

MODEL RAD DALAM PERANCANGAN FRONT END APLIKASI PAYMENT POINT ONLINE BANK (PPOB) BERBASIS ANDROID

Dian Ambar Wasessa¹; Frieyadie²

Sistem Informasi^{1,2}

STMIK Nusa Mandiri^{1,2}

<http://www.nusamandiri.ac.id/>^{1,2}

dian.dbw@nusamandiri.ac.id¹, frieyadie@nusamandiri.ac.id²

Abstract—*The conventional purchase of credit by sending sms to the agent server is often has problem. The error was caused by different format of SMS for each server. This error has an impact on the high cost of using SMS. Another problem when the customer wants to check the electricity bill, the website provided by PLN is less attractive or have to come directly to the PLN payment counter. With these problems, an Android-based application was developed to provide convenience in purchasing credit without worrying about writing format errors and checking until payment of various types of bills becomes more efficient. Bill payments become more practical in one unit, so there is no need to go to different payment counters each month and not get stuck in a long queue that wash time. Development model of this PPOB application is RAD which consists of business modeling, data modeling, process modeling, application development and testing.*

Keywords: PPOB, Electonic Payment, Digital Wallet, RAD.

Intisari—Sistem pembelian pulsa konvensional dengan cara mengirim sms ke server agen sering terjadi kesalahan. Kesalahan ini karena format penulisan tiap - tiap server berbeda sehingga sulit diingat. Kesalahan ini berdampak pada biaya penggunaan SMS yang tinggi. Begitupun dengan pengecekan tagihan listrik, terkendala dengan kurang diminatinya website yang telah disediakan oleh PLN atau harus datang langsung ke loket pembayaran PLN. Dengan permasalahan tersebut maka dikembangkanlah suatu aplikasi berbasis android untuk memberikan kemudahan dalam pembelian pulsa tanpa khawatir terdapat kesalahan format penulisan dan pengecekan sampai pada pembayaran berbagai jenis tagihan menjadi lebih efisien. Pembayaran tagihan menjadi lebih praktis dalam satu kesatuan, sehingga tidak perlu mendatangi beberapa loket pembayaran yang berbeda setiap bulannya dan tidak terjebak dalam antrian panjang yang memakan waktu. Model pengembangan dari aplikasi PPOB ini adalah RAD yang terdiri dari pemodelan bisnis,

pemodelan data, pemodelan proses, pembuatan aplikasi dan pengujian.

Kata Kunci: PPOB, Pembayaran Elektronik, Domet Digital, RAD.

PENDAHULUAN

Dahulu pengisian pulsa dilakukan dengan mengirim sms kepada server agen pulsa menggunakan telepon selular, pengiriman sms harus sesuai dengan format yang sudah ditentukan oleh tiap server. Namun, kesalahan penulisan format sms sering terjadi dalam sistem pengisian pulsa, yang menyebabkan kegagalan dalam proses pengisian pulsa (Kurniawan, Hijriani, & Hartawan, 2015). Kesalahan format transaksi baik dalam menginputkan nominal pulsa dan id tagihan pelanggan sehingga berdampak pada biaya dari setiap transaksi yang membengkak karena melalui sms (Wahyudi & M.T, 2017). Masalah lain yang diangkat yaitu pelanggan PLN sulit untuk mendapatkan informasi mengenai tagihan listriknya (Njatrijani, Widanarti, & Noveliasari, 2016), meskipun sudah disediakan website yang bisa diakses oleh pelanggan. Namun alamat website tersebut kurang populer. Sedangkan fasilitas lainnya dengan sms juga terdapat kendala, seperti format sms yang harus diingat dan biaya dari sms balasan yang dibebankan kepada pelanggan (Zuriati & Widyawati, 2018). Untuk mengatasi beberapa kelemahan tersebut dirasa sangat dibutuhkan sebuah perubahan dalam berbagai aspek, salah satunya dalam bentuk penyebaran informasi tagihan pembayaran sehingga informasi dapat dinilai lebih efektif dan efisien (Rahardja, Aini, & Hardini, 2018). Sehingga masyarakat tidak perlu lagi memeriksa jumlah tagihan melalui website PLN atau datang langsung ke PLN (Agan & Santoso, 2013). Dan masyarakat tidak lagi terjebak pada antrian yang panjang, tentunya memakan waktu yang lama (Wahyudi & M.T, 2017).

