

**ANALISIS DAN PERANCANGAN SISTEM INTEGRASI
MIDDLEWARE PADA INTERNET BANKING**



Disusun Oleh:

N a m a : Yuvirdha Bektie Widiyandari

NIM : 18523147

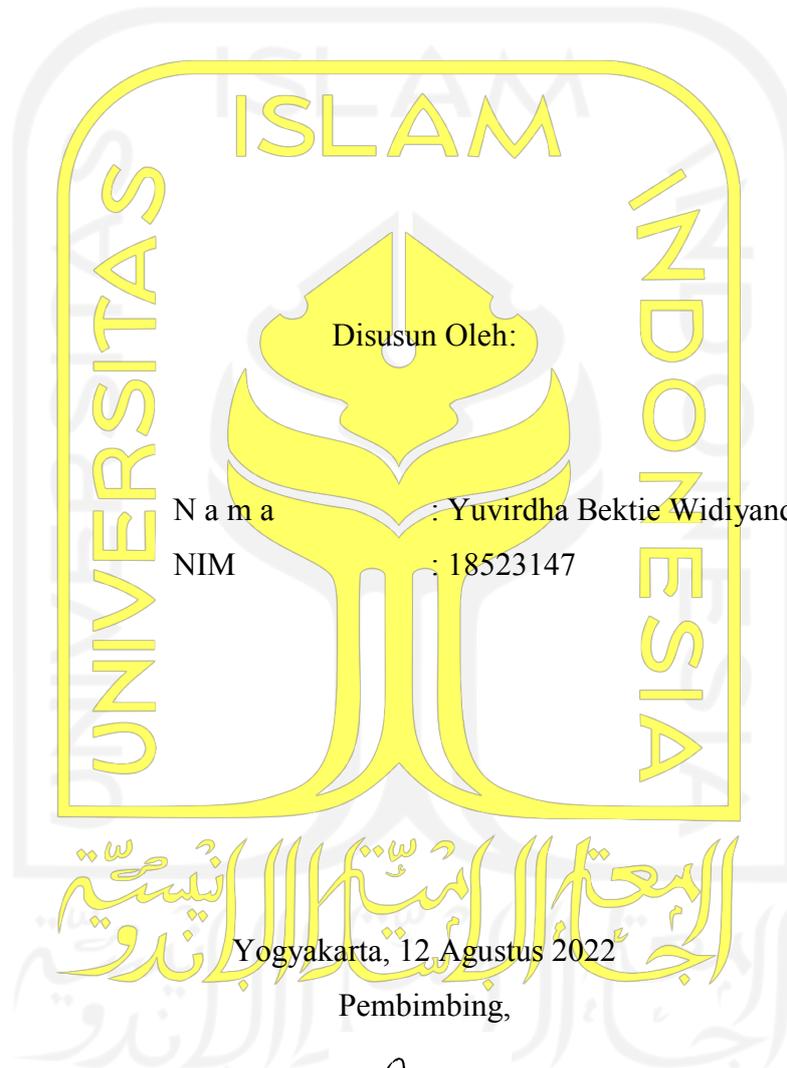
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2022

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**ANALISIS DAN PERANCANGAN SISTEM INTEGRASI
MIDDLEWARE PADA INTERNET BANKING**

TUGAS AKHIR JALUR MAGANG



Fayz

(Fayruz Rahma, S.T., M.Eng.)

HALAMAN PENGESAHAN DOSEN PENGUJI

ANALISIS PERANCANGAN SISTEM INTEGRASI MIDDLEWARE PADA INTERNET BANKING

TUGAS AKHIR JALUR MAGANG

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 12 Agustus 2022

Tim Penguji

Fayruz Rahma, S.T., M.Eng.

Anggota 1

Sheila Nurul Huda, S.Kom., M.Cs.

