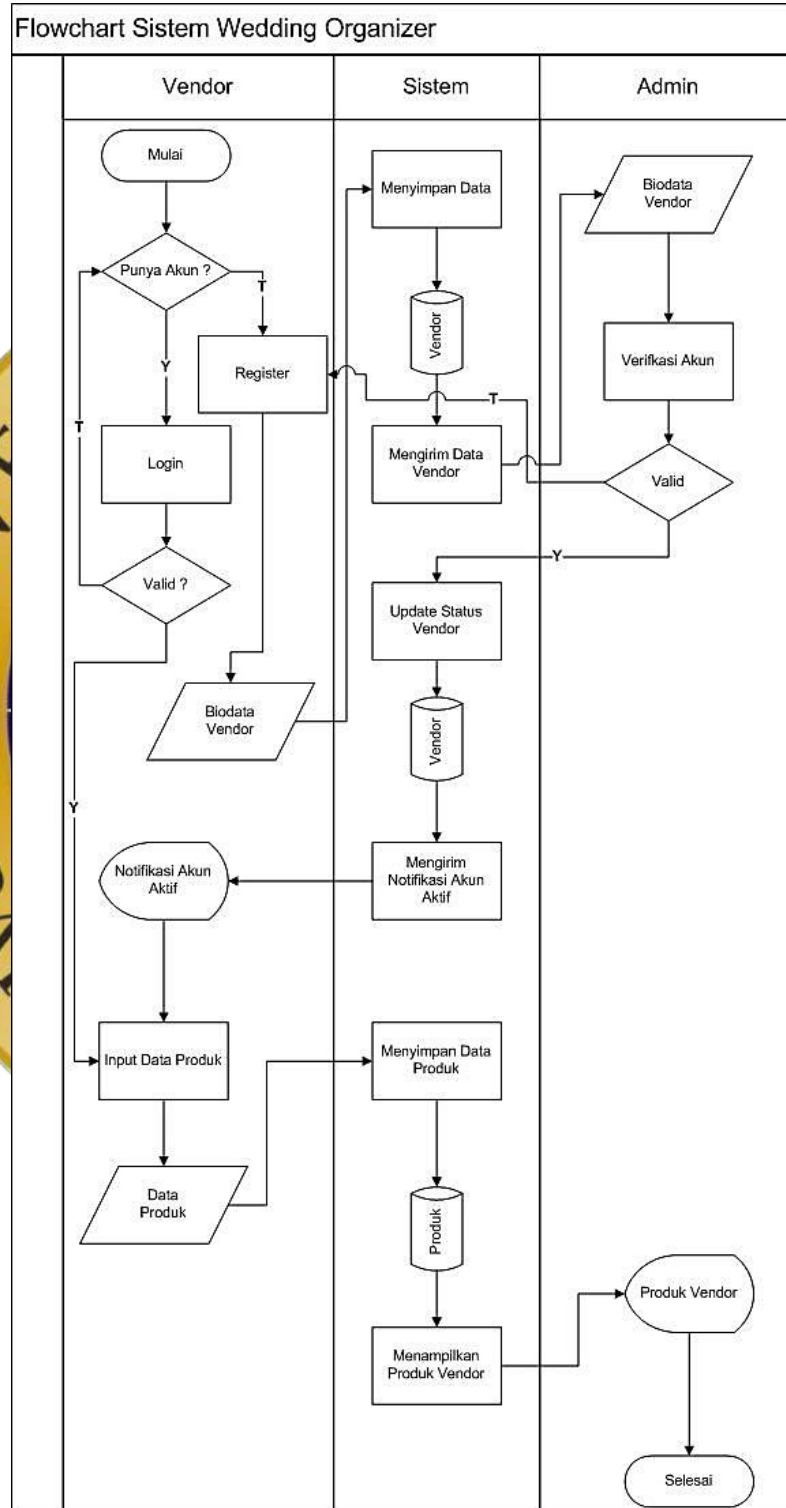
*Flowchart* sistem lama yang berjalan selama ini pada proses penyewaan *Wedding Organizer* sebagaimana gambar berikut :



**Gambar 4.1** *Flowchart* sistem lama

b. *Flowchart Sistem Baru*

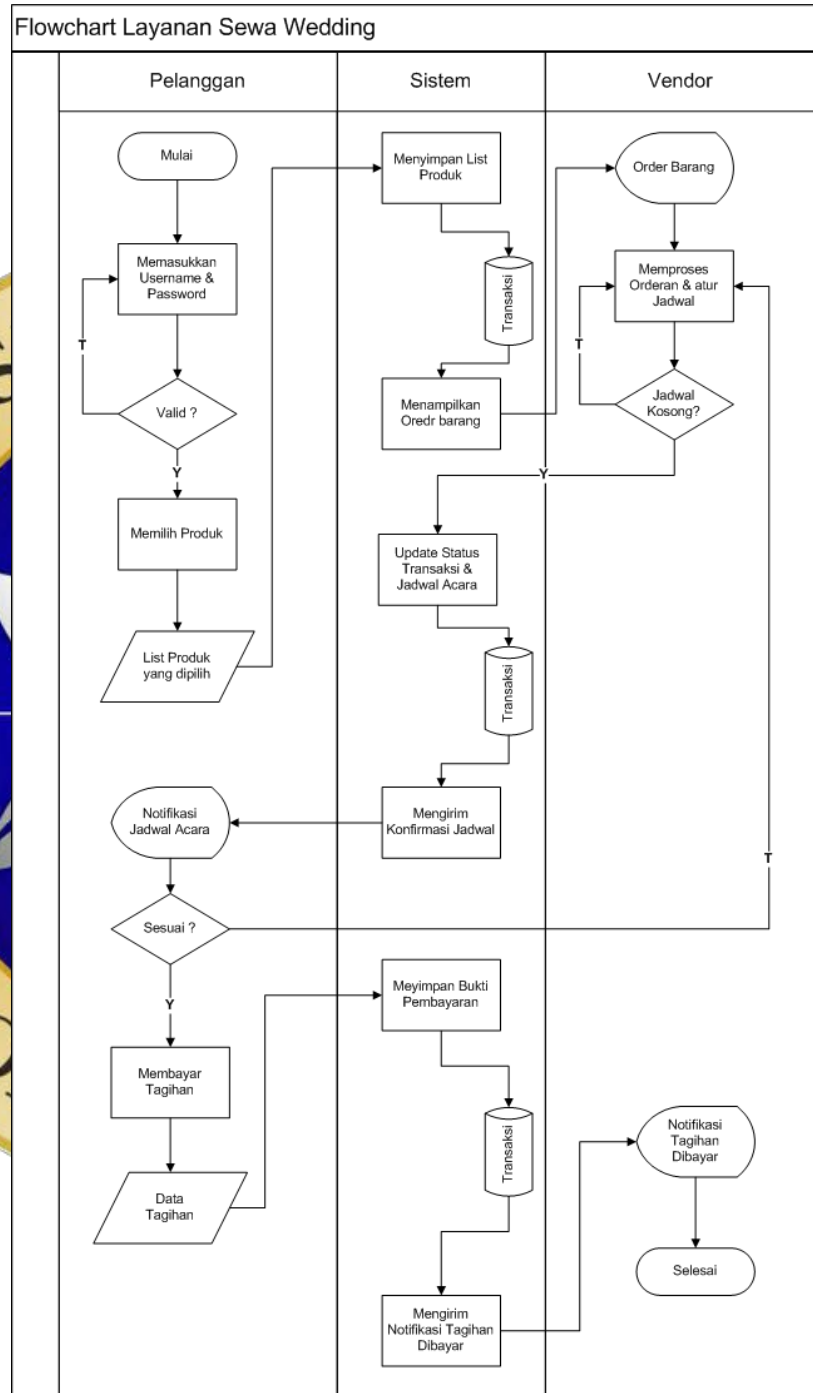
*Flowchart sistem baru pada aplikasi e-marketplace Wedding Organizer dapat dilihat pada gambar berikut:*