Saat ini internet telah berkembang pesat dan memberikan banyak kemudahan bagi dunia bisnis. Salah satunya pada bisnis PPOB (*Payment Point*



Online Bank). Pada dasarnya, PPOB adalah loket pembayaran *online* dengan menggunakan fungsi bank. Berbagai jenis tagihan dapat dibayarkan melalui PPOB ini seperti PDAM, BPJS, PLN, telepon, internet, asuransi, tv kabel, kartu kredit, *multi finance* sampai pada pembelian pulsa prabayar, token listrik dan isi ulang dompet digital. Selain menggunakan fungsi bank, PPOB juga bekerja sama dengan beberapa *biller payment provider* yang menyediakan koneksi *Host-To-Host*.

Layanan PPOB yang dikembangkan berjalan pada perangkat *mobile* berbasis android. Penelitian ini bertujuan untuk memberikan kemudahan dan kenyamanan bagi para pelanggan dalam melakukan berbagai transaksi yang tersedia dalam satu aplikasi. Sehingga diharapkan pelanggan dimana saja dan kapan saja dapat mengetahui informasi mengenai jumlah tagihan baik itu PLN, telepon, *handphone* pasca bayar, BPJS kesehatan, PDAM, TV berlangganan dan internet dengan mudah. Serta langsung dapat melakukan pembayaran terhadap tagihan tersebut. Begitupun dengan pembelian pulsa, paket data, token listrik dan pengisian ulang dompet digital seperti OVO dan Gopay menjadi lebih praktis.

BAHAN DAN METODE

A. Teknik Pengumpulan Data

Untuk mendapatkan data-data yang menunjang, teknik yang dilakukan adalah metode observasi, yaitu pengamatan secara langsung terhadap kegiatan jual - beli pulsa dan beberapa jenis pembayaran tagihan. Juga dilakukan observasi *online* dengan cara membandingkan layanan-layanan PPOB baik dalam bentuk aplikasi *mobile* maupun website yang sudah ada sebelumnya. Selanjutnya metode yang digunakan yaitu studi pustaka dengan cara mengumpulkan referensi yang menunjang penelitian ini melalui buku, jurnal, majalah dan sumber lainnya yang mendukung penelitian.

B. Model Pengembangan Sistem

Penelitian ini menggunakan model pengembangan sistem Rapid Application Development (RAD)(Elyana, Kholil, & Schaduw, 2019). Tahapan dari model RAD ini yaitu :

1. Pemodelan Bisnis

User harus mendaftarkan diri terlebih dahulu menjadi agen ppob dengan mengisi form registrasi. Pada proses registrasi ini, agen akan mendapatkan nomor rekening virtual, yang diperuntukan untuk pengisian deposit agen. Kemudian untuk masuk kedalam aplikasi ppob, agen diwajibkan untuk login dengan mengisi nomor *handphone* dan *password*. Sebelum memulai transaksi pastikan saldo deposit agen sudah terisi. Jika agen ingin melakukan transaksi pembelian pulsa, agen harus

memasukkan nomor *handphone* tujuan pengisian pulsa lalu memilih denom yang diinginkan. Aplikasi akan mengirimkan *request* dengan format JSON ke aplikasi *middle* untuk selanjutnya akan diteruskan ke *switching* tertentu. Aplikasi akan menerima balasan terkait berhasil atau tidaknya transaksi pembelian pulsa tersebut.

Berbeda lagi jika agen akan melakukan pembayaran suatu tagihan. Jika pada pembelian pulsa, *request* hanya dilakukan sekali pada aplikasi *middle* namun untuk pembayaran tagihan *request* dilakukan dua kali. *Request* pertama disebut *inquiry* dan *request* kedua disebut *payment*. Pertama agen diminta untuk memasukkan nomor pelanggan, kemudian besaran tagihan atas nomor pelanggan tersebut dicek melalui *request inquiry*. *Inquiry response* biasanya memuat nama dari pemilik tagihan dan total tagihan yang harus dibayarkan. Setelah itu agen boleh membatalkan atau melanjutkan pembayaran atas tagihannya. Jika agen ingin melanjutkan pembayaran maka aplikasi mengirimkan *request* kedua ke aplikasi *middle*. Selanjutnya aplikasi *middle* akan mengirimkan response sukses atau gagal. terkait berhasil atau gagalnya pembayaran tersebut.

2. Pemodelan Data

Pemodelan data menggunakan *Entity Relationship Diagram*.

