

IMPLEMENTASI BLOC PATTERN PADA PENGEMBANGAN FRONT-END TOP UP LINKAJA UNTUK APLIKASI AGEN46

Farah Dila Anastasia
Jurusan Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
18523274@students.uii.ac.id

Irving V Papatungan
Jurusan Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
irving@uui.ac.id

Abstrak—PT. Bank Negara Indonesia Tbk (BNI) merupakan perusahaan yang bergerak di bidang layanan finansial yang saat ini sedang mengembangkan aplikasi Agen46 untuk layanan m-banking berbasis multiplatform. Salah satu fasilitas yang dikembangkan adalah layanan pembayaran non tunai berupa top up uang elektronik (e-money). Framework yang digunakan adalah Flutter. Framework ini dapat membuat aplikasi mobile Android maupun iOS hanya dengan satu basis coding (codebase) dalam sekali waktu. Bahasa pemrograman yang digunakan adalah Dart. Pola desain yang diterapkan adalah BLoC Pattern. BLoC Pattern memudahkan developer untuk fokus dalam melakukan konversi event menjadi state sehingga jika implementasi diubah, maka developer tidak perlu lagi melakukan banyak perubahan pada code lainnya. Code yang dihasilkan menjadi lebih mudah dimengerti dan reusable pada pengembangan selanjutnya sehingga dapat meminimalisir resource. Metode pengembangan yang digunakan adalah Agile Scrum dengan beberapa tahapan dari mulai Intialization, Analysis & Design, Development, Testing & Evaluation hingga Deployment. Berdasarkan hasil yang didapat, diketahui bahwa implementasi BLoC Pattern pada pengembangan frontend dapat mempermudah developer mengembangkan aplikasi mobile banking berbasis multiplatform. Terbukti dengan pengembangan dan pengujian yang telah dilakukan menghasilkan fitur yang dapat berjalan dalam dua sistem operasi yaitu Android maupun iOS serta fungsionalitas dari fitur tersebut dapat berjalan dengan baik.

Kata Kunci—*m-banking, top up, e-money, BLoC, flutter*

I. PENDAHULUAN

Pada era globalisasi zaman ini pertumbuhan teknologi sangat pesat dengan munculnya berbagai inovasi. Adanya teknologi telah membawa dampak positif bagi masyarakat terutama dalam penggunaan internet. Penggunaan internet dinilai dapat membantu aktivitas sehari-hari sehingga menjadi kebutuhan primer bagi masyarakat. Perkembangan internet juga berpengaruh pada sektor ekonomi khususnya dalam industri perbankan. Bank menjadi salah satu sektor penting dalam menjaga kestabilan ekonomi di Indonesia. Dalam menjalankan tugasnya bank berkewajiban memberikan dan menjamin layanan terbaik kepada nasabahnya. Layanan yang disediakan oleh bank tentu harus bisa menyesuaikan dengan mobilitas konsumennya sehingga diperlukan pemanfaatan teknologi yang tepat untuk mendukung perkembangan bisnis. Penerapan teknologi yang tepat juga perlu memperhatikan minimum biaya yang

dikeluarkan untuk meminimalisir operasional agar bank dapat beroperasi secara lebih optimal.

Mobile banking merupakan inovasi terbaru untuk diterapkan di industri perbankan (internet banking) [1]. Mobile banking memungkinkan nasabah bank untuk mengakses layanan perbankan dengan menggunakan ponsel mereka untuk melakukan fungsi seperti memeriksa saldo rekening mereka, mengajukan permohonan kartu debit, melakukan transaksi pembayaran, membayar tagihan elektronik, dan lain-lain [2]. Mobile banking berguna untuk berkomunikasi dengan bank melalui perangkat mobile, seperti ponsel atau perangkat elektronik untuk menawarkan layanan mobile banking kepada nasabahnya [3]. Layanan ini memberikan keuntungan bagi pihak nasabah maupun bank. Nasabah akan mendapatkan informasi secara efisien, efektif, dan cepat dalam bertransaksi sehingga mengurangi waktu tunggu jika dibandingkan dengan transaksi konvensional di bank secara langsung. Layanan m-banking dapat menjadi peluang bagi bank untuk menawarkan nilai tambah kepada nasabah. Keefektifan dan keefisienan penggunaan layanan m-banking tidak terlepas dari dukungan telepon seluler dan internet. Pengguna platform layanan mobile banking dapat memperoleh informasi tentang rekening mereka secara real time dan melakukan pembayaran untuk transaksi kapanpun dan dimanapun mereka pilih [4].

Adanya peningkatan kebutuhan transaksi masyarakat di era globalisasi saat ini tentu berdampak pada peningkatan penggunaan uang elektronik. Hal ini dilandasi sejak munculnya aturan yang dikeluarkan oleh Bank Indonesia dalam peraturan No. 26/6/PBI/2018 mengenai uang elektronik. Pada Pasal 1 ayat 3 dijelaskan bahwa uang elektronik adalah alat pembayaran yang diterbitkan atas dasar nilai uang yang disetor terlebih dahulu oleh pemegang kepada penerbit. Sistem pembayaran jumlah atau nilai uangnya diserahkan terlebih dahulu untuk melakukan pengisian nominal uang (top up) kepada penerbit baik melalui agen penerbit atau secara langsung. E-money merupakan salah satu inovasi sistem pembayaran yang muncul dari dampak perkembangan teknologi. E-money adalah alat yang digunakan untuk transaksi non tunai yang dilakukan melalui sistem elektronik [5].

Penggunaan uang elektronik memiliki kelebihan jika dibandingkan dengan uang tunai. Pengguna tidak perlu membawa banyak uang tunai untuk melakukan pembayaran. Transaksi yang dilakukan juga lebih cepat dan efisien. Namun e-money juga memiliki kekurangan yaitu masyarakat belum sepenuhnya menggunakan e-money sebagai sistem

pembayaran. Industri perbankan dan telekomunikasi juga berlomba-lomba mengeluarkan produk e-money [6]. Tindakan ini juga didukung oleh para pelaku fintech startup di Indonesia yang bergerak di sektor pembayaran seperti DANA, LinkAja, GoPay, ShopeePay, iSaku dan masih banyak lainnya. E-money dapat digunakan untuk melakukan transaksi pembayaran di internet maupun pada merchant yang telah bekerjasama dengan bank penerbit e-money.

BNI merupakan salah satu Bank BUMN (Badan Usaha Milik Negara) yang bergerak pada bidang layanan finansial. BNI menyediakan layanan penyimpanan dana maupun fasilitas pinjaman baik pada segmen koperasi, menengah maupun kecil. Beberapa produk dan layanan terbaik disesuaikan dengan kebutuhan nasabah sejak kecil, remaja, dewasa, hingga pensiun. BNI melayani layanan perbankan personal dimulai dari simpanan, pinjaman, hingga layanan kartu kredit. Adapun beberapa layanan dan produk e-banking milik BNI seperti BNI ATM, BNI SMS Banking, BNI Internet Banking, BNI Phone Banking, BNI Mobile Banking, Layanan Gerak, BNI Agen46, Tapcash, BNI Debit Online, BNI SMS Notifikasi, BNI IPay, BNI Smartpay, EDC BNI, Buka Rekening Tabungan Digital, QRIS, dan Internet Payment Gateway. Tidak hanya itu BNI juga melayani solusi pendanaan untuk perbankan bisnis dari tingkat UKM hingga transaksi bisnis internasional. Dalam menjalankan tugasnya BNI didukung oleh sejumlah anak perusahaan yakni BNI Syariah, BNI Multifinance, BNI Sekuritas, BNI Life Insurance dan BNI Remittance [7].

BNI Agen46 adalah salah satu layanan yang tersedia di BNI. BNI Agen46 merupakan mitra BNI (perorangan atau badan hukum yang telah bekerjasama dengan BNI) untuk menyediakan layanan perbankan kepada masyarakat (Layanan Laku Pandai, Layanan LKD, dan Layanan e-Payment) [8]. Meningkatnya kebutuhan sistem transaksi secara digital BNI membutuhkan pengembangan sistem yang baru pada aplikasi m-banking nya. Salah satu proyek yang dikembangkan adalah m-banking Agen46 berbasis multiplatform. Pengembangan yang dilakukan tentu harus memperhatikan biaya yang dikeluarkan untuk proses pemeliharaan sistem yang berjalan. Hal ini bertujuan agar sistem dapat memenuhi kebutuhan penggunaannya dan menjangkau user yang lebih banyak lagi. Proyek Agen46 merupakan proyek yang masih berjalan pada masa pengembangan. Pengembangan dibuat dengan menerapkan desain interface yang baru dari aplikasi sebelumnya sehingga terdapat perubahan yang cukup besar dari segi tampilan. Pembuatan desain ulang aplikasi ini diharapkan dapat menjadi aplikasi yang lebih mudah dipahami pengguna dan menghasilkan tampilan aplikasi yang lebih menarik.