**Gambar 4.2** *Flowchart sistem baru*

c. Flowchat Layanan Sewa Wedding

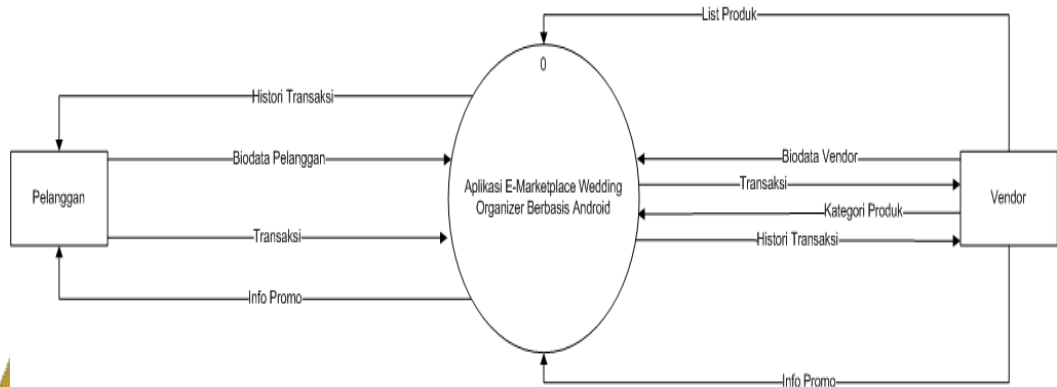
Flowchat layanan sewa wedding organizer pada aplikasi e-marketplace dapat dilihat pada gambar berikut:



**Gambar 4.3** Flowchart layanan sewa wedding organizer

d. *Diagram Context*

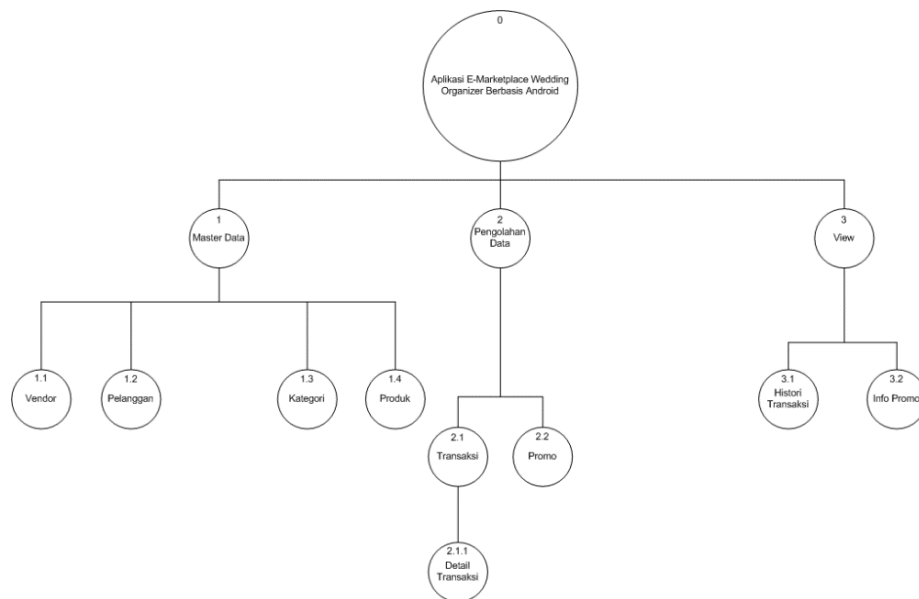
*Data Flow Diagram* (DFD) pada *Diagram Context* yang juga dapat disebut DFD level 0 menggambarkan keterkaitan antara Sistem informasi dengan entitas-entitas yang saling berhubungan secara global. Adapun diagram context Aplikasi *E-marketplace Wedding Organizer* adalah sebagai berikut :



**Gambar 4.4** *Diagram Context*

e. *Decomposition Diagram*

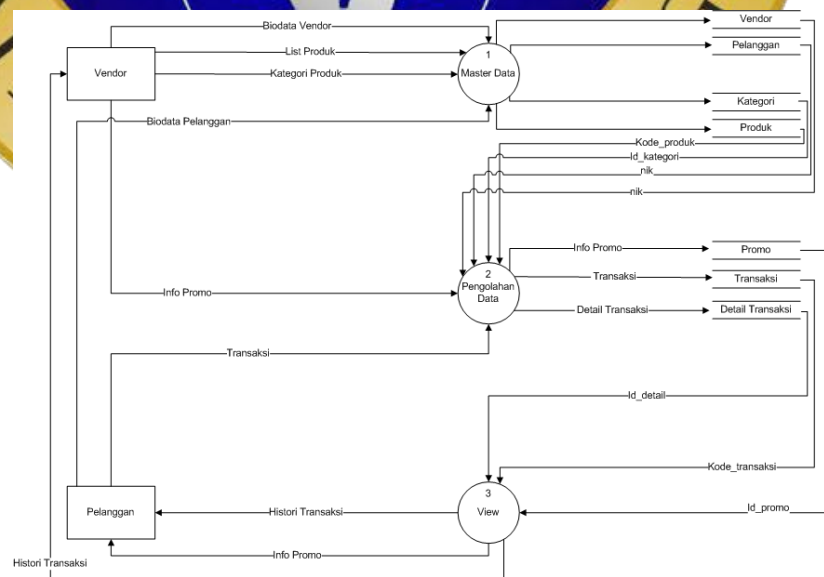
*Decomposition Diagram* adalah alat yang digunakan untuk menggambarkan bagan berjenjang dan alat perencanaan untuk model proses yang lebih detail. Bagan ini disebut juga bagan hierarkikarena menunjukkan dekomposisi fungsional *top-down* dan struktur sistem. Dekomposisi diagram dapat di gambarkan dengan menggunakan notasi proses yang digunakan dalam *Data Flow Diagram* (DFD). Gambar di bawah ini merupakan *Decomposition Diagram* pada Aplikasi *E-marketplace Wedding Organizer*.