3. Pemodelan Proses

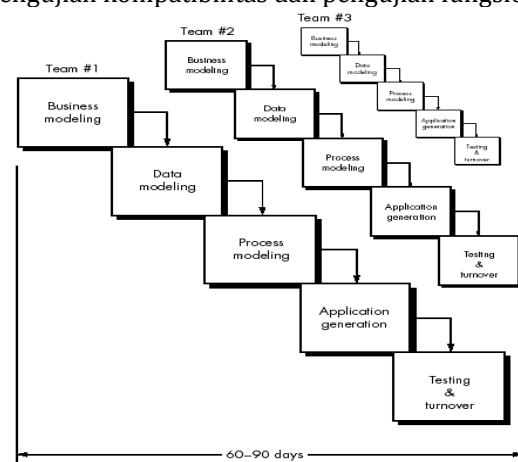
Pemodelan proses dilakukan untuk menjelaskan alur kerja sistem pada transaksi - transaksi yang tersedia pada aplikasi PPOB. Pemodelan proses ini dibuat menggunakan diagram aktivitas.

4. Pembuatan Aplikasi

Bahasa program yang dipilih yaitu java untuk android menggunakan IDE Android Studio. Dan pertukaran data antara aplikasi PPOB ke aplikasi *middle* menggunakan format JSON.

5. Pengujian dan Pergantian

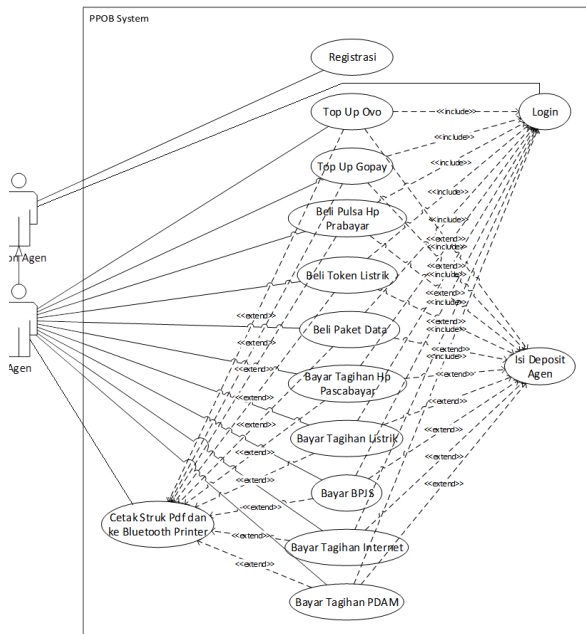
Proses pengujian yang dilakukan adalah pengujian kompatibilitas dan pengujian fungsional.



Sumber: (Pressman, 2003)

**Gambar 1. Model RAD
HASIL DAN PEMBAHASAN**

1. Pemodelan Bisnis



Sumber: (Wasesha & Frieyadie, 2019)
Gambar 2. Usecase diagram aplikasi PPOB

Penjelasan dari gambar 2 adalah sebagai berikut, *User* harus mendaftar sebagai agen PPOB terlebih dahulu. Kemudian agen harus login sebelum melakukan transaksi. Transaksi baru bisa dilakukan jika agen telah mengisi deposit agen. Agen dapat melakukan berbagai transaksi seperti *topup ovo*, *gopay*, beli pulsa *handphone* prabayar, token listrik, paket data, bayar tagihan *handphone* pascabayar, listrik, BPJS, Internet dan PDAM. Setelah melakukan transaksi, agen dapat mencetak struk sebagai bukti pembayaran baik menggunakan *printer bluetooth* maupun disimpan dalam bentuk format pdf.