Anggota 2

Irving Vitra Papatungan, S.T., M.Sc., Ph.D.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Yuvirdha Bektie Widiyandari

NIM : 18523147

Tugas akhir dengan judul:

**ANALISIS DAN PERANCANGAN SISTEM INTEGRASI
MIDDLEWARE PADA INTERNET BANKING**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 12 Agustus 2022



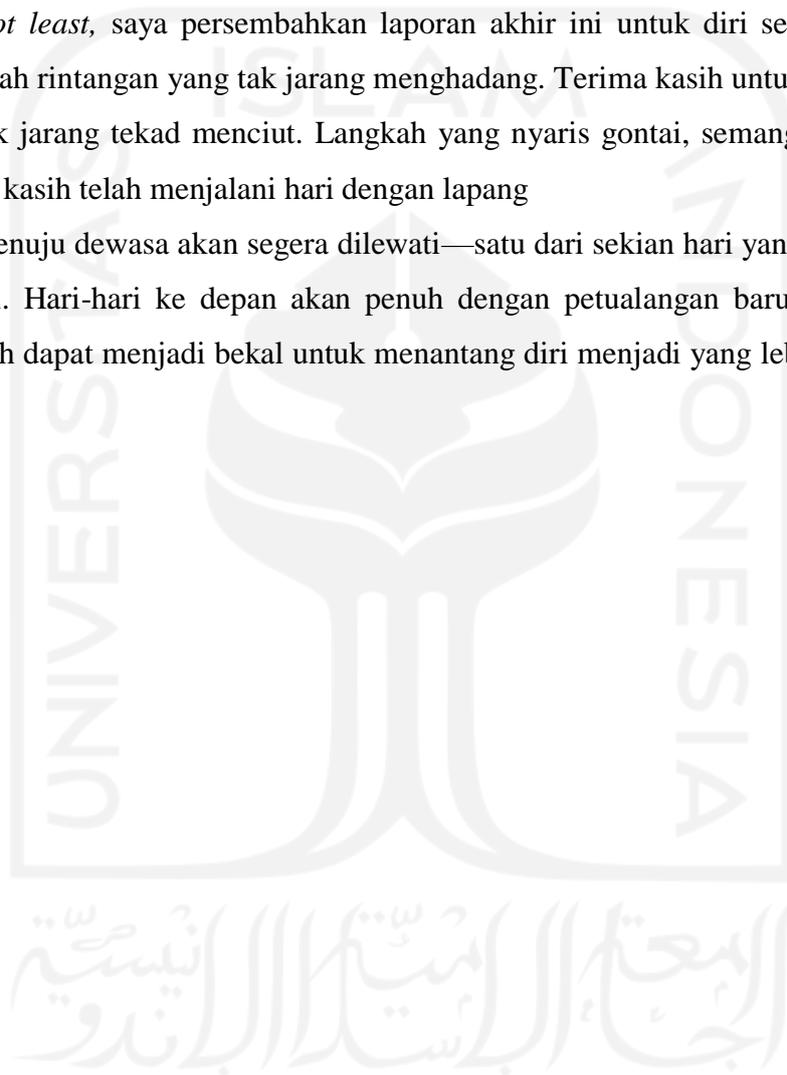
(Yuvirdha Bektie Widiyandari)

HALAMAN PERSEMBAHAN

Laporan akhir ini saya persembahkan untuk orang tua yang tidak pernah berhenti memberi dukungan dan doa, yang mendidik dengan tulusnya, serta kasih sayang yang tiada kira. Terima kasih untuk Mama dan Papa, yang telah berusaha sekuat tenaga memberi kesempatan dan pendidikan yang luar biasa. Untuk kedua adik tercinta, Arya dan Rizky, yang telah menjadi penjaga api semangat Kakak agar tidak padam.

Last, but not least, saya persembahkan laporan akhir ini untuk diri sendiri yang tetap berjuang di tengah rintangan yang tak jarang menghadang. Terima kasih untuk diri yang tetap teguh, meski tak jarang tekad menciut. Langkah yang nyaris gontai, semangat yang hampir tercerai. Terima kasih telah menjalani hari dengan lapang

Satu fase menuju dewasa akan segera dilewati—satu dari sekian hari yang dinanti selama empat tahun ini. Hari-hari ke depan akan penuh dengan petualangan baru. Semoga kelak yang telah diasah dapat menjadi bekal untuk menantang diri menjadi yang lebih baik di masa yang datang.