**Gambar 4.5** *Decomposition Diagram*

f. **Data Flow Diagram (DFD) Level 1**

Pada Data Flow Diagram Level 1 alur sistem lebih terperinci. Diagram ini menambah proses hubungan yang terjadi pada sistem entitas dan proses akan membentuk sistem secara utuh. DFD level 1 Aplikasi E-marketplace Wedding Organizer dapat dilihat pada gambar 4.5 di bawah ini

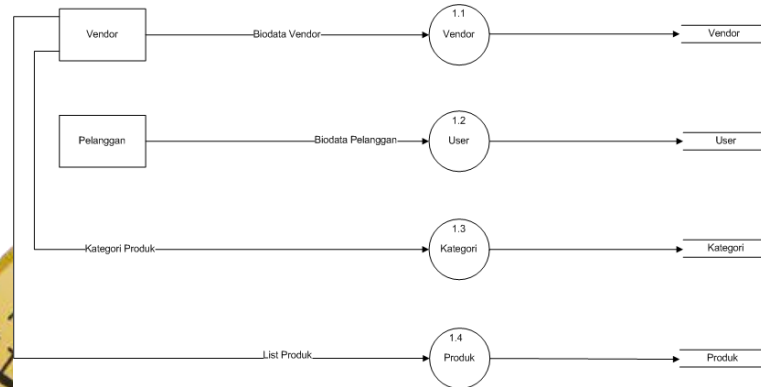


**Gambar 4.6** *Data Flow Diagram (DFD) Level*



g. Data Flow Diagram (DFD) Level 2 Proses 1 Master Data

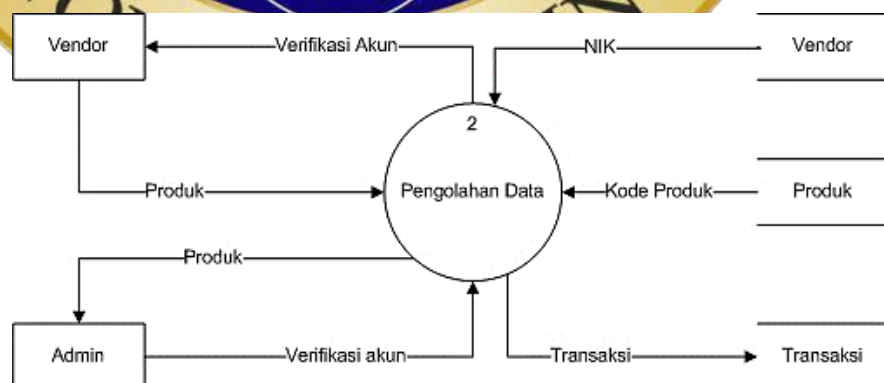
Pada level ini DFD hanya fokus terhadap pembahasan kinerja proses Master Data pada sistem. Sedangkan Master data sendiri sebagai variabel database Vendor. Adapun aliran data pada Master data dapat dilihat pada gambar di bawah ini :



**Gambar 4.7** Data Flow Diagram (DFD) Level 2 Proses 1 Master Data

h. Data Flow Diagram (DFD) Level 2 Proses 2 Pengolahan Data

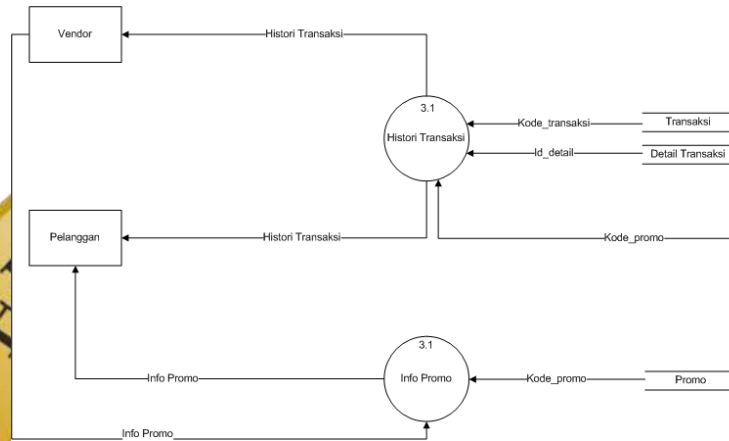
Data Flow Diagram (DFD) Level 2 yang berupa proses pengolahan data mencakup proses data Vendor, produk, transaksi . Dalam hal ini pihak-pihak yang terlibat dalam pengolahan data di antaranya adalah Vendor dan Admin. Gambar di bawah ini memaparkan alir data pada proses pengolahan data



**Gambar 4.8** Data Flow Diagram (DFD) Level 2 Proses 2 pengolahan Data

i. Data Flow Diagram (DFD) Level 2 Proses 3 Laporan

Laporan merupakan proses terakhir dari sistem monitoring dan evaluasi *Wedding Organizer*. Laporan yang dimaksud berupa laporan transaksi penjualan yang diserahkan kepada Admin dan Vendor. Di bawah ini merupakan gambar proses Laporan sistem monitoring dan evaluasi *Wedding Organizer*.

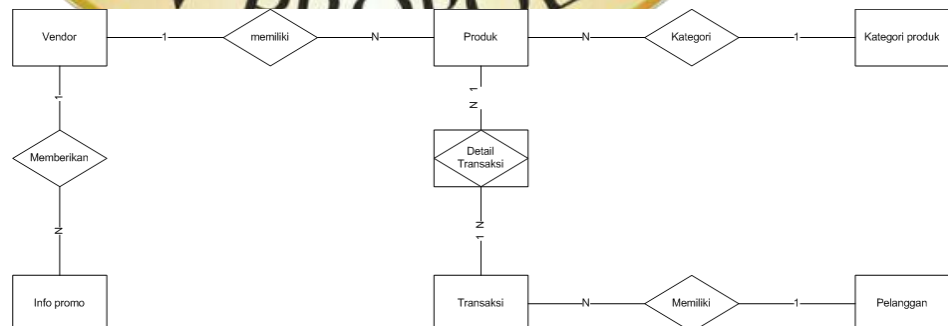


**Gambar 4.9** Data Flow Diagram (DFD) Level 2 Proses 3 Laporan

j. Entity Relationship Diagram (ERD)

Setelah merancang DFD yang diterapkan terhadap sistem yang akan dibangun, tahap yang akan dilewati selanjutnya adalah penentuan pola hubungan antar entitas. Oleh karena itu dibutuhkan

Entity Relationship Diagram (ERD) dalam penyempurnaan desain sistem pada suatu sistem. Sebab dengan adanya ERD akan dapat dipahami istilah hubungan Many to Many, One to Many, Many To One dan One to One. Dengan begitu beberapa entitas yang saling berkaitan akan dipahami secara mendetail. ERD tersebut dapat dilihat pada gambar 4.10 di bawah ini.