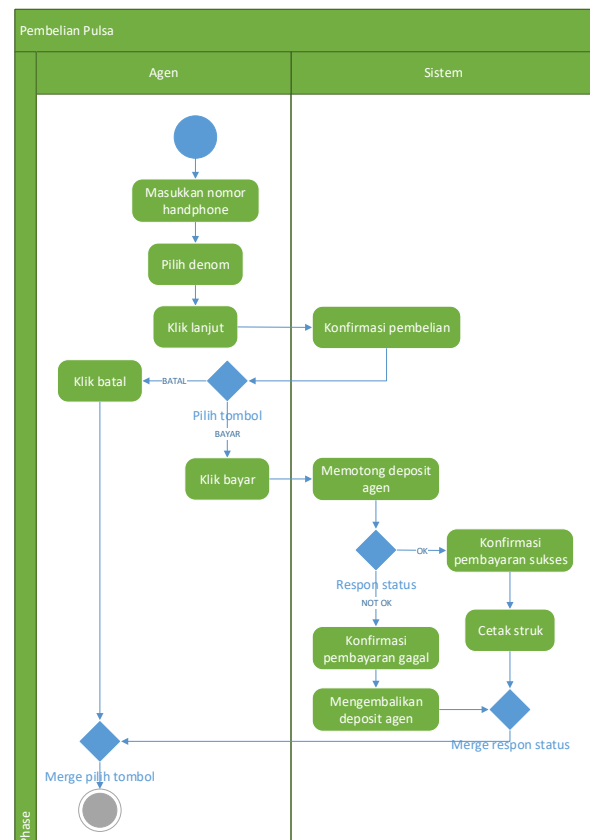
2. Pemodelan Data

Pemodelan data pada sistem PPOB seperti terlihat pada gambar 3. Data agen disimpan pada tabel agen, dimana setiap agen memiliki nomor virtual account BNI yang disimpan pada tabel *bniva*. Setiap agen memiliki level yang berbeda diatur pada tabel *levelagent*. Total Saldo deposit dari tiap-tiap agen disimpan pada tabel *deposit*, deposit ini dapat bertambah jika agen melakukan *topup* sesuai yang tersimpan pada tabel *topupdeposit* atau dapat berkurang jika agen melakukan transaksi seperti yang tersimpan pada tabel *transaksi*. PPOB ini terdiri dari banyak produk yang disimpan pada tabel *produk*, produk tersebut bisa berasal dari mitra-mitra yang berbeda disimpan pada tabel *mitra*. Dan tiap mitra

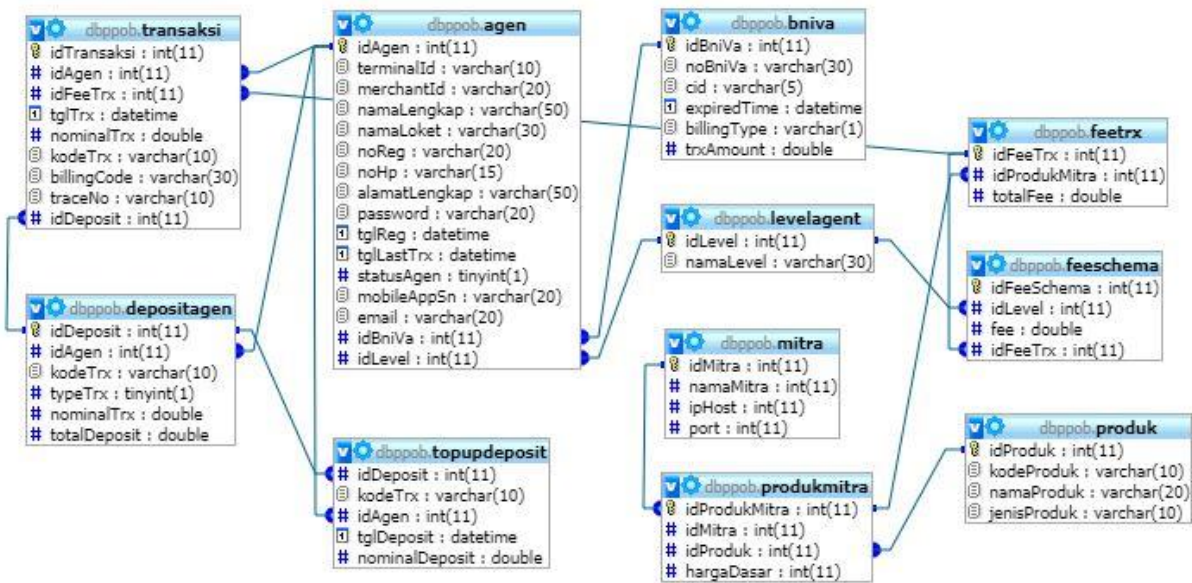
dapat menetapkan harga dasar untuk produknya, maka disimpan ditabel *produkmitra*. Managemen fee antar agen diatur pada tabel *feeSchema* dan *feeTrx*.