Perbedaan sistem operasi setiap pengguna mendorong BNI untuk memilih framework yang dapat membantu pengembangan secara efisien dari segi biaya pengembangan. Framework yang digunakan adalah Flutter. Penggunaan framework ini bertujuan agar aplikasi mobile banking nantinya dapat berjalan pada dua sistem operasi baik iOS maupun Android untuk layanan keagenan. Layanan keagenan yang dikembangkan yaitu fitur top up uang elektronik (e-money). Pengembangan aplikasi m-banking Agen46 pada penelitian ini lebih fokus pada pengembangan frontend aplikasi dan pengintegrasian API. Salah satu fitur top up e-money yang dibahas pada penelitian ini adalah LinkAja. LinkAja merupakan salah satu penyedia jasa layanan uang elektronik yang bekerjasama dengan BNI.

Aplikasi m-banking Agen46 dikembangkan dengan menggunakan framework Flutter. Dilihat dari sejarahnya bahwa Flutter merupakan framework yang resmi diluncurkan oleh Google pada bulan Desember 2018 [9]. Diketahui bahwa framework Flutter merupakan platform yang digunakan oleh para pengembang untuk membangun aplikasi multiplatform (iOS dan Android) hanya dengan satu basis coding saja [10]. Bahasa yang digunakan dalam framework Flutter adalah Dart. Dart merupakan bahasa pemrograman yang berorientasi objek (OOP) yang dikembangkan oleh Google sejak tahun 2007. Syntax yang dimiliki oleh Dart memiliki kemiripan dengan C++, Java dan Javascript. Penggunaan Dart menjadikan Flutter sebagai cross platform framework tercepat yang memiliki performa seperti native [11]. Dengan kemampuan tersebutlah yang mendukung pengembangan aplikasi mobile secara native untuk kedua platform Android dan iOS. Konsep pengembangan aplikasi yang digunakan pada m-banking adalah BLoC Pattern. BLoC (Business Logic Component) adalah desain pattern yang membantu untuk memisahkan presentation dengan business logic. BLoC Pattern memudahkan developer untuk fokus dalam mengkonversi event menjadi state sehingga saat implementasi dirubah developer tidak perlu lagi melakukan banyak perubahan pada code lainnya. Code yang dibuat menjadi lebih mudah dimengerti dan reusable pada pengembangan sistem operasi yang lain sehingga resource dikeluarkan untuk development Android dan iOS menjadi lebih minim [12]. Penelitian terdahulu yang dilakukan oleh Regawa Rama Prayoga, Ghifari Munawar, Rahil Jumiyani, dan Alifia Syalsabila (2021) dalam penelitiannya berjudul "Performance Analysis of BLoC and Provider State Management Library on Flutter" menjelaskan mengenai performa dari penggunaan BLoC pattern diantaranya adalah memiliki kinerja yang lebih efisien dibandingkan dengan penggunaan setState jika diukur dari penggunaan CPU utilization, penggunaan memory, dan waktu eksekusi. Hal ini karena object widget yang harus dirender saat proses build pada BLoC Pattern memiliki jumlah yang lebih sedikit sehingga prosesnya menjadi lebih ringan. Hal ini menyebabkan alokasi memory yang digunakan menjadi lebih efisien karena jumlah object yang disimpan tentu lebih sedikit. Tidak hanya itu semakin sedikit jumlah widget yang dibuat, semakin singkat proses build, dan waktu render maka waktu eksekusinya menjadi lebih efisien [13].

II. KAJIAN TEORI

A. Mobile Banking

Mobile banking atau m-banking merupakan fasilitas yang memudahkan nasabah untuk melakukan transaksi secara online. Kegiatan transaksi yang biasanya dilakukan dengan cara mendatangi ke bank secara langsung kini dapat dilakukan hanya dengan menggunakan handphone saja. Hal ini memungkinkan nasabah untuk dapat melakukan transaksi dimana saja dan kapan saja. Penggunaan m-banking dimulai dengan melakukan pendaftaran akun terlebih dahulu kemudian nasabah dapat menggunakan untuk melakukan transaksi dengan cara memasukkan user ID atau password dan PIN untuk menyelesaikan transaksi [14]. Nasabah dapat menghemat waktu serta biaya. Selain itu pemanfaatan handphone yang biasanya hanya digunakan sebagai media komunikasi juga bisa dimanfaatkan untuk media berbisnis dan bertransaksi. Layanan yang tersedia pada m-banking memberikan kemudahan kepada nasabah untuk melakukan

transaksi seperti cek saldo, transfer antar rekening, pembayaran tagihan, isi pulsa, dan lain-lain [2]. Selain itu di dalam m-banking juga terdapat laporan atau ringkasan terkait aktivitas keuangan yang dilakukan oleh nasabah. Pada aplikasi m-banking juga terdapat layanan pembelian dan top up dompet digital seperti OVO, Gopay, hingga LinkAja. Layanan - layanan m-banking tersebut tentunya memiliki kelebihan selain efisien dan mudah diakses, nasabah juga dapat melakukan jadwal transaksi keuangan dalam satu waktu sekaligus. Namun selain banyak kelebihan m-banking juga memiliki kekurangan yaitu rentan dengan pencurian data dan membutuhkan jaringan yang bagus untuk dapat mengaksesnya [15].

B. Top Up

Top up adalah sebuah kegiatan untuk melakukan pengisian atau penambahan sejumlah dana ke rekening agar dapat melakukan transaksi seperti pembayaran maupun pembelian. Terdapat juga arti lain dari top up yaitu sebagai pemberian pinjaman dari bank dengan tujuan untuk meningkatkan kredit dari jumlah peminjam. Fasilitas ini diberikan oleh bank agar dapat berhubungan langsung dengan riwayat transaksi nasabah [16]. Tentunya setiap bank memiliki kriteria dan syarat tertentu dalam pemberian fasilitas top up. Kegiatan pengisian ulang saldo bisa dilakukan dengan berbagai metode seperti melalui minimarket, ATM Bank, m-banking atau gerai yang melayani pengisian ulang.

C. E-Money

Uang elektronik merupakan sebuah alat pembayaran dalam bentuk elektronik dimana nilai uangnya disimpan dalam media elektronik tertentu. Pengguna harus menyetorkan uang terlebih dahulu kepada penerbit dan disimpan dalam media elektronik sebelum digunakan untuk keperluan bertransaksi. Nilai uang yang tersimpan di dalam uang elektronik juga akan berkurang sesuai dengan besar nilai transaksi yang dilakukan dan dapat diisi kembali (top up). Media yang digunakan untuk menyimpan e-money berupa chip atau server [17]. Penggunaan e-money ini dinilai sebagai alat pembayaran yang inovatif serta praktis dan diharapkan dapat membantu kelancaran pembayaran kegiatan ekonomi yang bersifat massal, cepat dan mikro. Perkembangan e-money juga diharapkan sebagai alternatif pembayaran non tunai sehingga dapat menjangkau masyarakat yang selama ini belum mempunyai akses kepada sistem perbankan. Adapun beberapa resiko penggunaan dari e-money yaitu dapat hilang atau digunakan oleh orang lain. Hal ini dikarenakan prinsip yang terdapat pada e-money sama dengan uang tunai yang apabila hilang maka tidak dapat diklaim kepada penerbit. Selain itu pengguna yang tidak memahami betul cara pemakaian e-money juga bisa berakibat penempelan dua kali pada reader sehingga besar transaksi yang dilakukan bisa lebih besar dari nilai aslinya. Berdasarkan jenis nya e-money terbagi menjadi dua jenis yaitu e-money registered dan e-money unregistered. E-money registered yaitu uang elektronik yang data identitas pemegangnya tercatat/terdaftar pada penerbit. Dalam hal ini penerbit harus menerapkan prinsip mengenal nasabah dalam menerbitkan uang elektronik registered dan batas maksimum nilai uang elektronik ini sebesar Rp5.000.000 (lima juta rupiah). Sedangkan e-money unregistered merupakan uang elektronik yang data identitas pemegangnya tidak terdaftar pada penerbit uang elektronik dan batas maksimum yang

tersimpan pada uang ini sebesar Rp1.000.000 (satu juta rupiah) [18].

D. Flutter

Flutter dirilis resmi pada bulan Desember 2018 oleh Google. Flutter merupakan SDK untuk mengembangkan aplikasi baik mobile, web maupun desktop [9]. Flutter adalah framework yang membantu developer untuk membuat aplikasi mobile multiplatform. Banyak perusahaan yang telah menggunakan framework ini seperti Alibaba, ebay, hingga Google Ads. Terdapat 2 komponen penting dalam Flutter yaitu:

- Software Development Kit (SDK) adalah alat yang berfungsi untuk membuat aplikasi agar dapat berfungsi sebagai multiplatform.
- Framework UI adalah komponen UI seperti teks, button, navigation dan lainnya yang dapat dikustomisasi [10].