**Gambar 4.10** Entity Relationship Diagram(ERD)

Kamus data :

Vendor	id, nik, nama_user, tgl_lahir, jenis_kelamin, provinsi, kabupaten, kecamatan, desa, alamat, no_telphone, username, password, foto, verifikasi, komentar, blokir
Pelanggan	id, nik, nama_user, tgl_lahir, jenis_kelamin, provinsi, kabupaten, kecamatan, desa, alamat, no_telphone, username, password, foto, verifikasi, komentar, blokir
Produk	id, nama_produk, kode_produk, harga, deskripsi, nik_vendor, kategori, status, foto
Kategori Produk	id, nama
Transaksi	id, nik_pelanggan, nik_vendor, kode_transaksi, tgl_acara, tgl_persetujuan, status
Detail Transaksi	id, kode_produk, nama_produk, harga, jumlah, kode_transaksi
Info Promo	id_promo, kode_promo, harga_promo, kode_produk, status

#### 4.2.4. Desain Database

*Database* merupakan variabel penyimpanan data pada suatu sistem serta pengolahannya. Oleh sebab itu sangat diperlukan pola struktur *database* yang baik untuk menghindari redudansi data. *Database* pada sistem ini menggunakan aplikasi *MySQL* yang terdiri dari beberapa tabel yang dipaparkan di bawah ini :

##### 1. Tabel Vendor

**Tabel 4.1** Tabel Vendor

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	nik	varchar(16)	

**Tabel 4. 1** Tabel Vendor (lanjutan)

3	nama_user	varchar(100)	
4	tgl_lahir	date	
5	jenis_kelamin	varchar(1)	
6	provinsi	varchar(100)	
7	kabupaten	varchar(100)	
8	kecamatan	varchar(100)	
9	desa	varchar(100)	
10	alamat	varchar(255)	
11	no_telphone	varchar(100)	
12	username	varchar(100)	
13	password	varchar(100)	
14	foto	varchar(100)	
16	komentar	varchar(255)	
17	blokir	varchar(1)	

**Tabel 4. 2** Tabel Pelanggan

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	nik	varchar(16)	
3	nama_user	varchar(100)	
4	tgl_lahir	date	
5	jenis_kelamin	varchar(1)	
6	provinsi	varchar(100)	
7	kabupaten	varchar(100)	
8	kecamatan	varchar(100)	
9	desa	varchar(100)	
10	alamat	varchar(255)	
11	no_telphone	varchar(100)	
12	username	varchar(100)	

**Tabel 4. 3** Tabel Pelanggan (Lanjutan)

13	password	varchar(100)	
14	foto	varchar(100)	
15	verifikasi	enum('0', '1', '2')	
16	komentar	varchar(255)	
17	blokir	varchar(1)	

**Tabel 4. 3** Tabel Pelanggan

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	nama_produk	varchar(100)	
3	kode_produk	varchar(25)	
4	harga	int(11)	
5	deskripsi	text	
6	nik_vendor	varchar(16)	
7	kategori	varchar(5)	
8	status	enum('T', 'F')	
9	foto	varchar(100)	

**Tabel 4. 4** Tabel Kategori Produk

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	nama	varchar(100)	

**Tabel 4. 5** Tabel Transaksi

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	user	bigint(16)	
3	vendor	bigint(16)	
4	kode_transaksi	varchar(25)	

**Tabel 4. 5 Tabel Transaksi (lanjutan)**

5	tgl_acara	Datetime	
6	tgl_persetujuan	Datetime	
7	Dp	Int	
8	bukti_dp	Varchar	
9	Lunas	Int	

**Tabel 4. 6 Tabel Detail Transaksi**

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	Kode_produk	varchar(25)	
3	nama_produk	varchar(100)	
4	Harga	Int(11)	
5	Jumlah	Int(11)	
6	kode_transaksi	varchar(25)	

**Tabel 4. 7 Tabel Promo**

#	Name	Type	Extra
1	id	bigint(20)	AUTO_INCREMENT
2	Kode_Promo	varchar(25)	
3	Harga_Promo	Int(11)	
4	Kode_produk	varchar(25)	
5	Status	Int(1)	

#### 4.2.5. Desain Interface

Pada desain ini akan ditampilkan gambaran *interface* aplikasi yang akan diimplementasikan ke dalam sebuah perangkat lunak. Gambaran aplikasi yang dibuat berupa *interface* untuk melakukan proses yang akan berjalan. Gambar-gambar di bawah ini akan menunjukkan desain *interface* pada Aplikasi *E-marketplace Wedding Organizer*.