3. Pemodelan Proses

Pada transaksi PPOB terdapat dua jenis proses. Ada transaksi yang hanya satu kali melakukan *request* ke aplikasi *middle* kemudian mendapatkan *response* sukses dan transaksi berakhir. Namun pada jenis kedua yaitu transaksi melakukan dua kali *request* dan dua kali mendapat *response*. Untuk *request* pertama biasa disebut dengan *inquiry* dan yang kedua disebut *payment*. *Inquiry request* dilakukan untuk mendapatkan data detail dari nomor id yang dikirim beserta besaran tagihannya. Pada *response inquiry* terdapat tombol untuk melanjutkan atau membatalkan transaksi tersebut. Jika transaksi akan dilanjutkan maka dikirim *request* yang kedua. Transaksi berakhir ketika *response* kedua yang didapat sukses. Pada gambar 4 dijelaskan alur dari transaksi yang hanya membutuhkan satu kali *request* dan satu kali *response* seperti pada transaksi pembelian pulsa prabayar.

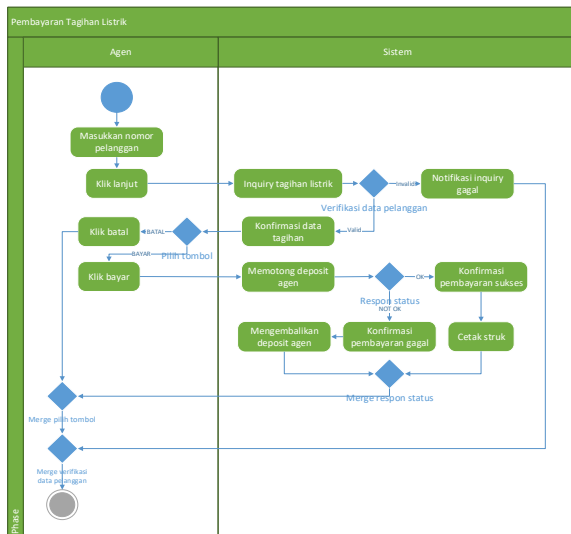


Sumber: (Wasesha & Frieyadie, 2019)
Gambar 4. Activity diagram pembelian pulsa



Sumber: (Wasesha & Frieyadie, 2019)

Gambar 3. Desain Database MySQL



Sumber: (Wasesha & Frieyadie, 2019)

Gambar 5. Activity diagram pengecekan dan pembayaran tagihan listrik

Model proses yang kedua yaitu transaksi yang memerlukan dua kali *request* dan dua kali *response* sebagai contoh untuk pembayaran tagihan listrik. Perbedaannya dengan model proses sebelumnya adalah pada pembayaran tagihan listrik ini ada proses untuk pengecekan data pelanggan dan besaran tagihan tertunggak, sehingga dikirimkan *request inquiry* terlebih dahulu. Setelah mendapatkan *response* sukses, data detail pelanggan akan ditampilkan beserta besaran tagihannya jika pelanggan menyetujui atas rincian tersebut maka proses dilanjutkan dengan pembayaran. Alur sistemnya seperti gambar 5.

4. Pembuatan Aplikasi

Pembuatan aplikasi *front-end* ini menggunakan bahasa pemrograman java untuk android. Setiap *request* yang dikirimkan ke aplikasi *middle* menggunakan API dengan format *request* dan *response* berupa JSON.

a) Halaman Login

Login agen dapat dilakukan dengan memasukkan nomor *handphone* dan *password*. Tampilan loginnya seperti terlihat pada gambar 6.

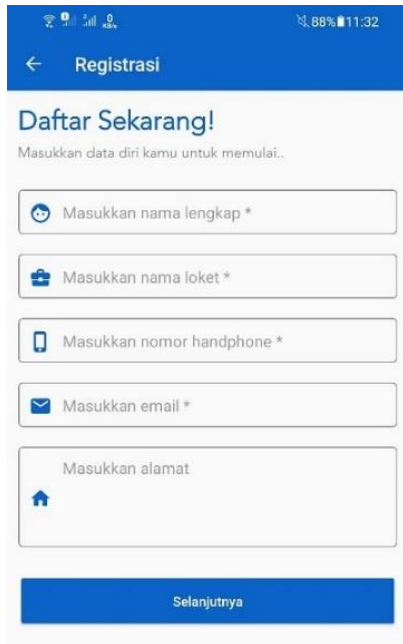


Sumber: (Wasesha & Frieyadie, 2019)