HALAMAN MOTO

“Dan Dia mendapatimu sebagai seorang yang bingung, lalu Dia memberikan petunjuk.”

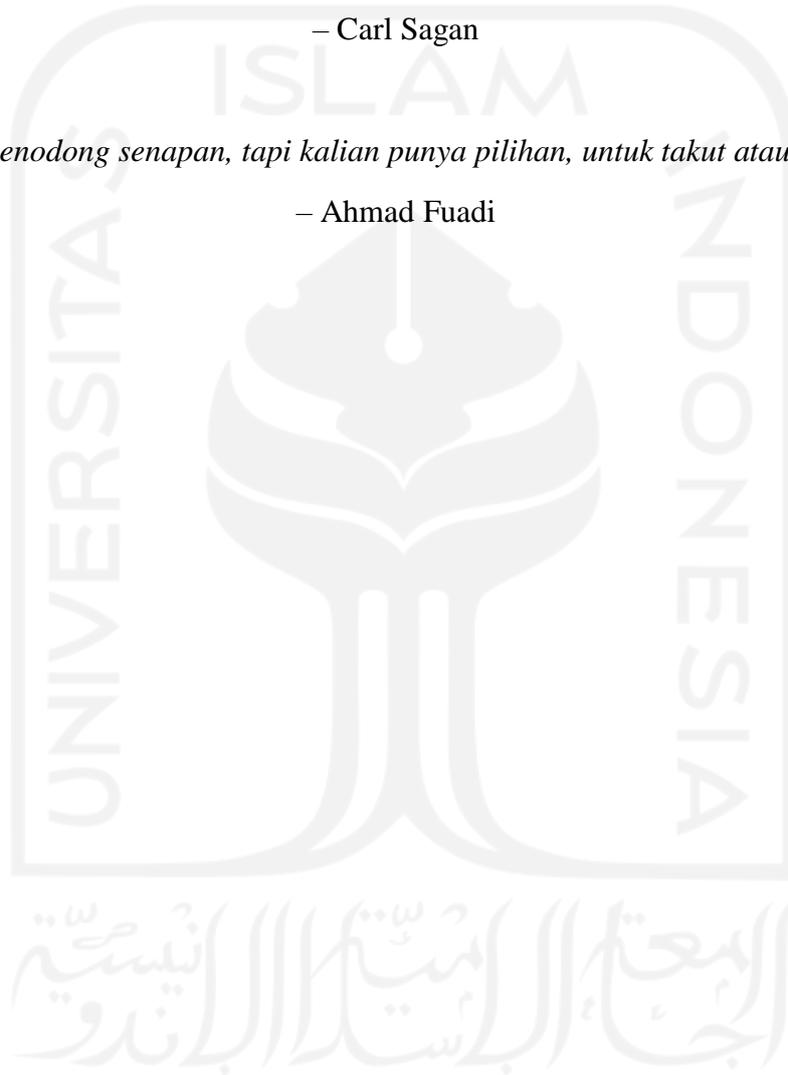
– Ad-Duha: 7

“Jauh lebih baik dalam kerentanan kita, adalah bersyukur setiap hari atas kesempatan singkat namun luar biasa yang disediakan kehidupan.”

– Carl Sagan

"Orang boleh menodong senapan, tapi kalian punya pilihan, untuk takut atau tetap tegar."

– Ahmad Fuadi



KATA PENGANTAR

Alhamdulillah, puji syukur ke hadirat Allah Swt. yang telah melimpahkan rahmat dan hidayah-Nya sehingga laporan akhir yang berjudul “Analisis dan Perancangan Sistem Integrasi *Middleware* pada Internet Banking” ini dapat selesai tepat waktu.