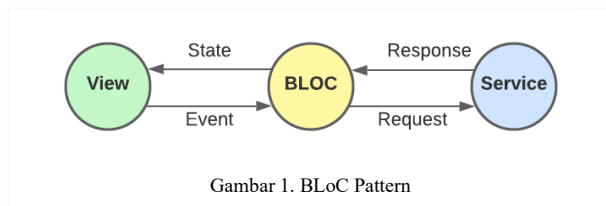
Setiap membuat UI di dalam Flutter akan menciptakan berbagai widget. Beberapa karakteristik dari widget yaitu dapat digunakan untuk membuat UI seperti column, row, text, stack, card, form, padding. Developer juga dapat melakukan styling seperti tipe font, size, color, margin, shadow. Selain itu widget dapat berupa kumpulan bentuk dan formular [9].

Di dalam Flutter terdapat fitur hot-reload. Fitur ini memudahkan developer dalam membangun UI, menambahkan fitur dan memperbaiki bug dengan melihat langsung perubahan UI yang terdapat pada aplikasi. Hot-reload bekerja dengan menanamkan kode file terbaru ke dalam Dart Virtual Machine (VM). Kerangka pada Flutter membuat ulang semua widget secara otomatis setelah VM diperbarui [10]. Fitur ini yang memudahkan developer untuk melihat langsung perubahan UI yang dikembangkan.

Untuk pengembangan aplikasi mobile, Flutter dapat digunakan untuk mengembangkan aplikasi berbasis Android dan iOS dengan menggunakan bahasa Dart. Dart merupakan bahasa pemrograman berbasis class dan berorientasi objek. Di dalam Flutter terdapat dua kombinasi class yang digunakan yaitu: Stateless Widgets dan Stateful Widgets. Class ini merupakan widget yang bertugas menampilkan tampilan user interface dengan membangun widget lainnya agar menjadi lebih terlihat nyata. Stateless Widgets adalah widgets yang bersifat statis dan seluruh konfigurasi yang dimuat sudah diinisialisasi dari awal atau hanya dapat menggambar satu kali saat widget dimuat sehingga tidak dapat menggunakan fitur hot-reload. Sedangkan Stateful widgets adalah widgets yang bersifat dinamis dan dapat berubah pada kondisi tertentu sehingga dapat menggunakan fitur hot-reload [9].

E. BLoC Pattern

Terdapat beberapa penelitian dan pekerjaan sebelumnya yang sudah memanfaatkan BLoC Pattern dalam pengembangan aplikasi. Penelitian terdahulu oleh Muhammad Zulfa Assyfa dengan judul "Implementasi Desain Pattern BLoC dan Pattern Repository pada Aplikasi Energy Management System PT. PLN Berbasis Mobile dengan Flutter" pengembangan yang dilakukan tentang pembuatan perangkat lunak berbasis mobile dengan Flutter yang berfungsi untuk membantu kegiatan monitoring dan



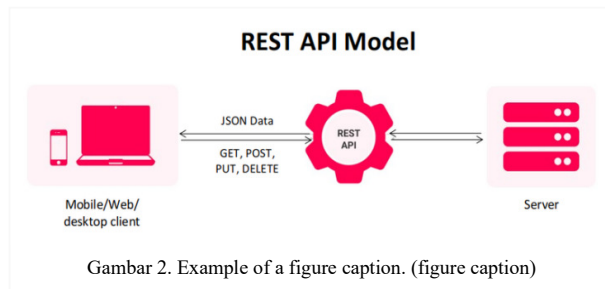
Gambar 1. BLoC Pattern

pembayaran pemakaian listrik PLN Group. Metode desain yang digunakan adalah BLoC Pattern. Penggunaan BLoC Pattern ini diterapkan dalam pembuatan aplikasi EMS. Pada aplikasi EMS hanya terdapat satu aktor yang diberi akses untuk memonitoring pemakaian energi. BLoC Pattern diterapkan dalam pemisahan file BLoC yang berisi business logic, event dan state. Berdasarkan pengujian yang telah dilakukan antara aplikasi dengan BLoC Pattern dan tanpa BLoC Pattern diketahui bahwa aplikasi yang menerapkan BLoC Pattern memiliki keunggulan dalam kecepatan berjalannya suatu program. Dalam uji performa yang dilakukan BLoC unggul tiga kali lebih cepat dari 4 kali percobaan [19].

Penelitian kedua yaitu oleh Agung Wibowo dengan judul “Pemanfaatan Flutter pada Fitur Kenaikan Gaji Berkala dalam Aplikasi mobile ASN Memayu (Studi Kasus CV. Atsoft Teknologi)”, dalam penelitiannya membahas tentang pengembangan aplikasi ASN Memayu. Aplikasi ini berfungsi untuk memudahkan para pegawai di BKD DIY dalam mengakses sistem kepegawaiannya serta memberi fleksibilitas waktu dalam pengecekan data secara berulang. Fitur kenaikan gaji berkala tersebut berfungsi untuk melihat data gaji sesuai dengan golongannya. Dalam pengembangannya, BLoC Pattern digunakan untuk memisahkan antara file service, UI, dan BLoC. BLoC berfungsi sebagai wadah untuk mendeklarasikan data yang ada pada server. BLoC digunakan untuk mencari dan menampilkan data-data kenaikan gaji berkala yang akan dipanggil oleh widget BlocBuilder(). Data output BLoC yang dipanggil oleh widget tersebut akan dimasukkan kedalam widget list Kenaikan Gaji. Berdasarkan pengujian yang dilakukan oleh peneliti, diketahui bahwa penggunaan BLoC Pattern mempermudah developer untuk memperbarui aplikasi ASN Memayu selanjutnya [20].

Business Logic Component Pattern adalah pola desain yang membantu memisahkan user interface dengan business logic dan memungkinkan developer untuk membagi komponen pada proyek menjadi komponen presentational, BLoC, dan backend [12]. Gambar alur proses dengan BLoC Pattern dapat dilihat pada Gambar 1. Pengembang dapat fokus untuk mengkonversi sebuah event menjadi state saat menggunakan BLoC Pattern. Hal ini memudahkan pengembang untuk memahami kode yang dikembangkan sebelumnya serta membuat aplikasi yang lebih kompleks menjadi komponen yang lebih kecil agar lebih mudah untuk diuji. Ada beberapa komponen dalam menerapkan BLoC Pattern yaitu:

- 1) View berfungsi sebagai tempat memasukkan input dan menampilkan output.
- 2) Bloc
 - a) Bloc berfungsi sebagai penghubung antara view dengan service.
 - b) State berfungsi untuk memproses response dari service (yang ditampung oleh model) dan



Gambar 2. Example of a figure caption. (figure caption)

menyatakan apakah response failed, error atau success.

- c) Event berfungsi sebagai kondisi ketika proses meng-hit atau mengirim ke service.
- 3) Model berfungsi menerima response dari Restful API berupa JSON yang ditampung dalam sebuah class.
- 4) Service
 - a) Service berfungsi untuk meng-hit Restful API atau sebagai servernya.
 - b) Params request berfungsi sebagai data berupa class yang dibutuhkan ketika meng-hit ke Restful API.
- 5) Route berfungsi sebagai penghubung antara view satu dengan yang lainnya.

F. Restful API

RESTful API adalah API (Application Programming Interface) yang mengimplementasikan konsep REST (Representational State Transfer). API ini bertugas sebagai penghubung yang memastikan sebuah aplikasi untuk dapat berkomunikasi atau berbagi data satu sama lain tanpa harus menggunakan bahasa pemrograman yang sama. API terdiri dari beberapa elemen seperti protocols, function, tools dan lainnya. Komunikasi yang digunakan oleh REST adalah arsitektur berbasis web dan umumnya menggunakan HTTP (Hypertext Transfer Protocol) sebagai komunikasi datanya. Gambar alur proses dengan RESTful API dapat dilihat pada Gambar 2. Ada beberapa komponen utama yaitu:

1) Desain URL

Penamaan dan struktur URL yang konsisten akan menghasilkan API yang baik dan mudah dimengerti developer. URL API biasa disebut dengan endpoint. URL dalam RESTful API sangatlah penting. Hal ini karena diakses melalui teknologi WWW (World Wide Web). Contoh penulisan URL yang baik adalah sebagai berikut:

- a) `http://example.com/payments`
- b) `http://example.com/payments/2`

2) HTTP Verbs

HTTP Verbs merupakan method request ketika client melakukan request. Method yang sering digunakan adalah GET (membaca data atau mendapatkan data), POST (membuat data baru dengan menyisipkan data dalam body saat request dilakukan), PUT (mengupdate data), dan DELETE (menghapus data). Selain itu terdapat juga PATCH (memodifikasi data tertentu), HEAD (mengambil informasi URL dari server), dan OPTIONS (mencari tahu method HTTP mana yang dapat diakses client) [21].