Gambar 6. Halaman login

b) Halaman Pendaftaran Agen

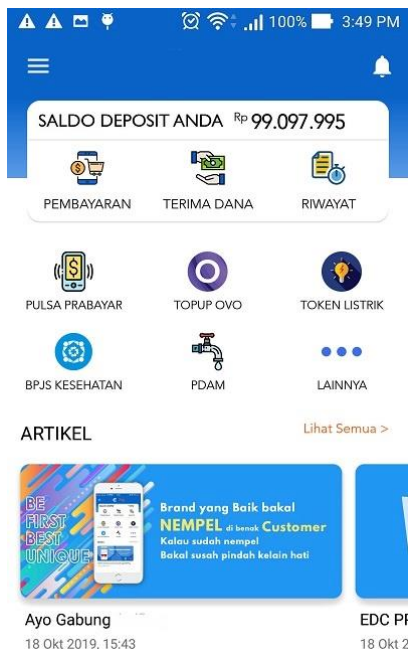
Untuk menjadi agen PPOB maka diharuskan untuk melakukan pendaftaran terlebih dahulu, seperti terlihat pada gambar 7.



Sumber: (Wasesha & Frieyadie, 2019)
Gambar 7. Halaman pendaftaran agen

c) Halaman *Home*

Setelah agen berhasil login maka ia akan dibawa ke halaman *home*. Pada halaman *home* terdapat saldo dari agen, beberapa kategori transaksi dan juga ada daftar artikel. Gambar 8 merupakan tampilan dari halaman home ini.



Sumber: (Wasesha & Frieyadie, 2019)
Gambar 8. Halaman *home*

d) Halaman Semua Kategori

Setelah dari halaman home lalu diklik icon titik tiga (lainnya), maka akan dibawa ke halaman semua kategori. Halaman ini berisi semua kategori

transaksi yang dapat dilakukan oleh agen. Terlihat pada gambar 9.



Sumber: (Wasesha & Frieyadie, 2019)
Gambar 9. Halaman semua kategori

e) Halaman Pembelian Pulsa

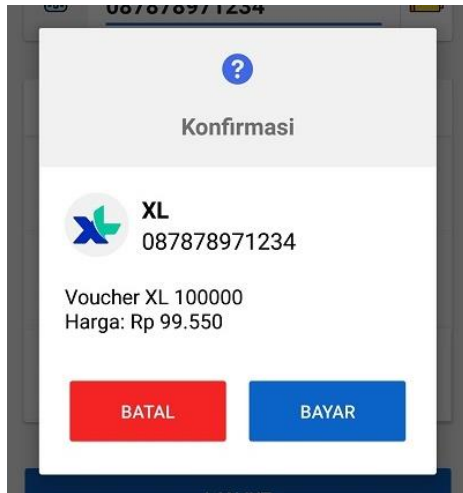
Halaman ini tampil setelah agen memilih kategori "PULSA PRABAYAR". Untuk memulai transaksi isi terlebih dahulu nomor *handphone* yang ingin dituju baik dilakukan dengan manual maupun dengan mengklik *icon* kontak disebelah kanan dari nomor *handphone*. Setelah itu agen harus memilih denom yang diinginkan kemudian menekan tombol "LANJUT". Seperti yang ditampilkan pada gambar 10.



Sumber: (Wasesha & Frieyadie, 2019)
Gambar 10. Halaman pembelian pulsa

f) Notifikasi Konfirmasi Pembelian

Notifikasi ini tampil untuk memastikan kembali apakah agen akan membeli pulsa untuk nomor *handphone* dan dengan denom tersebut. Jika agen sudah yakin untuk melanjutkan transaksi ini maka ia akan menekan tombol "BAYAR". Notifikasi yang ditampilkan seperti gambar 11.



Sumber: (Wasesha & Frieyadie, 2019)

Gambar 11. Halaman konfirmasi pembelian

Pada gambar 11, ketika tombol "BAYAR" ditekan maka aplikasi akan mengirimkan request pada aplikasi *middle* berupa format JSON seperti pada gambar 12.

```
{  
  "amount": "100000",  
  "billing_code": "087878123456",  
  "method": "1141",  
  "processing_code": "299",  
  "product": "XL",  
  "reg_no": "00100000000913",  
  "traceNo": "000010",  
  "trxdatetime": "2019-11-13 16:44:35"  
}
```

Sumber: (Wasesha & Frieyadie, 2019)

Gambar 12. Json request dari pembelian pulsa

Penjelasan gambar 12 seperti berikut, *billing_code* adalah nomor *handphone* tujuan dari pengisian pulsa, *amount* merupakan denom yang dipilih, *product* adalah jenis provider dari nomor *handphone* yang dimasukkan dan *reg_no* adalah nomor registrasi dari agen.