Laporan ini disusun untuk memenuhi persyaratan tugas akhir penjaluran Magang untuk dapat memperoleh gelar sarjana pada Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia. Pada kesempatan ini, turut diucapkan terima kasih kepada:

1. Allah Swt. yang selalu melimpahkan nikmat, rahmat, dan hidayah-Nya.
2. Bapak Poko Bektie Harijadi dan Ibu Amidah, kedua orang tua yang telah memberikan doa dan dukungannya.
3. Arya Bagus Bektie Murtadha dan Rizky Bektie Ramadhan, yang selalu menjadi penyemangat saya.
4. Ibu Fayruz Rahma, S.T., M.Eng. selaku dosen pembimbing yang telah banyak membantu dalam kegiatan magang dan penyusunan laporan akhir ini.
5. Seluruh pimpinan, dosen, dan staf Program Studi Informatika – Program Sarjana yang telah memberikan ilmu, pengalaman, dan bantuan selama perkuliahan, terutama Ibu Arrie Kurniawardhani, S.Si., M.Kom. selaku dosen pembimbing akademik.
6. Seluruh pimpinan, staf, dan rekan kerja di mitra magang yang telah memberikan kesempatan dan pengalaman magang, terutama Mas Fandi, Mas Sean, dan Mbak Ariani.
7. Anisa Amalia, yang selalu setia mendengarkan keluh kesah saya selama kuliah.
8. Kartika Ningsih, teman seperjuangan dari SMA hingga kuliah.
9. Annida Rizki Luthfi Astuti dan Chilya Chumaira yang selalu menemani dan memberikan semangat dalam proses penyusunan laporan ini.
10. Teman-teman tercinta yang telah menjadi teman berbagi ilmu, cerita, dan pengalaman selama masa perkuliahan.

Laporan akhir ini masih jauh dari kesempurnaan sehingga diharapkan saran dan kritik dapat menyempurnakan laporan ini. Semoga laporan ini dapat bermanfaat bagi pembaca.

Yogyakarta, 12 Agustus 2022



(Yuwirdha Bektie Widiyandari)



SARI

Perkembangan internet telah mempercepat laju digitalisasi bank. Masyarakat saat ini dapat dengan mudah melakukan transaksi perbankan melalui Internet Banking, mulai dari pembayaran, transfer, maupun pembelian. Dalam penyediaan layanan menggunakan Internet Banking, dibutuhkan peran dari perangkat tengah (*middleware system*) yang membantu komunikasi antarsistem di internal dan eksternal perusahaan. Lingkup yang diangkat dalam laporan akhir ini adalah analisis dan perancangan pengembangan sistem integrasi *middleware* dalam transaksi menggunakan Internet Banking yang dimiliki sebuah perusahaan perbankan, yang selanjutnya disebut Bank. Internet Banking menjadi kanal transaksi yang paling banyak dikerjakan selama kegiatan magang. Terdapat empat dari sebelas proyek yang menggunakan Internet Banking sebagai kanal transaksi, antara lain Pembayaran Tagihan Perusahaan Gas Negara (PGN), Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB), Top Up Voucher melalui *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melalui *Online Travel Agent* (OTA).

Perancangan sistem integrasi *middleware* pada transaksi melalui Internet Banking yang dilakukan selama kegiatan magang menggunakan metode Waterfall SDLC, serta menggunakan pendekatan *model-driven analysis* yang menekankan visualisasi model dalam penggambaran dan pendokumentasian sistem. Tahapan perancangan yang dilakukan, antara lain analisis kebutuhan sistem, perancangan *inquiry*, *payment*, dan *reversal transaction*, serta perancangan skenario pengujian. Proses pengembangan sistem integrasi *middleware* untuk setiap proyek akan melibatkan *Collecting Agent Aggregator* (CAA), OTA, dan *Voucher Agent*. Perancangan sistem ini selanjutnya dapat dijadikan panduan bagi pengembang dan penguji dalam mengembangkan sistem pembayaran tagihan pada Internet Banking.