3) HTTP Status Code

HTTP Status Code merupakan kode standarisasi yang memberitahu hasil request kepada client. Terbagi menjadi 5 kelompok yaitu:

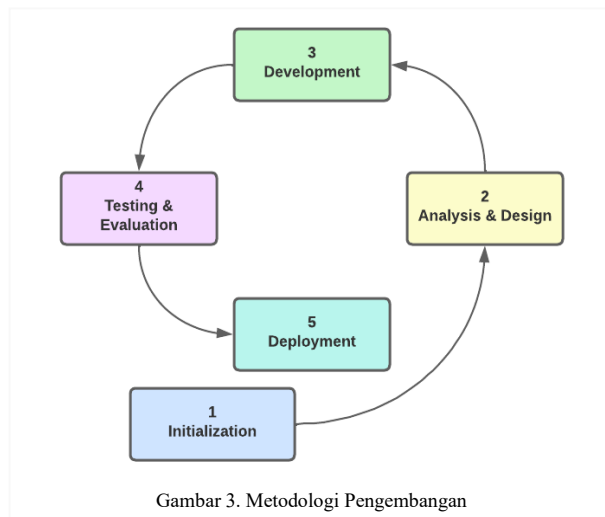
- a) 1XX: response code yang menampilkan respons informasi bahwa permintaan diterima atau melanjutkan proses.
 - b) 2XX: response code yang menampilkan bahwa request berhasil.
 - c) 3XX: response code yang menampilkan pesan pengalihan atau perlu tindakan lebih lanjut yang harus diambil untuk menyelesaikan permintaan.
 - d) 4XX: response code yang menampilkan bahwa request terjadi kesalahan pada sisi client.
 - e) 5XX: response code yang menampilkan bahwa request terjadi kesalahan pada sisi server.
- 4) Format *Response*

Format yang digunakan pada RESTful API dapat berupa XML (Extensible Markup Language) atau JSON (Javascript Object Notation). Setelah mendapatkan data response, client baru dapat menggunakan dengan cara mem-parsing data tersebut dan diolah berdasarkan kebutuhan [22].

III. METODOLOGI

Pengembangan fitur aplikasi mobile banking Agen46 dikerjakan bersama dalam sebuah tim yang terdiri dari product owner, team member (frontend mobile engineer), dan scrum master. Pengembangan dilakukan dalam beberapa tahapan, yaitu:

- 1) Initialization, yaitu tahapan yang dilakukan dengan tujuan melakukan perencanaan awal untuk mengumpulkan segala sumber daya yang dibutuhkan dalam menyelesaikan proyek. Pada tahap ini persiapan teknologi sudah ditentukan dari awal baik instalasi software maupun persiapan hardware/device yang digunakan untuk membantu proses pengembangan aplikasi. Selain itu terdapat pembuatan git repository yang digunakan untuk mendokumentasikan fitur yang telah dikembangkan. Setiap satu fitur disimpan dalam satu branch git.
- 2) Analysis & Design, yaitu tahapan yang dilakukan untuk menguraikan segala kebutuhan dan bagaimana proyek dikerjakan. Dalam hal ini ada beberapa dokumentasi yang mendukung pelaksanaan pengembangan seperti dokumentasi spesifikasi API dan dokumentasi desain UI/UX. Dokumentasi spesifikasi API sudah dibuat oleh tim lain dalam bentuk file pdf yang terdiri dari path, request definition, url, format request, response definition, success response code, response success, error response code, dan response (error). Desain tampilan aplikasi sudah dibuat oleh tim UI/UX designer. Tampilan fitur ini memiliki perbedaan yang cukup besar dari aplikasi sebelumnya yang sudah ada. Selama proses pengembangan tim frontend engineer juga diberi akses kedalam Figma untuk melihat desain aplikasinya.
- 3) Development, yaitu tahapan pengerjaan pengembangan fitur aplikasi dengan cara mengimplementasikan desain dan kebutuhan yang diperlukan ke dalam code pemrograman. Pada tahap ini frontend engineer



Gambar 3. Metodologi Pengembangan

menerapkan BLoC Pattern untuk membuat fitur dan membuat integrasi API. Selama proses pengembangan engineer harus melihat panduan dokumentasi spesifikasi API dan desain UI/UX yang sudah diberikan. Apabila terdapat kesalahan ataupun perbedaan data antara bagian backend dengan frontend maka frontend mobile engineer harus berkoordinasi dahulu dengan product owner. Product owner dalam hal ini adalah senior engineer. Device yang digunakan untuk melihat hasil pengembangan adalah simulator device. Setiap engineer diberikan satu hak akses username dan password yang unik untuk menjalankan simulasi transaksi yang dilakukan.

- 4) Testing & Evaluation, yaitu tahapan untuk melakukan pengujian terhadap fitur yang telah dibuat serta melakukan review hasil kepada product owner. Tahap ini juga digunakan untuk mendapatkan saran serta improvement yang diperlukan terhadap fitur yang dibuat agar dapat berjalan dengan baik. Pengujian yang dilakukan adalah dengan metode blackbox testing. Pengujian dilakukan untuk mengetahui apakah fungsionalitas dalam fitur tersebut sudah berjalan dengan baik. Hal ini dapat dilihat ketika transaksi top up berhasil maka data saldo user akan berkurang. Pengujian top up dilakukan menggunakan nomor tujuan transaksi tersendiri yang telah disetting oleh bagian backend engineer. Evaluasi yang dilakukan terdiri dari weekly meeting dan retrospective meeting. Weekly meeting dilaksanakan oleh team engineer dan product owner terkait progress pengembangan fitur sedangkan retrospective meeting dilakukan oleh team engineer dan scrum master untuk membahas kendala/hambatan pelaksanaan teknis pengembangannya.
- 5) Deployment, yaitu tahapan yang dilakukan untuk mendokumentasikan hasil dari fitur yang telah selesai dikembangkan. Deployment yang dilakukan tidak sampai pada stage production namun masih pada stage pengembangan saja. Hasil pengembangan fitur yang telah dilakukan nantinya akan dideploy kedalam platform DigiLabs dengan bantuan Jenkins sebagai tools automation deploy nya.

Fitur top up e-money LinkAja pada aplikasi m-banking Agen46 dikembangkan menggunakan metode agile dengan sistem manajemen kerja scrum dan weekly sprint planning sebagai media pelaporan hasil progress pengembangan yang telah dilakukan dan konsultasi terkait kendala yang dihadapi selama proses pengembangan berlangsung. Gambar metodologi pengembangan fitur dapat dilihat pada Gambar 3.

IV. HASIL DAN PEMBAHASAN

A. Initialization

Pada tahap analisis kebutuhan, spesifikasi proyek Agen46 ditentukan agar pengembangan yang dilakukan dapat sesuai dengan harapan dan tujuan. Beberapa teknologi yang digunakan dalam pengembangan fitur dapat dilihat pada Tabel 1. Setelah spesifikasi teknologi ditentukan, tahap selanjutnya adalah membuat git repository baru dan menerima git repository yang telah disiapkan agar dapat membuat fitur dengan framework yang telah ditentukan.

TABEL 1. SPESIFIKASI TEKNOLOGI M-BANKING AGEN46

Aspek	Teknologi
Bahasa pemrograman	Dart 2.14.4
Framework	Flutter 2.5.3
Source Code Version	Gitlab
Code Editor	Android Studio
Deployment Platform	Jenkins

Pengembangan proyek aplikasi ini terdiri juga terbagi dalam beberapa peran penting, di antaranya adalah:

- Product Owner yang bertanggung jawab serta memastikan bahwa aplikasi yang dibuat oleh scrum team memenuhi nilai bisnis dan kebutuhan pengguna. Selain itu product owner juga bertugas membagi backlog untuk setiap fitur yang dikembangkan oleh scrum team.
- Team Member/Frontend Mobile Engineer yang bertugas melakukan eksekusi dan pengembangan aplikasi hingga selesai. Pengembangan yang dilakukan berdasarkan kebutuhan sistem yang telah ditentukan sebelumnya. Team member terdiri dari frontend, backend, ui/ux design, quality assurance dan lainnya.
- Scrum Master yang bertugas untuk memastikan bahwa proses scrum dalam pengerjaan proyek berjalan dengan lancar. Selain itu scrum master juga bertugas memastikan praktik agile dijalankan oleh semua anggota tim serta menjadi penghubung antara product owner dengan tim pengembang.

B. Maintaining the Integrity of the Specifications

Agen46 merupakan mitra BNI (perorangan atau badan hukum) yang telah bekerjasama dengan BNI untuk menyediakan layanan perbankan kepada masyarakat seperti Layanan Laku Pandai, Layanan LKD dan Layanan e-Payment. Proyek Agen46 merupakan aplikasi berbasis mobile yang akan digunakan oleh para nasabah BNI untuk melakukan kegiatan transaksi. Pada pengembangan proyek ini lebih fokus kepada pengembangan frontend salah satu fitur yang ada pada aplikasi m-banking Agen46 yaitu fitur

top up uang elektronik (e-money) LinkAja. Ada dua cara melakukan top up yaitu sebagai user baru dan sebagai user favorit. Dalam fitur tersebut ada beberapa aktivitas yang dapat dilakukan dalam melakukan top up e-money yaitu:

- Melakukan nominal top up.
- Membagikan bukti transaksi.
- Meng-set user favorit.
- Mencetak bukti transaksi.