##### 1. Desain Interface Login



##### 2. Desain Interface Register



**Gambar 4.12** Desain Interface Register

##### 3. Desain Interface Dashboard



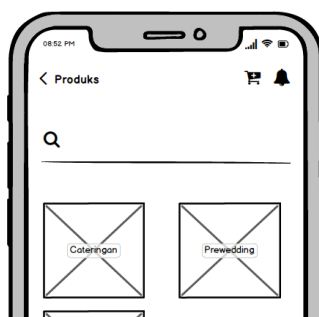


Gambar 4.13 Desain *Interface Dashboard*

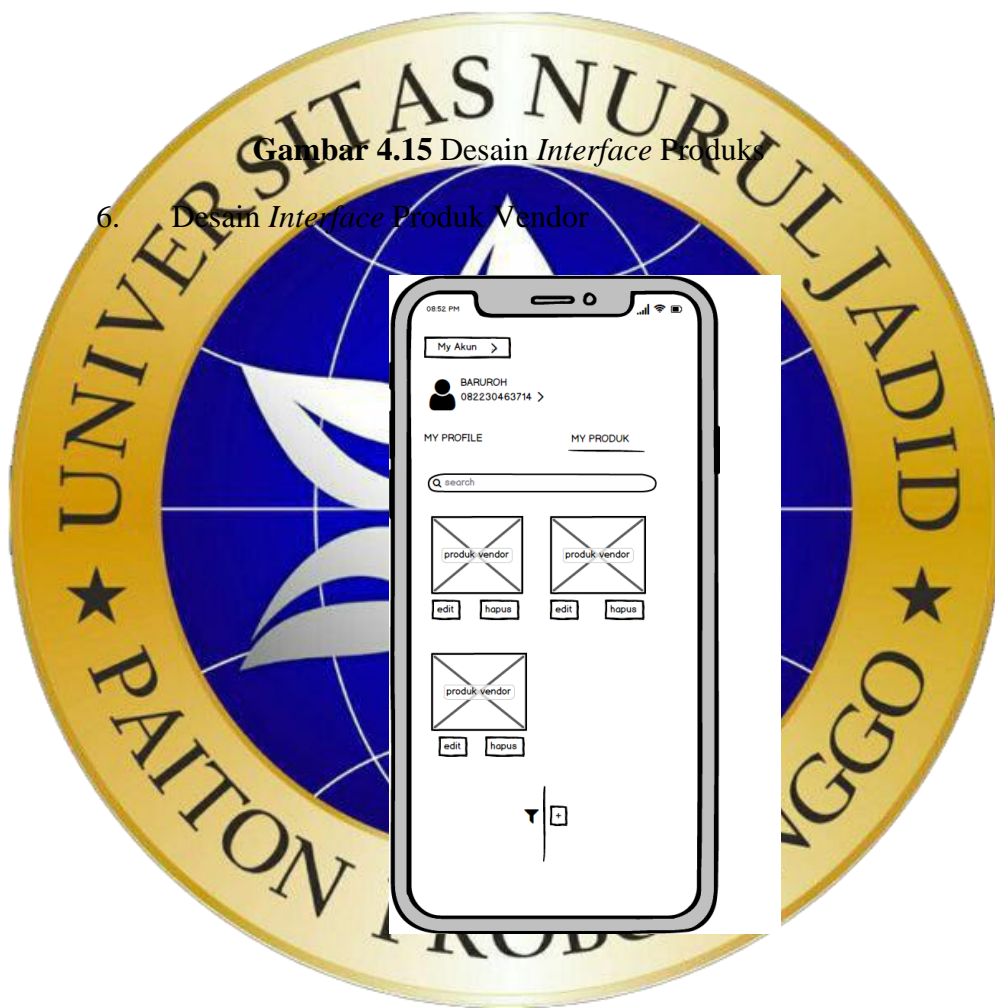
4. Desain *Interface Profil*

Gambar 4.14 Desain *Interface Profil*

5. Desain *Interface Produk*





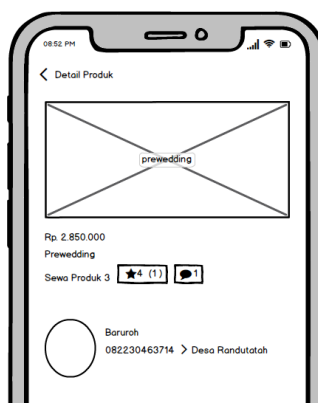


Gambar 4.15 Desain *Interface* Produk

6. Desain *Interface* Produk Vendor

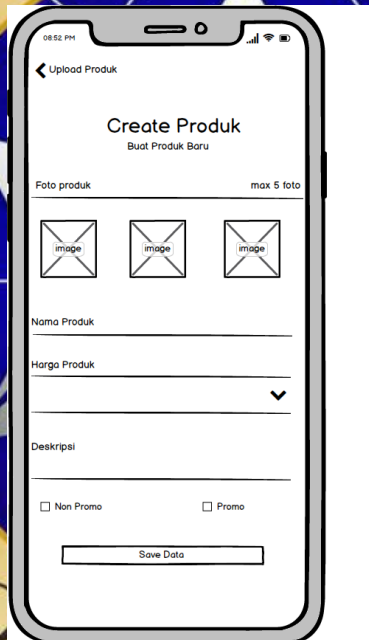
Gambar 4.16 Desain *Interface* Produk Vendor

7. Desain *Interface* Detail Produk



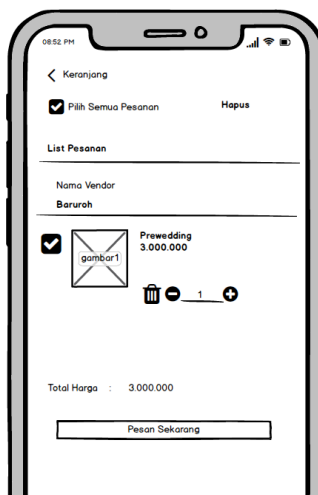
**Gambar 4.17** Desain *Interface* Detail Produk

8. Desain *Interface* Upload Produk



**Gambar 4.18** Desain *Interface* Upload Produk

9. Desain *Interface* Keranjang Pembelian



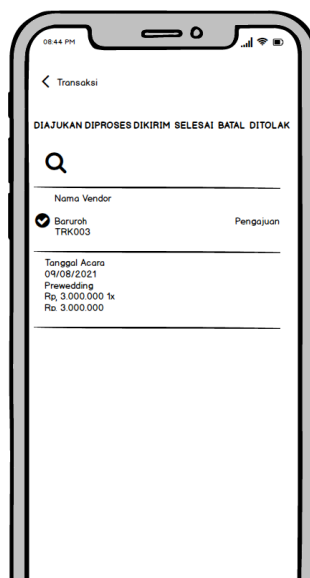
Gambar 4.19 Desain *Interface* Keranjang Pembelian