Kata kunci: CAA, internet banking, middleware, OTA, Voucher Agent

GLOSARIUM

Bitmap	komponen ISO 8583 yang memuat informasi elemen data yang digunakan
Biller	pihak yang memiliki tagihan yang harus dibayar nasabah
CA	<i>Collecting Agent</i> ; pengumpul dana tagihan yang dibayar oleh konsumen; Bank
CAA	<i>Collecting Agent Aggregator</i> ; pihak yang menjadi perantara antara CA dan Biller
Core banking	sistem pada perbankan yang bertanggung jawab dalam proses penarikan, pengkreditan, serta pencatatan transaksi
Data element	komponen ISO 8583 yang memuat informasi pesan yang ada dalam transaksi
Feasibility study	dokumen studi kelayakan
Host-to-host	sistem yang terhubung dua arah dan bersifat <i>realtime</i> antar institusi yang melakukan kerja sama
Interoperabilitas	suatu kemampuan berbagai ragam sistem atau aplikasi untuk bekerja sama dan bisa berinteraksi dengan aplikasi lainnya yang berbeda untuk memungkinkan terjadinya pertukaran data/informasi melalui suatu protokol yang disetujui bersama
Inquiry transaction	transaksi pengecekan informasi tagihan
Internet banking	produk/layanan perbankan yang menggunakan jaringan internet
ISO 8583	standar internasional yang berlaku untuk pertukaran pesan pada transaksi finansial
ISO Message Format	dokumen spesifikasi pertukaran data menggunakan ISO
Middleware	sistem yang menghubungkan setiap sistem dan aplikasi di internal dan eksternal perbankan
Model driven analysis	pendekatan yang menekankan visualisasi model dalam penggambaran dan pendokumentasian sistem
MTI	<i>Message Type Identifier</i> ; komponen ISO 8583 yang memuat tipe pesan yang akan dikirim/diterima

OTA	<i>Online Travel Agent</i> ; pihak <i>3rd Party</i> yang berperan dalam penyediaan hotel dalam proyek Pemesanan Hotel pada Fitur Perjalanan
Payment transaction	transaksi pembayaran tagihan
Rekonsiliasi	suatu proses pencocokan dan penyesuaian atas informasi catatan transaksi atau saldo kas antara catatan menurut nasabah dan bank
Reversal	pengembalian dana transaksi yang tidak berhasil
Suspect	transaksi yang tidak jelas informasi berhasil atau tidaknya
Switching	teknik menghubungkan dua atau lebih jaringan/sistem
System design	dokumen yang berisi spesifikasi serta desain fungsional dan teknis proyek
Test script	dokumen skenario pengujian
Voucher agent	pihak <i>3rd Party</i> yang berperan dalam penyediaan voucher dalam proyek <i>Top Up Voucher</i> pada Fitur Perjalanan
Waterfall SDLC	metode pengembangan perangkat lunak; salah satu metode <i>Software Development Life Cycle</i> yang dilakukan secara sekuensial mengalir ke bawah seperti air terjun

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Ruang Lingkup.....	3
1.2.1 Analisis <i>Existing Middleware System</i>	3
1.2.2 Pembayaran Tagihan Perusahaan Gas Negara (PGN)	4
1.2.3 Otomasi Jurnal <i>Fee Waiver</i>	4
1.2.4 Perubahan Cetak Bukti Setor Haji	4
1.2.5 Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB).....	4
1.2.6 Top Up Voucher pada Fitur Perjalanan	4
1.2.7 Pemesanan Hotel pada Fitur Perjalanan melalui <i>Online Travel Agent (OTA)</i>	5
1.3 Tujuan	5
1.4 Manfaat	5
1.5 Sistematika Penulisan	5
BAB II LANDASAN TEORI DAN TINJAUAN PUSTAKA	7
2.1 <i>Internet Banking</i>	7
2.2 <i>Middleware</i>	7
2.3 ISO 8583	10
2.4 Metode dan Pendekatan	12
2.5 Tinjauan Pustaka.....	17
BAB III PELAKSANAAN MAGANG.....	19
3.1 Manajemen Proyek	19