Analisis merupakan tahapan yang dilakukan oleh tim developer internal BNI yang berposisi sebagai System Analyst bersama Project Manager (PM). Pada tahap ini, penulis sebagai Software Engineer (Frontend) tidak melakukan analisis karena semua kebutuhan telah dianalisis sebelumnya. Setelah dianalisis, dilanjutkan dengan pengembangan tahap desain. Pembuatan tampilan baru bertujuan untuk menciptakan sebuah tampilan aplikasi yang lebih menarik. Tahapan ini dilakukan oleh UI/UX Designer.

C. Maintaining the Integrity of the Specifications

1) Membuat Integrasi API

Pembuatan Integrasi API terdiri dari beberapa bagian yaitu bloc dan service. Adapun beberapa rincian terkait struktur file pembuatan fitur top up uang digital LinkAja dapat dilihat pada Tabel 2.

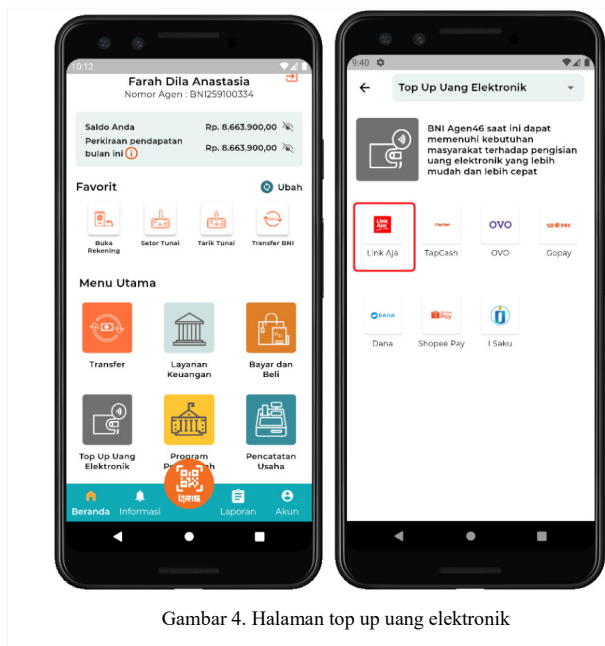
TABEL 2. STRUKTUR FILE FITUR TOP UP E-MONEY LINKAJA

No	Bagian	Keterangan
1	Bloc	LinkAja bloc
2		LinkAja event
3		LinkAja state
4	Model	LinkAja payment model
5		LinkAja inquiry model
6		LinkAja set favorite billing model
7	Service	LinkAja service
8	Routes	LinkAja url ist
9		LinkAja routes
10	View	LinkAja favorite page
11		LinkAja input page
12		LinkAja top up page
13		LinkAja inquiry page
14		LinkAja bukti transaksi page

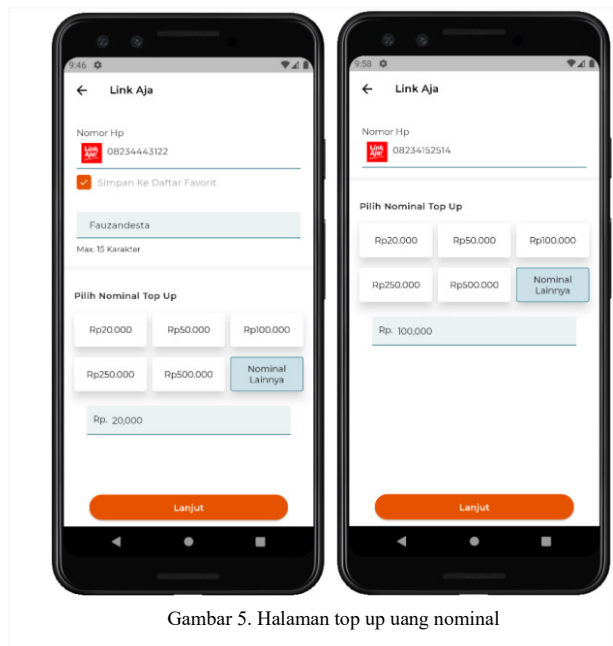
Alur proses semua berjalan dari view. View akan meng-hit ke bloc dengan event inquiry. Bloc akan diteruskan ke service inquiry dengan params request. Kemudian dari service mendapat response berupa model yang kemudian diteruskan ke bloc. Bloc akan mengecek model apakah yang masukan sesuai dengan state. Dari view akan mengecek state, memastikan apakah proses akan meneruskan ke view inquiry atau tidak. Dan hal tersebut berulang sama pada model payment.

2) Membuat route pada menu top up LinkAja

Salah satu fitur top up e-money yang berada pada aplikasi ini adalah top up e-money dengan merchant LinkAja. Pengguna harus login terlebih dahulu dengan username dan password kemudian setelah login, nasabah akan diarahkan ke halaman utama aplikasi. Di halaman utama terdapat menu pilihan layanan salah satunya adalah menu “Top Up Uang Elektronik.”



Gambar 4. Halaman top up uang elektronik



Gambar 5. Halaman top up uang nominal

Penambahan route dari fitur top up LinkAja dilakukan untuk mengarahkan event ke view yang dituju yaitu dengan cara membuat list-route dan route sesuai dengan dokumentasi spesifikasi API yang telah dibuat. Route berfungsi untuk menghubungkan antara view satu dengan yang lainnya. Route dibuat dengan menggunakan percabangan case berdasarkan menu yang akan dibuat. Menu “Top Up Uang Elektronik” dapat dilihat pada Gambar 4.

Di dalam route diimplementasikan penerapan widget seperti PageRouteBuilder() dan SlideTransation(). PageRoute Builder() bertugas untuk mengarahkan view menuju service dan bloc nya dengan bantuan BlocProvider(). SlideTransation() berfungsi untuk memberikan efek animation ketika di klik. Selanjutnya membuat list-route untuk mengarahkan ke dalam dua model yaitu inquiry dan payment sesuai dengan penamaan path yang telah ditentukan. Di dalam list-route juga membuat path untuk mendapatkan last transaction yang dilakukan, meng-set user favorit, dan meng-get list favorit.

Ketika nasabah mengklik menu “Top Up Uang Elektronik” maka akan diarahkan ke dalam halaman utama uang elektronik yang berisi pilihan menu top up seperti DANA, LinkAja, Tapcash, dan lain-lain.

3) Membuat Halaman Top Up Nominal

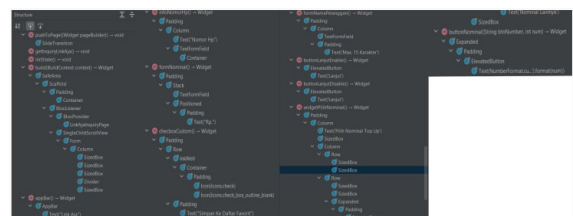
Pada halaman top up nominal nasabah bisa menampilkan dua kemungkinan perbedaan tampilan. Pertama ketika nasabah menggunakan data user favorit untuk melakukan transaksi yang terdapat pada gambar bagian sebelah kanan dan kedua adalah ketika nasabah menginputkan data baru sebagai user favorit yang terdapat pada gambar bagian sebelah kiri. Perbedaan keduanya terletak pada bagian checkbox untuk meng-set sebagai user favorit. Apabila melakukan transaksi sebagai user baru maka akan muncul checkbox untuk meng-set ke dalam daftar user favorit dengan mengisi nama pengguna. Nama pengguna dapat diisi dengan batas maksimal 15 karakter. Sedangkan apabila

melakukan transaksi dengan daftar user favorit tidak akan muncul checkbox tersebut.

Kedua bagian ini memiliki kesamaan pada pilihan nominal top up. Terdapat dua cara pilihan nominal yaitu dengan jumlah nominal button seperti Rp 20.000, Rp 50.000, Rp 100.000, Rp 250.000 dan Rp 500.000. Selain itu jika nominal yang diinginkan nasabah tidak sesuai dengan yang ada pada button maka bisa dengan button “Nominal Lainnya” dan otomatis akan menampilkan field untuk mengisi nominal yang diinginkan.

Button pada halaman ini terdapat dua kondisi yaitu saat disable dan enable. Button akan aktif (enable) dan berwarna jingga jika nasabah telah berhasil memilih atau mengisi nominal top up nya yang nantinya akan mengarahkan ke halaman konfirmasi transaksi. Namun jika nasabah belum mengisi nominal top up nya maka button “Lanjut” tidak dapat diklik untuk dapat lanjut ke halaman konfirmasi transaksi atau berstatus disable. Halaman top up nominal dapat dilihat pada Gambar 5. Struktur frontend halaman ini terbagi menjadi beberapa widget yaitu BuildContext(), appBar(), infoNomorHp(), formNominal(), checkboxCustom(), formNamaPelanggan(), buttonLanjutEnable(), buttonLanjutDisable(), widgetPilihanNominal(), dan buttonNominal(). Struktur widget halaman top up nominal dapat dilihat pada Gambar 6.