10. Desain *Interface* Pesanan



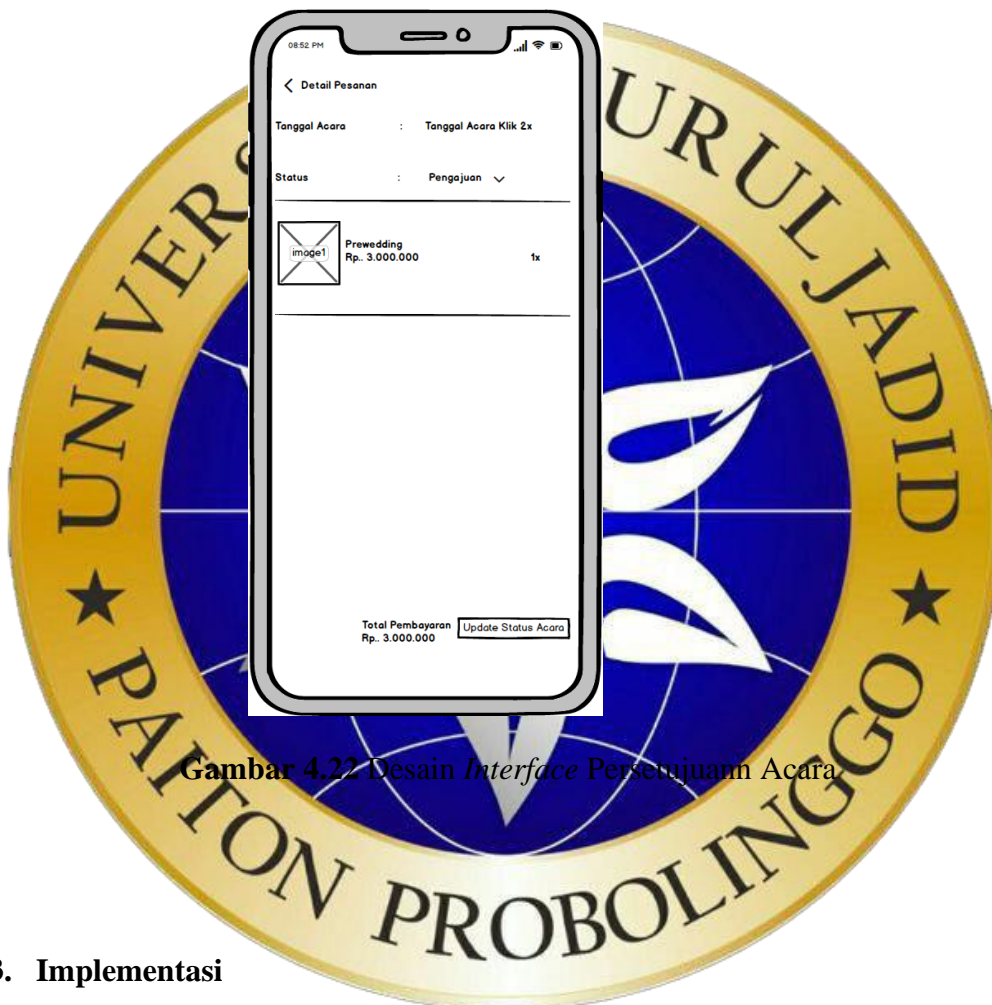
Gambar 4.20 Desain *Interface* Pesanan

11. Desain *Interface* Transaksi



**Gambar 4.21** Desain *Interface* Transaksi

12. Desain *Interface* Persetujuan Acara



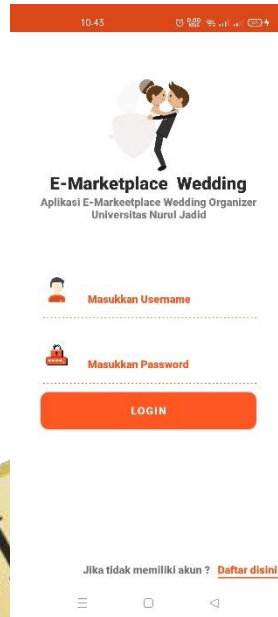
**Gambar 4.22** Desain *Interface* Persetujuan Acara

### 4.3. Implementasi

Dalam implementasi Aplikasi *E-marketplace Wedding Organizer* ini menghasilkan beberapa tampilan (*Interface*) yang digunakan, di antaranya adalah sebagai berikut

#### 4.3.1. Halaman Login

Untuk mengakses aplikasi *E-marketplace Wedding organizer* Langkah pertama yang harus dilakukan ialah login.. Tampilan login terlihat pada gambar 4.24.



Gambar 4.24 Halaman Login

#### 4.3.2. Halaman Register

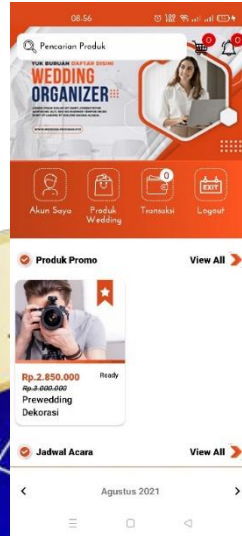
Apabila pengguna tidak memiliki akun untuk loginn maka harus mendaftar terlebih dahulu, sebagai vendor maupun pelanggan. Tampilan register terlihat pada gambar 4.25.



Gambar 4.25 Halaman Register

### 4.3.3. Halaman *Dashboard*

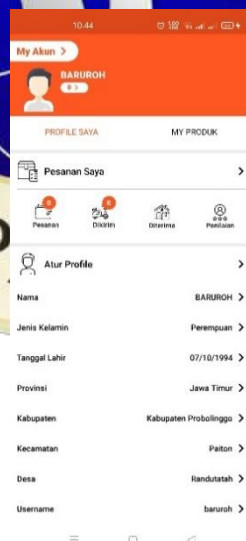
Setelah berhasil login maka pengguna akan disuguhkan dengan dashboard yang menarik dengan berbagai menu. Tampilan *Dashboard* terlihat pada gambar 4.26.



Gambar 4.26 Halaman *Dashboard*

### 4.3.4. Halaman *Profile*

Pada halaman produk vendor, vendor akan ditampilkan dengan berbagai produk dari vendor tersebut. Tampilan halaman *Profile* vendor terlihat pada gambar 4.27.



Gambar 4.27 Halaman *Profile*

Segmen Program 4.1 *Profile*

```

@Override
public void onResponse(Call<rlogin> call, Response<rlogin>
response) {
    String value = response.body().getValue();
    String message = response.body().getMessage();
    try {
        if (value.equals("1")) {
            resLogin = response.body().getResult();
            mlogin data = resLogin.get(0);
            tnama.setText(data.getNama());
            tjenis.setText(data.getGender());
            ttgl.setText(data.getTgl());
            tpro.setText(data.getPro());
            tkab.setText(data.getKab());
            tkec.setText(data.getKec());
            tdesa.setText(data.getDesa());
            tuser.setText(data.getUser());
            tpass.setText(data.getPass());
            tkontak.setText(data.getKontak());
            talamat.setText(data.getAlamat());
            SharedPreferences pref =

```

#### Segmen Program 4.1 Profile(lanjutan)

```

getActivity().getSharedPreferences("akun",
getActivity().MODE_PRIVATE);
    SharedPreferences.Editor editor = pref.edit();
    editor.putString("id", data.getId());
    editor.putString("nik", data.getNik());
    editor.putString("nama", data.getNama());
    editor.putString("gender", data.getGender());
    editor.putString("tgl", data.getTgl());
    editor.putString("alamat", data.getAlamat());
    editor.putString("pro", data.getPro());
    editor.putString("kab", data.getKab());
    editor.putString("kec", data.getKec());
    editor.putString("desa", data.getDesa());
    editor.putString("kontak", data.getKontak());
    editor.putString("foto", data.getFoto());

```

```

        editor.putString("user", data.getUser());
        editor.putString("pass", data.getPass());
        editor.putString("status", data.getStatus());
        editor.commit();
    } else {
        Toast.makeText(getActivity(), message,
            Toast.LENGTH_SHORT) .

```

### Segmen Program 4.1 Profile(lanjutan)

```

} catch (Exception e){
    Toast.makeText(getActivity(), ""+e.toString(),
        Toast.LENGTH_SHORT) .show();
}

```

#### 4.3.5. Halaman Produk

pada menu produk pengguna akan disuguhkan dengan tampilan berbagai produk dari semua vendor pada menu produk pengguna akan disuguhkan dengan tampilan berbagai produk dari semua vendor, Tampilan halaman produk terlihat pada gambar 4.27.