3.2 Analisis Kebutuhan Proyek.....	21
3.2.1 Proyek Pembayaran Tagihan Perusahaan Gas Negara (PGN).....	22
3.2.2 Proyek Pembayaran Tagihan Pajak Bumi dan Bangunan.....	23
3.2.3 Proyek Pemesanan Hotel pada Fitur Perjalanan melalui OTA.....	24
3.2.4 Proyek <i>Top Up</i> Voucher pada Fitur Perjalanan melalui <i>Voucher Agent</i>	25
3.3 Perancangan Sistem	27
3.3.1 Perancangan Transaksi <i>Inquiry</i>	28
3.3.2 Perancangan Transaksi <i>Payment</i>	30
3.3.3 Perancangan Skenario <i>Reversal</i>	34
3.3.4 Perancangan Skenario Pengujian.....	46
3.4 Hasil Analisis dan Perancangan Sistem	47
3.4.1 Dokumen ISO Message Format.....	47
3.4.2 Dokumen System Design.....	48
3.4.3 Dokumen Test Script	49
BAB IV REFLEKSI PELAKSANAAN MAGANG	50
4.1 Relevansi Akademik	50
4.1.1 Pengembangan Integrasi <i>Middleware</i>	50
4.1.2 Perancangan ISO Message Format	51
4.1.3 Manajemen Proyek dan Implementasi Waterfall SDLC	52
4.2 Pembelajaran Magang.....	53
4.2.1 Analisis dan Perancangan Sistem <i>Middleware</i> pada <i>Internet Banking</i>	53
4.2.2 Keahlian Teknis dan Non-Teknis	55
BAB V PENUTUP	57
4.3 Kesimpulan	57
4.4 Saran	57
DAFTAR PUSTAKA	58

DAFTAR TABEL

Tabel 1.1 Aktivitas kegiatan magang sebagai Middleware SA	2
Tabel 2.1 Format elemen data dengan ISO 8583	11
Tabel 2.2 Perbandingan antara model SDLC	15
Tabel 2.3 Hasil tinjauan pustaka	18
Tabel 3.1 Deskripsi pekerjaan pada Divisi Middleware	20
Tabel 3.2 Cerita pengguna fitur pembayaran tagihan PGN	22
Tabel 3.3 Cerita pengguna pengembangan fitur pembayaran tagihan PBB	23
Tabel 3.4 Cerita pengguna layanan pemesanan hotel pada fitur perjalanan	25
Tabel 3.5 Cerita pengguna <i>top up voucher</i> pada fitur perjalanan	26
Tabel 3.6 Rincian perancangan setiap proyek melalui Internet Banking	28
Tabel 3.7 Informasi data yang dipertukarkan pada transaksi <i>inquiry</i>	29
Tabel 3.8 Informasi data yang dipertukarkan pada transaksi <i>payment</i>	31
Tabel 3.9 Rincian <i>reversal</i> setiap proyek	34
Tabel 4.1 Klasifikasi tipe pesan pada MTI	51