Kemudian aplikasi *middle* akan menanggapi request tersebut dengan mengirimkan *response* seperti gambar 13. Penjelasan sebagai berikut, *billAmount* adalah jumlah nominal yang dibebankan atas transaksi ini, *customer_data* adalah data yang nanti akan di *parsing* lagi oleh

aplikasi *front-end* untuk ditampilkan pada halaman pembelian pulsa berhasil, terdapat juga saldo awal agen sebelum melakukan transaksi dan saldo akhir setelah melakukan transaksi ini.

```
{  
  "status": "00",  
  "description": "SUCCESS",  
  "data": [  
    {  
      "amount": "100000",  
      "billAmount": 99550,  
      "billingCode": "087878123456",  
      "customer_data": "131119164436_XL_087878123456_100000_X100-087878123456",  
      "feeKoordinator": 50,  
      "feeOwner": 300,  
      "feePartner": 150,  
      "frontend": "Mobileapp",  
      "namaKoordinator": "ANIPAY",  
      "namaPartner": "ANIPAY",  
      "noreg": "00100000000913",  
      "procCode": "299",  
      "produk": "XL",  
      "saldoAkhir": 9855547,  
      "saldoAwal": 98655097,  
      "tidTransaction": "00",  
      "traceNo": "000010",  
      "trxid": "A00191113164436"  
    }  
  ],  
  "start": "",  
  "total": "",  
  "session": ""  
}
```

Sumber: (Wasesha & Frieyadie, 2019)

Gambar 13. Json response dari pembelian pulsa

g) Halaman Pembelian Pulsa Berhasil

Jika transaksi pembelian pulsa berhasil dilakukan maka aplikasi akan menampilkan halaman seperti pada gambar 14.



Sumber: (Wasesha & Frieyadie, 2019)

Gambar 14. Halaman pembelian pulsa berhasil

h) Halaman Struk Pulsa Format Pdf

Struk pembelian pulsa dapat dicetak langsung dengan *printer bluetooth* atau dikonversi kedalam format pdf seperti pada gambar 15.



Sumber: (Wasesha & Frieyadie, 2019)
Gambar 15. Halaman struk pulsa format pdf

5. Pengujian dan Pergantian

Pengujian yang dilakukan terdiri dari dua tahap yaitu pengujian kompatibilitas dan pengujian fungsional.

a) Pengujian kompatibilitas

Aplikasi ini diinstal ke beberapa perangkat android dengan spesifikasi yang berbeda untuk memastikan aplikasi dapat berjalan dengan baik. Berikut tabel hasil pengujian kompatibilitas:

Tabel 1. Hasil pengujian kompatibilitas

	Test 1	Test 2	Test 3	Test 4
Merk	Asus Zenfone 5	Asus Zenfone 2	Xiaomi Redmi 5 plus	Samsung Galaxy J7 Pro
CPU	SDM636	Intel Atom Z3580	SDM625	Exynos 7870
RAM	4 GB	4 GB	3 GB	3 GB
Layar	6.2 in	5.5 in	5.9 in	5.5 in
Versi Android	9	5	7	9
Hasil	Berjalan dengan baik	Berjalan dengan baik	Berjalan dengan baik	Berjalan dengan baik

Sumber: (Wasesha & Frieyadie, 2019)

b) Pengujian fungsional

Pengujian dilakukan pada setiap fitur yang ada, yang mencakup login, pendaftaran, lupa password, semua transaksi, percetakan struk sampai historikal dari transaksi-transaksi tersebut. Langkah-langkah pengujian ini dengan memberikan beberapa *input positif* ataupun *negatif* yang bertujuan untuk memastikan *output* dari aplikasi sudah sesuai dengan yang diharapkan. Tabel 2 adalah hasil dari pengujian halaman pendaftaran:

Tabel 2. Script test case halaman pendaftaran

No	Skenario test	Hasil yang diharapkan	Hasil aktual
1	Nama lengkap dikosongkan lalu tekan tombol "Selanjutnya".	Kolom nama lengkap berwarna merah dan terdapat notifikasi "nama lengkap harus diisi"	Sesuai
2	Input nama lengkap kurang dari 3 karakter. Lalu tekan tombol "Selanjutnya". Contoh data: Di	Kolom nama lengkap berwarna merah dan terdapat notifikasi "nama lengkap kurang dari tiga karakter"	Sesuai
3	Input nama lengkap lebih dari 3 karakter. Lalu tekan tombol "Selanjutnya". Contoh data: Dian	Pada kolom nama lengkap tidak berwarna merah dan tidak terdapat notifikasi <i>error</i> , namun pada kolom loket berwarna merah dan terdapat notifikasi "nama loket harus diisi".	Sesuai
4	Input loket, lalu tekan tombol "Selanjutnya". Contoh data: Dian Cellular	Kolom loket tidak berwarna merah dan tidak terdapat notifikasi <i>error</i> namun pada kolom nomor <i>handphone</i> berwarna merah dan terdapat notifikasi "nomor	Sesuai