Widget BuildContext terdapat class BlocListener() untuk menampilkan state dan BlocProvider() untuk



Gambar 6. Struktur widget halaman top up nominal

mengarahkan bloc ke halaman konfirmasi yaitu LinkAjaInquiryPage(). Di dalam class SingleChildScrollView terdapat class Form() yang tersusun atas beberapa class Column(). Class Column() ini berfungsi membungkus beberapa widget di dalamnya yaitu widget infoNomorHp(), checkboxCustom(), formNamaPelanggan(), dan widgetPilihanNominal(). Setiap widget ini dibungkus dalam sebuah class bernama SizedBox() yang bertugas memberi jarak baik secara vertikal maupun horizontal. Antara class formNamaPelanggan() dengan class widgetPilihanNominal() terpisahkan oleh sebuah class Divider().

Widget appBar() tersusun atas class AppBar() dan Text(). Di dalam class AppBar() terdapat properti seperti elevation, backgroundColor, iconTheme, dan title. Sedangkan di dalam class Text() ditambahkan properti style untuk mengatur teks bertuliskan "Link Aja".

4) Membuat halaman konfirmasi top up

Halaman selanjutnya setelah mengisi nominal top up adalah halaman konfirmasi top up. Pada halaman ini nasabah dapat melihat informasi konfirmasi top up yang dilakukan yaitu data payment dan data transaksi. Detail data transaksi yang dilakukan yaitu berupa jenis transaksi yang dilakukan beserta dengan informasi nomor handphone beserta nama penggunanya. Data payment pada halaman ini berupa nominal biaya yang harus dibayar yaitu total penjumlahan antara biaya administrasi dan nominal asli top up. Pada halaman ini juga terdapat bagian untuk mengisi password transaksi dan apabila telah selesai mengisi password nasabah bisa mengklik button "Lanjut". Apabila proses transaksi top up yang dilakukan berhasil dan password transaksi yang diinputkan benar maka akan mengarahkan ke halaman akhir dari proses yaitu bukti transaksi top up. Dan apabila nasabah ingin membatalkan top up yang dilakukan bisa mengklik button "Batal" untuk kembali ke halaman nominal top up.

Halaman ini tersusun atas beberapa widget yaitu BuildContext(), containerDataTransaksi(), containerDataPayment(), buttonNavigasi(), sessionExpiredWidget(), dan pushReplacement(). Halaman ini menggunakan class SafeArea() dan

Scaffold() untuk bagian context nya. Struktur widget halaman konfirmasi transaksi dapat dilihat pada Gambar 7.

Pada widget BuildContext() terdapat class BlocListener() dan Bloc Provider() untuk menampilkan state context nya serta mengarahkan pengiriman data yang dibuat pada inquiry ke halaman bukti transaksi yaitu LinkAjaStatusPage(). Class SingleChildScrollView() digunakan untuk membungkus widget containerDataTransaksi() dan containerDataPayment() yang tersusun dalam class Column(). Pada bagian atas class Container() terdapat class Text() bertuliskan "Konfirmasi Transaksi" yang diberi jarak oleh class Padding().

Widget containerDataTransaksi() berisi data dari transaksi yang dilakukan yang tersusun dalam class Row() serta diberi class Expanded() untuk memperluas row yang dibuat. Setiap data memiliki ukuran perbandingan yang sama dengan menambahkan properti flex beserta value di dalam class Expanded().

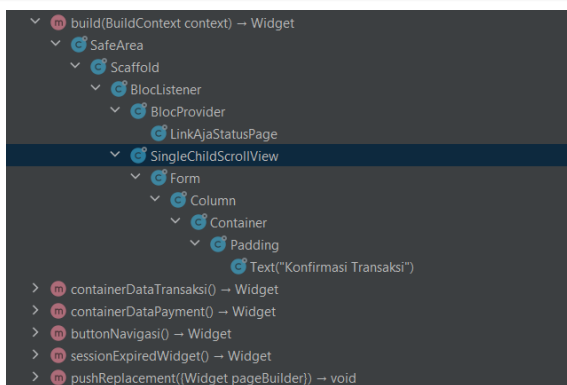
Widget containerDataPayment() terdiri dari data nominal yang harus dibayarkan dan tersusun juga dalam class Row() dan Expanded(). Pada bagian ini terdapat class Stack() untuk menumpuk class TextFormField() sebagai tempat mengisi password transaksi. Terdapat juga penambahan properti validator untuk memastikan bahwa password transaksi tidak null.

Widget buttonNavigasi() digunakan untuk menaruh button "Lanjut" dan "Batal". Kedua button ini menggunakan class InkWell() untuk memberikan properti onTap() yang bertugas mengarahkan halaman ketika salah satu button diklik.

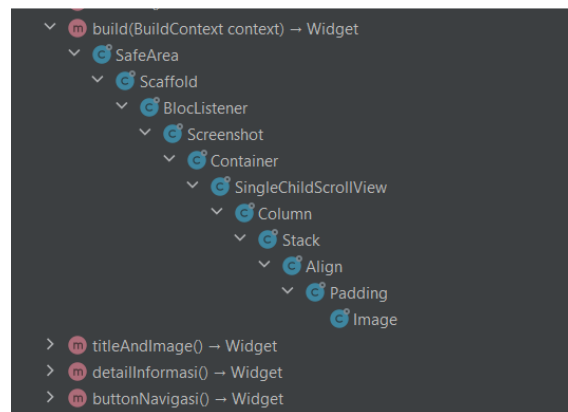
Widget sessionExpiredWidget() berguna untuk memastikan nasabah bahwa session berada dalam state yang benar. Apabila session tidak ditemukan maka nasabah akan diberi informasi log untuk melakukan login ulang.

Widget pushReplacement() digunakan sebagai navigator widget ketika proses konfirmasi yang dilakukan berhasil maka akan mengarahkan ke halaman terakhir yaitu LinkAjaStatusPage(). Class SlideTransition() digunakan untuk memberikan efek animation ketika berpindah ke halaman selanjutnya.

5) Membuat halaman bukti transaksi top up



Gambar 7. Struktur widget halaman konfirmasi transaksi



Gambar 8. Struktur widget halaman bukti transaksi

Halaman terakhir dari proses top up adalah bukti transaksi top up. Halaman bukti transaksi menampilkan beberapa informasi detail terkait top up yang dilakukan dari data transaksi, data payment, hingga detail kode bank nya. Pada halaman ini juga menampilkan status log aktivitas jika nasabah menyimpan data baru sebagai user favorit. Ada tiga aktivitas yang dapat diarahkan oleh button yaitu membagi bukti transaksi, mencetak bukti transaksi dengan device lain, dan kembali ke halaman utama aplikasi.

Halaman ini terdiri dari widget BuildContext(), titleAndImage(), detailInformasi(), dan buttonNavigasi(). Halaman ini menggunakan class SafeArea() dan Scaffold() untuk menampilkan context didalamnya. Struktur widget halaman bukti transaksi dapat dilihat pada Gambar 8.

Widget BuildContext() tersusun dari class BlocListener() yang berfungsi menampilkan status penyimpanan user favorit. Selain itu di sini juga membungkus widget lainnya seperti buttonNavigasi(), titleAndImage(), dan detailInformasi(). Tampilan background di widget detailInformasi() menggunakan class Stack() untuk meletakkan class Imageassets() sebagai watermark dari logo bank BNI.

Widget titleAndImage() berfungsi untuk meletakkan penulisan judul dari status dengan style penulisan yang dimiliki BNI sendiri.

Widget detailInformasi() tersusun dari properti Imageasset() untuk logo BNI Agen46, class Text() untuk judul status transaksi, dan class Column() untuk informasi detail transaksi yang dilakukan. Informasi detail berada dalam sebuah class Row() yang diberi jarak antar data dengan class Expanded().

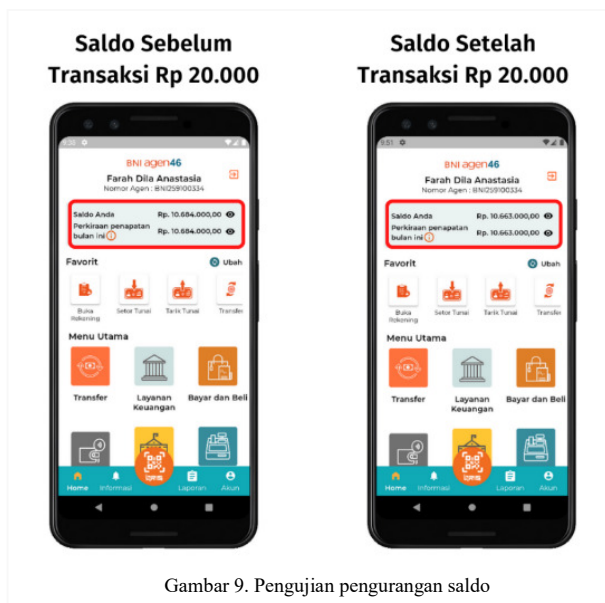
Widget buttonNavigasi() tersusun dari class Text() sebagai judul perintah penyimpanan resi transaksi, class ButtonNavigation(). Class ButtonNavigation() terdiri dari tiga bagian yaitu membagi bukti transaksi, mencetak bukti transaksi dengan device lain, dan kembali kehalaman utama aplikasi dimana setiap bagian menggunakan class InkWell() untuk

mengarahkan setiap bagian ke halaman atau event yang dituju.