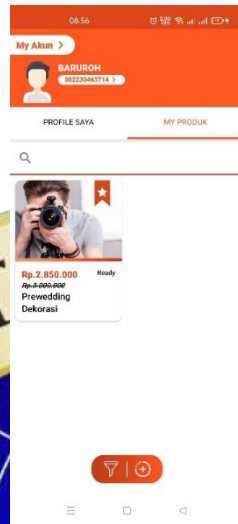


Gambar 4.28 Halaman Produk

#### 4.3.6. Halaman Produk Vendor

Pada menu produk vendor terdapat tampilan produk- produk dari vendor tersebut yang sudah di posting oleh vendor Tampilan halaman *dashboard* terlihat pada gambar 4.28.





**Gambar 4.29** Halaman Produk Vendor

#### 4.3.7. Halaman *Upload* Produk

Pada halaman produk wedding akan ditampilkan semua produk dari semua vendor. Tampilan halaman produk wedding terlihat pada gambar 4.29.



**Gambar 4.30** Halaman Produk

#### Segmen program 4.2. Upload Produk

```

public void showProduk(int animationSource) {
    View dialogView =

```

### Segmen program 4.2. Upload Produk(lanjutan)

```

        View.inflate(getActivity(), R.layout.formbarang,
null);
dialogproduk = new Dialog(getActivity(),
R.style.DialogTheme);
dialogproduk.requestWindowFeature(Window.FEATURE_NO_TITLE
);

dialogproduk.setContentView(dialogView);
img1 = dialogproduk.findViewById(R.id.img1);
img2 = dialogproduk.findViewById(R.id.img2);
img3 = dialogproduk.findViewById(R.id.img3);
img4 = dialogproduk.findViewById(R.id.img4);
img5 = dialogproduk.findViewById(R.id.img5);
delimg1 = dialogproduk.findViewById(R.id.delimg1);
delimg2 = dialogproduk.findViewById(R.id.delimg2);
delimg3 = dialogproduk.findViewById(R.id.delimg3);
delimg4 = dialogproduk.findViewById(R.id.delimg4);
delimg5 = dialogproduk.findViewById(R.id.delimg5);
tkat = dialogproduk.findViewById(R.id.tkat);
tnama = dialogproduk.findViewById(R.id.tnama);
tharga = dialogproduk.findViewById(R.id.tharga);
tdes = dialogproduk.findViewById(R.id.tdes);
bsave = dialogproduk.findViewById(R.id.bsave);
tpromo = dialogproduk.findViewById(R.id.tpromo);
rnon = dialogproduk.findViewById(R.id.rnon);
rpromo = dialogproduk.findViewById(R.id.rpromo);
viewpromo =
dialogproduk.findViewById(R.id.viewpromo);
clickCamera(img1, PIC_CAMERA1, PIC_GALERY1)

```

### Segmen program 4.2. Upload Produk(lanjutan)

```

clickCamera(img2, PIC_CAMERA2, PIC_GALERY2);
clickCamera(img3, PIC_CAMERA3, PIC_GALERY3);
clickCamera(img4, PIC_CAMERA4, PIC_GALERY4);
clickCamera(img5, PIC_CAMERA5, PIC_GALERY5);
viewKat();

tharga.addTextChangedListener(new
NumberTextWatcher(tharga));

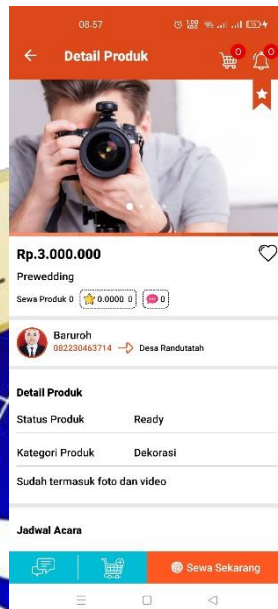
tpromo.addTextChangedListener(new

```

NumberTextWatcher (tpromo) );

#### 4.3.8. Halaman Detail Produk

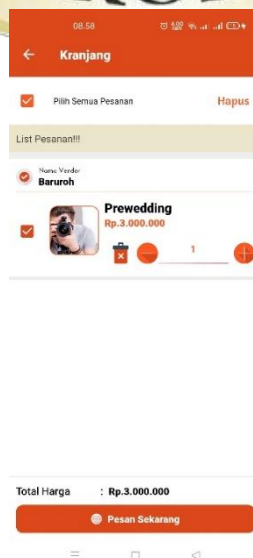
Pada halaman detail produk akan ditampilkan kejelasan dari setiap produk yang dimiliki vendor. Tampilan halaman detail transaksi terlihat pada gambar 4.30.



Gambar 4.31 Halaman Detail Produk

#### 4.3.9. Halaman Keranjang Pembelian

Pada halaman keranjang pembelian user akan ditampilkan dengan tampilan produk yang akan dipesan. Tampilan halaman keranjang pembelian terlihat pada gambar 4.31.



**Gambar 4.32** Halaman Keranjang Pembelian

#### 4.3.10. Halaman Buat Pesanan

Pada halaman buat pesanan user akan ditampilkan dengan tampilan pesanannya yang mana telah sesuai dengan alamat pemesan. Tampilan halaman buat pesanan terlihat pada gambar 4.32.



**Gambar 4.33.** Halaman Buat Pesanan

#### Segmen program 4.3. Buat Pesanan

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.mpesanan);
```

#### Segmen program 4.3. Buat Pesanan (lanjutan)

```
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
    setSupportActionBar(toolbar);  
    // add back arrow to toolbar  
    if (getSupportActionBar() != null) {  
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);  
        getSupportActionBar().setDisplayShowHomeEnabled(true);  
    }  
    getSupportActionBar().setDisplayShowTitleEnabled(false);  
    talamat = findViewById(R.id.talamat);  
    tttotal = findViewById(R.id.tttotal);  
    ttgl = findViewById(R.id.ttgl);  
    tpro = findViewById(R.id.tpro);  
    tttotalbayar = findViewById(R.id.tttotalbayar);
```

```

ttgl.setOnClickListener(new View.OnClickListener()
    @Override
    public void onClick(View v) {
        Calendar newCalendar = Calendar.getInstance();

        DatePickerDialog datePickerDialog = new
        DatePickerDialog(pesanan.this, new
        DatePickerDialog.OnDateSetListener() {
            @Override
                public void
                onDateSet(DatePicker view, int year, int monthOfYear,
                int dayOfMonth) {
                    Calendar newDate = Calendar.getInstance();
                    newDate.set(year, monthOfYear, dayOfMonth);
                    SimpleDateFormat dateFormatter = new
                    SimpleDateFormat("dd/MM/yyyy", Locale.US);
                    ttgl.setText(dateFormatter.format(newDate.getTime()));
                }
            });
    }
}

```

Segmen program 4.3. Buat Pesanan(lanjutan)

### Segmen program 4.3. Buat Pesanan(lanjutan)