		<i>handphone</i> harus diisi”.		
5	<i>Input</i> nomor <i>handphone</i> , lalu tekan tombol “Selanjutnya”. Contoh data: 087878123456	Pada kolom nomor <i>handphone</i> tidak berwarna merah dan tidak terdapat notif <i>error</i> namun pada kolom email berwarna merah dan terdapat notif <i>error</i> “email harus diisi”.	Sesuai	
6	<i>Input</i> email tidak sesuai format, lalu tekan tombol “Selanjutnya”. Contoh data: dian@y	Kolom email berwarna merah dan terdapat notif “format email salah”	Sesuai	
7	<i>Input</i> email dengan format yang benar, lalu tekan tombol “Selanjutnya”. Contoh data: dian@yahoo.com	Tampil halaman kode verifikasi.	Sesuai	

Sumber: (Wasesha & Friyadie, 2019)

KESIMPULAN

Dengan adanya aplikasi PPOB berbasis android ini siapapun dapat melakukan transaksi pembelian pulsa dengan mudah. Tidak ada lagi kesalahan dalam penulisan format sms untuk mengirimkan *request* ke server agen karena pengguna hanya tinggal memasukkan nomor *handphone* yang dituju saja. Keuntungan lainnya bagi pelanggan PLN adalah dapat kapan saja mengecek besaran tagihan listriknya dan melakukan pembayaran terhadap tagihan tersebut. Berbagai tagihan dapat dibayarkan pada satu aplikasi, tidak perlu lagi mendatangi masing-masing loket pembayaran yang berbeda setiap bulannya. Aplikasi PPOB ini merupakan cara pembayaran yang lebih aman, efektif dan efisien karena membayar berbagai jenis tagihan dapat dilakukan sendiri melalui *smartphone*.

REFERENSI

Agan, L., & Santoso, P. (2013). Pembuatan Aplikasi Cek Tagihan Listrik Berbasis Android. *Jurnal Dimensi Teknik Elektro*, 1(1), 24–28.

Elyana, I., Kholil, I., & Schadu, F. E. (2019). RANCANG BANGUN SISTEM INFORMASI MANAJEMEN DISPOSISI SURAT MENYURAT DENGAN MENGGUNAKAN MODEL RAD (RAPID APPLICATION DEVELOPMENT). *Jurnal Riset Informatika*, 1(2), 107–112. <https://doi.org/10.34288/JRI.V1I2.37>

Kurniawan, D., Hijriani, A., & Hartawan, F. T. (2015). Rancang Bangun Aplikasi Agen Pulsa Elektronik Bebas Android (Studi Kasus Distributor Pulsa Maxrefil). *Komputasi*, 3(2), 144–153.

Njatrijani, R., Widanarti, H., & Noveliasari, N. (2016). Perlindungan Hukum Bagi Konsumen terhadap Penggunaan Jasa Listrik Pascabayar dan Jasa Listrik Prabayar pada PT. Pln (Persero) di Kota Semarang. *Diponegoro Law Review*, 5(3). Retrieved from <https://ejournal3.undip.ac.id/index.php/dlr/article/view/12120>

Pressman, R. S. (2003). *Software Engineering A Practitioner’s Approach Fifth Edition*. Singapore: Mc Graw Hill.

Rahardja, U., Aini, Q., & Hardini, M. (2018). PENERAPAN SOFTWARE AKUNTANSI ONLINE SEBAGAI PENUNJANG PENCATATAN LAPORAN KEUANGAN. *SISFOTENIKA*, 8(2), 176–187.

Wahyudi, A. M., & M.T, R. (2017). *Aplikasi Payment Point Online Bank (PPOB) Menggunakan Web Service di PT. Sukses Mitra Mandiri Berbasis Android*. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*.

Wasesha, D. A., & Friyadie, F. (2019). *Laporan Akhir Penelitian Mandiri*. Jakarta.

Zuriati, Z., & Widyawati, D. K. (2018). Android based Agriculture products price information system. *Jurnal Ilmiah ESAI*, 8(2), 173–183. <https://doi.org/10.25181/ESAI.V8I2.963>