D. Testing dan Evaluasi

Pada tahap ini dilakukan pengujian dengan blackbox testing manual dan evaluasi fitur berupa review yang disampaikan kepada product owner sebagai ketua tim proyek. Blackbox testing dilakukan untuk menguji fungsionalitas setiap bagian dari fitur yang dibuat dan dapat diimplementasikan dengan baik [23]. Pengujian dilakukan dengan menggunakan simulator device. Simulator device ini hanya bisa dijalankan pada satu device saja baik melalui handphone maupun dengan emulator di Android Studio. Apabila ingin pindah device maka harus minta izin request kepada tim simulator. Selama pengujian berlangsung engineer diberikan simulasi akun berupa username, password, dan data saldo user dari tim backend engineer untuk mencoba apakah integrasi API yang dibuat sudah berjalan. Pada pengujian ini frontend engineer harus memastikan bahwa setiap widget yang dibuat harus sesuai dengan desain UI/UX dari tim designer. Selain itu pengiriman data dari backend menuju frontend juga harus dapat tampil pada tampilan user interface dari fitur yang dibuat.

Hal ini dapat terbukti berhasil jika terdapat pengurangan saldo user setelah dan sebelum bertransaksi. Pengujian pengurangan saldo dapat dilihat pada Gambar 9. Pada tahap ini frontend engineer hanya melakukan pengujian secara fungsionalitas dari data input hingga output halaman bukti transaksi. Pengujian performa aplikasi dengan Flutter akan diuji lebih lanjut oleh tim QA. Hasil pengujian fungsionalitas fitur dapat dilihat pada Tabel 3.



Gambar 9. Pengujian pengurangan saldo

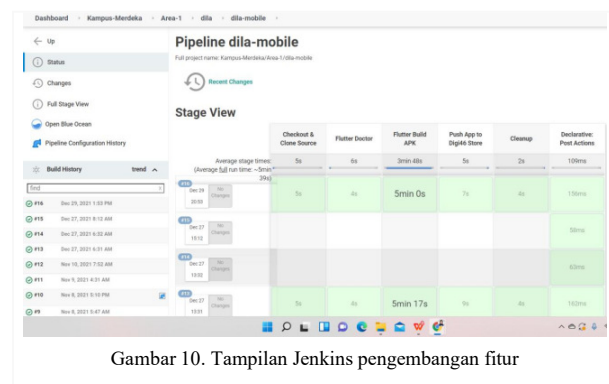
TABEL III. PENGUJIAN FUNGSIONALITAS

Fungsionalitas	Hasil diharapkan	Hasil pengujian	Status
Halaman top up nominal	Menampilkan halaman top up nominal	Menampilkan halaman top up nominal	Berhasil
	Menampilkan nomor handphone yang dituju	Menampilkan nomor handphone yang dituju	Berhasil
	Memilih nominal top up dengan pilihan button	Memilih nominal top up dengan pilihan button	Berhasil
	Memilih nominal top up dengan mengisi textfield	Memilih nominal top up dengan mengisi textfield	Berhasil
	Mengklik button "Lanjut" untuk menuju halaman konfirmasi	Mengklik button "Lanjut" untuk menuju halaman konfirmasi	Berhasil
Halaman konfirmasi transaksi	Menampilkan halaman konfirmasi transaksi	Menampilkan halaman konfirmasi transaksi	Berhasil
	Menampilkan data payment	Menampilkan data payment	Berhasil
	Mengisi password	Mengisi password	Berhasil

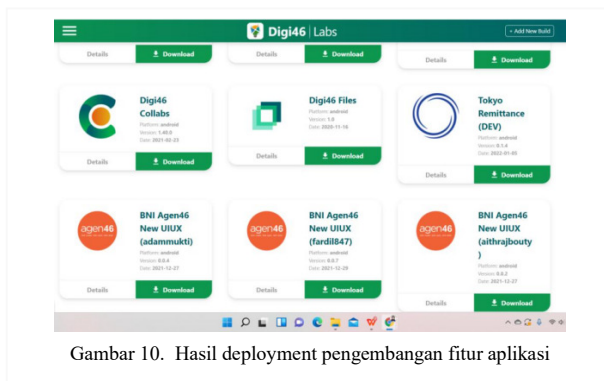
	transaksi	transaksi	
	Mengklik <i>button</i> “Lanjut” untuk menuju halaman bukti transaksi	Mengklik <i>button</i> “Lanjut” untuk menuju halaman bukti transaksi	Berhasil
	Mengklik <i>button</i> “Batal” untuk membatalkan konfirmasi transaksi dan kembali menuju halaman <i>top up</i> nominal	Mengklik <i>button</i> “Batal” untuk membatalkan konfirmasi transaksi dan kembali menuju halaman <i>top up</i> nominal	Berhasil
Halaman bukti transaksi	Menampilkan halaman bukti transaksi	Menampilkan halaman bukti transaksi	Berhasil
	Menampilkan keseluruhan detail data transaksi	Menampilkan keseluruhan detail data transaksi	Berhasil
	Menampilkan status transaksi	Menampilkan status transaksi	Berhasil
	Mengklik <i>button</i> “Cetak” untuk mencetak bukti transaksi dengan <i>device</i> lain	Mengklik <i>button</i> “Cetak” untuk mencetak bukti transaksi dengan <i>device</i> lain	Berhasil
	Mengklik <i>button</i> “Bagikan” untuk membagikan bukti transaksi dalam berbagai format file	Mengklik <i>button</i> “Bagikan” untuk membagikan bukti transaksi dalam berbagai format file	Berhasil
	Mengklik <i>button</i> “Home” untuk kembali ke halaman utama aplikasi	Mengklik <i>button</i> “Home” untuk kembali ke halaman utama aplikasi	Berhasil
	Mengecek terdapat pengurangan saldo pada halaman “Home” setelah melakukan top up	Mengecek terdapat pengurangan saldo pada halaman “Home” setelah melakukan top up	Berhasil

E. Deployment

Deployment dilakukan dengan menggunakan tools automation deployment yaitu Jenkins. Jenkins menyediakan layanan integrasi custom bagi pengembangan perangkat lunak. Penggunaan Jenkins dapat mempercepat Software



Gambar 10. Tampilan Jenkins pengembangan fitur



Gambar 10. Hasil deployment pengembangan fitur aplikasi

Development Lifecycle (SDLC) bagi perusahaan. Pengembang diberi akses untuk melihat tracking progress deployment yang telah dilakukan sehingga dapat mengecek langsung apabila terjadi terjadi kegagalan build di stage tertentu [24]. Tampilan Jenkins pengembangan fitur dapat dilihat pada Gambar 10.

Jenkins pipeline sudah dibuat sebelumnya sehingga pengembang cukup menambahkan tag pada branch git yang dikembangkan kemudian akan otomatis ter deploy pada Jenkins. Hasil deployment dari aplikasi yang telah dibuild akan langsung terbentuk ke dalam tools bernama DigiLabs. Tools ini merupakan buatan milik internal BNI yang berisi kumpulan aplikasi-aplikasi yang telah dibuat sehingga memudahkan product owner atau scrum master jika ingin menginstall dan mencoba langsung fitur aplikasi yang telah dikembangkan. Tampilan hasil deployment pengembangan fitur aplikasi dapat dilihat pada Gambar 10.

V. KESIMPULAN

Berdasarkan pengembangan frontend fitur top up e-money LinkAja pada aplikasi m-banking Agen46 dengan framework Flutter dapat disimpulkan bahwa:

- 1) Pengembangan fitur top up e-money LinkAja pada aplikasi m-banking Agen46 dengan BLoC Pattern memiliki beberapa bagian dalam struktur file, yaitu 1) Bloc, 2) Model, 3) Service, 4) Route, 5) View.
- 2) Pengembangan frontend fitur top up e-money LinkAja pada aplikasi m-banking Agen46 memiliki beberapa bagian view, yaitu 1) Halaman list daftar favorit, 2) Halaman user input baru, 3) Halaman top up nominal 4) Halaman konfirmasi transaksi, 5) Halaman bukti transaksi.
- 3) Implementasi BLoC Pattern pada pengembangan frontend fitur top up e-money LinkAja dengan framework Flutter mempermudah developer dalam mengembangkan aplikasi m-banking berbasis multiplatform dan pengembangan dilakukan dengan menggunakan metodologi agile scrum.
- 4) Penelitian ini telah menghasilkan pengembangan frontend fitur top up e-money LinkAja pada m-banking Agen46 sesuai dengan analisis kebutuhan sistem yang diperlukan.
- 5) Berdasarkan hasil pengujian fungsionalitas, fitur top up e-money dapat berjalan dengan baik. Dengan hasil tersebut maka diharapkan dapat membantu BNI dalam mengembangkan aplikasi m-banking berbasis multiplatform.