```

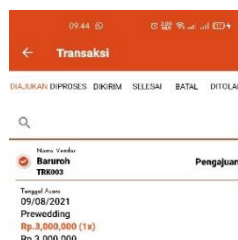
        }, newCalendar.get(Calendar.YEAR),
        newCalendar.get(Calendar.MONTH),
        newCalendar.get(Calendar.DAY_OF_MONTH));

        datePickerDialog.show();
    }
});

```

#### 4.3.11. Halaman Transaksi Pelanggan

Pada menu transaksi pelanggan bisa membayar sesuai harga yang telah tertera pada harga produk vendo dengan menggunakan dp atau tidak sebagaimana gambar berikut. Tampilan halaman transaksi pelanggan terlihat pada gambar 4.34.



**Gambar 4.34** Halaman Transaksi

**Segmen program 4.4** Transaksi

@Override

```
public void onStart() {  
    super.onStart();  
}
```

```
private class ViewPagerAdapter extends  
FragmentPagerAdapter {
```

```
    private final List<Fragment>
```

**Segmen program 4.4** Transaksi

```
public ViewPagerAdapter(FragmentManager manager)  
{  
    super(manager);  
}  
  
@Override  
public Fragment getItem(int position) {  
    return mFragmentManager.get(position);  
}  
  
@Override  
public int getCount() {  
    return mFragmentManager.size();  
}  
  
public void addFragment(Fragment fragment) {  
    mFragmentManager.add(fragment);  
}  
  
@Override  
public CharSequence getPageTitle(int position) {  
    return null;  
}
```

#### **4.3.12. Halaman Persetujuan Acara**

Pada halaman persetujuan acara vendor akan memverifikasi bahwa pada tanggal tersebut sedang tidak ada jadwal. Tampilan halaman transaksi vendor terlihat pada gambar 4.33.



Gambar 4.36 Halaman Persetujuan Vendor

#### 4.4. Uji Coba

Pada penelitian ini terdapat dua proses tes uji sistem. Adapun dua proses tes uji sistem tersebut di antaranya adalah *Testing Internal* dan *Testing External*. Pihak yang menangani *Testing Internal* dan *Testing External* tidak lain adalah dari pihak *Users*. Berikut merupakan perolehan hasil dari uji coba dengan keterangan sebagai berikut:

##### 4.4.1. Hasil *Internal Testing*

Tabel 4.8. Draft *Internal Testing* (Black Box)

No	Event	Hasil Yang Diharapkan	Pernyataan	
			Ya	Tidak
1	Apakah proses login sesuai dengan kebutuhan vendor dan user?	Proses login sesuai dengan kebutuhan	√	

2	Apakah proses register sesuai dengan kebutuhan vendor dan user?	Proses register sesuai dengan kebutuhan	✓	
3	Apakah mengupload produk vendor sesuai dengan kebutuhan?	Mengupload produk vendor sesuai dengan kebutuhan	✓	
4	Apakah proses pemesanan produk sesuai dengan kebutuhan?	Proses pemesanan produk sesuai dengan kebutuhan	✓	

**Tabel 4.8.** Draft *Internal Testing (Black Box)* (lanjutan)

5	Apakah proses pemberitahuan promo sesuai dengan kebutuhan?	Proses pemberitahuan promo sesuai dengan kebutuhan	✓	
6	Apakah proses transaksi pelanggan sesuai dengan kebutuhan?	Proses transaksi pelanggan sesuai dengan kebutuhan		
7	Apakah proses transaksi vendor sesuai dengan kebutuhan?	Proses transaksi vendor sesuai dengan kebutuhan	✓	
8	Apakah proses persetujuan acara vendor sesuai dengan kebutuhan?	Proses persetujuan acara sesuai dengan kebutuhan	✓	

#### 4.4.2. Hasil *External Testing*



**Tabel 4.9** Draft Hasil *External Testing*

No	Draf Pertanyaan	Jawaban				
		SB	B	C B	TB	STB
1.	Apakah dengan adanya aplikasi ini pihak WO serta masyarakat lebih mudah dalam penyewaan WO?	4	1			
2	Apakah fitur-fitur yang digunakan telah sesuai dengan kebutuhan?	2	3			

No	Skala Jawaban	Skor
1	Sangat Baik (SB)	5 x 11 = 55
2	Baik (B)	4 x 14 = 56
3	Cukup (C)	3 x 0 = 0
4	Tidak Baik (TB)	2 x 0 = 0
5	Sangat Tidak Baik (STB)	1 x 0 = 0
		Total = 111

**Tabel 4.9** Draft Hasil *External Testing* (lanjutan)

3	Apakah sistem ini membutuhkan fitur-fitur baru untuk mendukung keadaan sistem?		5			
4	Apakah sistem ini telah memenuhi standart keamanan?	1	4			
5	Apakah sistem ini telah layak digunakan?	4	1			

**Tabel 4.10** Perhitungan skala likert

Skor tertinggi(Y) = skor tertinggi x jumlah responden Y

$$= 5 \times 5 = 25$$

Skor terendah(X) = skor terendah x jumlah respondenX

$$= 1 \times 5 = 5$$

Jumlah skor ideal = skor tertinggi x Jumlah soal x Jumlah responden

$$5 \times 5 \times 5 = 125$$

Berikut interval dari terendah (0%) hingga tertinggi (100%):

**Tabel 4.11** Kriteria Interpretasi persen

No	Persentase	Keterangan
1	0% - 19,99%	Sangat Tidak Baik
2	20% - 39,99%	Tidak Baik
3	40% - 59,99%	Cukup
4	60% - 79,99%	Baik
5	80% - 100%	Sangat Baik

Rumus penentuan nilai presentase dari kuesioner yang peneliti lakukan adalah Penentuan nilai presentase = (Total Skor yang diperoleh / Skor ideal) x 100%, Sehingga di dapat nilai presentase = (111/125) x 100% = 88%. Berdasarkan tabel presentase nilai diatas dapat diambil kesimpulan bahwa Aplikasi E-marketplace Wedding Organizer Berbasis Android ini sangat baik

#### 4.1 Pemeliharaan (*Maintenance*)

Setelah observasi permasalahan, analisis sistem yang dilanjut dengan pengujian terhadap sistem baru, maka tahap terakhir dalam suatu penelitian adalah melakukan perawatan (*Maintenance*) terhadap sistem tersebut. Hal ini bertujuan untuk mengetahui adanya kesalahan dan kekurangan yang kemungkinan tidak dijumpai sebelumnya. Selain itu juga berguna dalam penambahan fitur baru nantinya sebagai bentuk pengembangan sistem.