Penggunaan Flutter dan BLoC Pattern pada pengembangan m-banking Agen46 memberikan dampak positif bagi developer maupun perusahaan. Developer dapat mengembangkan aplikasi dengan performa kinerja dan desain yang lebih baik dari aplikasi sebelumnya. Dengan penggunaan BLoC Pattern membuat aplikasi yang kompleks menjadi lebih mudah untuk diuji. Selain itu dengan kelebihan yang dimiliki oleh BLoC Pattern membuat biaya pengembangan aplikasi berbasis multiplatform menjadi lebih minim serta dapat menjangkau user yang lebih banyak bagi pengguna iOS maupun Android.

Masih terdapat berbagai kekurangan dalam pelaksanaan pengembangan fitur top up e-money LinkAja pada aplikasi m-banking Agen46 dengan metodologi agile scrum. Pada penerapannya scrum yang dilakukan selama proses pengembangan kurang berjalan secara maksimal. Kurangnya pemahaman anggota tim pengembang mengenai metodologi agile scrum dan perbedaan waktu kerja karena dilakukan secara work from home sehingga pelaksanaan daily meeting kurang berjalan. Selain itu terdapatnya kendala dalam pembuatan backlog selama proses pengembangan sehingga setiap perubahan status progress yang dilakukan masih menggunakan manual. Saran untuk pengembangan selanjutnya, yaitu mengoptimalkan pembuatan backlog dan implementasi agile scrum untuk menunjang pembuatan aplikasi yang lebih baik lagi kedepannya. Selain itu perlunya pengujian lebih lanjut terkait performa dari penggunaan BLoC Pattern untuk mengetahui seberapa banyak pengguna yang dapat menggunakan aplikasi secara bersamaan.

DAFTAR PUSTAKA

- [1] R. S. Johar and D. Suhartanto, "The Adoption of Online Internet Banking in Islamic Banking Industry," IOP Conference Series: Materials Science and Engineering, vol. 662, no. 3, p. 032032, Nov. 2019, doi: 10.1088/1757-899x/662/3/032032.
- [2] R. Ismail Abdelfattah, S. Awad, and M. E. Nasr, "Simplified Hybrid Secure Algorithm for Mobile Banking Application," Journal of Physics: Conference Series, vol. 1447, no. 1, p. 012052, Jan. 2020, doi: 10.1088/1742-6596/1447/1/012052.
- [3] A. Gumelar, M. I. Nasution, I. F. Oesman, F. Ramadini, M. Irfan, and Nurliana, "Technology mobile banking on customer Satisfaction," Journal of Physics: Conference Series, vol. 1477, no. 7, p. 072020, Mar. 2020, doi: 10.1088/1742-6596/1477/7/072020.
- [4] A. Yussaivi, D. Suhartanto, and Moch. E. Syarief, "An Analysis of the Determining Factors of Mobile Banking Adoption in Islamic Banks," IOP Conference Series: Materials Science and Engineering, vol. 879, p. 012174, Aug. 2020, doi: 10.1088/1757-899x/879/1/012174.
- [5] Surtikanti and R. H. Mustafa, "Utilization of Electronic Money," IOP Conference Series: Materials Science and Engineering, vol. 662, p. 022013, Nov. 2019, doi: 10.1088/1757-899x/662/2/022013.
- [6] W. Liu, X. Wang, and W. Peng, "State of the Art: Secure Mobile Payment," IEEE Access, vol. 8, pp. 13898–13914, 2020, doi: 10.1109/access.2019.2963480.
- [7] BNI, "Beranda | BNI," www.bni.co.id, 1992. https://www.bni.co.id/id-id/ (accessed May 30, 2022).
- [8] BNI, "BNI Agen46 | BNI," www.bni.co.id, 1992. https://www.bni.co.id/id-id/ebanking/bniagen46 (accessed Jun. 01, 2022).
- [9] Flutter, "Flutter Documentation," flutter.dev, 2018. https://flutter.dev/docs (accessed May 29, 2022).
- [10] S. Boukhary and E. Colmenares, "A Clean Approach to Flutter Development through the Flutter Clean Architecture Package," IEEE Xplore, 2019. https://ieeexplore.ieee.org/abstract/document/9071367.
- [11] G. Idan Arb and K. Al-Majdi, "A Freights Status Management System Based on Dart and Flutter Programming Language," Journal of Physics: Conference Series, vol. 1530, p. 012020, May 2020, doi: 10.1088/1742-6596/1530/1/012020.
- [12] L. H. Hoang, "State Management Analyses of the Flutter Application," Bachelor's Thesis, Metropolia University of Applied Sciences Bachelor of Engineering, 2019. Accessed: May 29, 2022. [Online]. Available: https://www.theseus.fi/bitstream/handle/10024/267674/Ly_Hoang.pdf?sequence=2&isAllowed=y
- [13] J. Mantik, R. Rama Prayoga, G. Munawar, R. Jumiyani, and A. Syalsabila, "Jurnal Mantik is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0). Performance Analysis of BLoC and Provider State Management Library on Flutter," Jurnal Mantik, vol. 5, no. 3, pp. 1591–1597, 2021.
- [14] M. N. O. Sadiku, M. Tembely, S. M. Musa, and O. D. Momoh, "Mobile Banking," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 7, no. 6, pp. 75–76, Jun. 2017, doi: 10.23956/ijarsse/v7i6/01615.
- [15] G. Zamadze and I. V. Pereira, "The impact of WOM on the acceptance of mobile banking," Proceedings of the International Conference on Business Excellence, vol. 15, no. 1, pp. 377–391, Dec. 2021, doi: 10.2478/picbe-2021-0035.
- [16] D. Octabriyantingtyas, E. Suryani, and A. R. Jatmiko, "Modeling Customer Satisfaction with the Service Quality of E-Money in Increasing Profit of PT. Telekomunikasi Indonesia," Procedia Computer Science, vol. 161, pp. 943–950, 2019, doi: 10.1016/j.procs.2019.11.203.
- [17] L. Puspitawati and P. Gurning, "Electronic payment for Micro, Small and Medium Enterprises in Developing Countries," IOP Conference Series: Materials Science and Engineering, vol. 662, p. 032060, Nov. 2019, doi: 10.1088/1757-899x/662/3/032060.
- [18] Bank Indonesia, "Apa itu Uang Elektronik," www.bi.go.id, Dec. 01, 2020. https://www.bi.go.id/id/edukasi/Pages/Apa-itu-Uang-Elektronik.aspx (accessed May 28, 2022).
- [19] M. Z. Assyfa, "Implementasi Desain Pattern BLoC dan Pattern Repository pada Aplikasi Energy Management System PT. PLN Berbasis Mobile dengan Flutter," Thesis, Program Studi Teknik Informatika Jurusan Teknik Informatika dan Komputer Politeknik Negeri Jakarta, 2020.
- [20] A. Wibowo and R. Kurniawan, "Pemanfaatan Flutter Pada Fitur Kenaikan Gaji Berkala Dalam Aplikasi mobile ASN Memayu (Studi Kasus CV. Atsoft Teknologi)," AUTOMATA, vol. 3, no. 1, p. -, 2022, Accessed: Jun. 03, 2022. [Online]. Available: https://journal.uui.ac.id/AUTOMATA/article/view/21888
- [21] I. O. Suzanti, N. Fitriani, A. Jauhari, and A. Khozaimi, "REST API Implementation on Android Based Monitoring Application," Journal of Physics: Conference Series, vol. 1569, p. 022088, Jul. 2020, doi: 10.1088/1742-6596/1569/2/022088.
- [22] B. Wijayanto, E. Maryanto, S. P. Rahayu, and D. Iskandar, "The development of REST API-based android application for Micro, Small and Medium Enterprises (MSME) in Purbalingga Regency," Journal of Physics: Conference Series, vol. 1367, p. 012005, Nov. 2019, doi: 10.1088/1742-6596/1367/1/012005.
- [23] S. W. G. AbuSalim, R. Ibrahim, and J. A. Wahab, "Comparative Analysis of Software Testing Techniques for Mobile Applications," Journal of Physics: Conference Series, vol. 1793, no. 1, p. 012036, Feb. 2021, doi: 10.1088/1742-6596/1793/1/012036.
- [24] U. Saadah, A. Yunanto, I. Desy, and Permatasari, "Automatic Testing Framework Based On Serenity and Jenkins Automated Build," Dec. 2021, doi: 10.12962/j24068535.v19i2.a1017.