DAFTAR GAMBAR

Gambar 2.1 Konsep Middleware sebagai sistem integrasi dalam perbankan.....	8
Gambar 2.2 CAA sebagai jembatan antara CA dan Biller	9
Gambar 2.3 Komponen utama ISO 8583	11
Gambar 2.4 Metode Waterfall SDLC	12
Gambar 3.1 Kegiatan rapat bersama tim Middleware	21
Gambar 3.2 Aliran bisnis sistem pembayaran tagihan PGN.....	23
Gambar 3.3 Aliran bisnis sistem pembayaran tagihan PBB	24
Gambar 3.4 Aliran bisnis pemesanan hotel pada fitur perjalanan	25
Gambar 3.5 Aliran bisnis sistem <i>top up</i> voucher pada fitur perjalanan.....	27
Gambar 3.6 Alur proses transaksi <i>inquiry</i>	30
Gambar 3.7 Alur proses transaksi <i>payment</i> pembayaran tagihan PBB dan PGN.....	32
Gambar 3.8 Alur proses transaksi <i>payment</i> untuk pemesanan hotel pada.....	33
Gambar 3.9 Alur proses transaksi <i>payment</i> untuk <i>top up</i> voucher melalui <i>Voucher Agent</i> ...	33
Gambar 3.10 Alur transaksi <i>timeout/no response</i> dari CAA	36
Gambar 3.11 Alur transaksi gagal/ditolak oleh CAA.....	37
Gambar 3.12 Alur transaksi pemesanan hotel pada fitur perjalanan	40
Gambar 3.13 Alur transaksi <i>top up</i> voucher pada fitur perjalanan	41
Gambar 3.14 Alur transaksi pemesanan hotel pada fitur perjalanan	42
Gambar 3.15 Alur transaksi <i>top up</i> voucher fitur perjalanan.....	43
Gambar 3.16 Alur transaksi <i>timeout/no response</i> dari Core Banking	45
Gambar 3.17 Alur transaksi gagal/ditolak oleh Core Banking	46

BAB I PENDAHULUAN

Bab ini membahas latar belakang, ruang lingkup, tujuan, manfaat, serta sistematika penulisan laporan akhir yang berjudul “Analisis Perancangan Sistem Integrasi *Middleware* pada Internet Banking”.

1.1 Latar Belakang

Kemajuan teknologi informasi yang beriringan dengan kemajuan ilmu pengetahuan dan peradaban manusia telah mengubah pola dan sistem kehidupan di masyarakat. Mobilitas masyarakat yang meningkat di era ini turut membuat berbagai bank di Indonesia berlomba-lomba menerapkan teknologi informasi ke sistemnya. Kualitas kinerja teknologi informasi pada suatu bank menjadi salah satu penentu keberhasilan bank (Syarif dkk., 2018). Era teknologi informasi mendukung berkembangnya sistem perbankan elektronik atau *e-banking*. Melalui *e-banking*, nasabah dapat melakukan transaksi menggunakan berbagai kanal elektronik, misalnya Anjungan Tunai Mandiri (ATM), *Electronic Data Capture* (EDC), *internet banking*, *Short Messaging Service* (SMS), dan *mobile banking* (Sailaja & Thamodaran, 2016).

Internet banking sebagai salah satu produk perbankan yang berkembang seiring dengan perkembangan internet juga menjadi bagian dari masyarakat yang tidak bisa dilepaskan. Masyarakat saat ini dapat dengan mudah melakukan pembayaran, pembelian, dan transfer melalui *internet banking* di mana saja dan kapan saja. Skala dan ruang lingkup internet telah berkembang menembus aspek kehidupan masyarakat dengan perannya sebagai perantara komunikasi dan transaksi (Perset, 2010). Internet membuat sektor perbankan beralih ke saluran transaksi yang bersifat otonom (Pikkarainen dkk., 2004). Pada penerapannya, layanan *internet banking* dalam sebuah bank melibatkan sistem *middleware*. *Middleware* memungkinkan sistem dan aplikasi pada internal dan eksternal institusi dapat saling berkomunikasi karena kemampuannya dalam menyediakan koneksi jarak jauh antarobjek (Sanjaya dkk., 2015).

Laporan akhir ini disusun berdasarkan kegiatan magang sebagai *Middleware System Analyst* (*Middleware SA*) pada sebuah lembaga keuangan bank swasta yang terletak di kawasan Bintaro, Tangerang, yang selanjutnya disebut Bank (perusahaan tidak berkenan

disebutkan namanya). Kegiatan magang ini merupakan salah satu program Merdeka Belajar Kampus Merdeka (MBKM) yang dilaksanakan oleh Kementerian Pendidikan dan Kebudayaan (Kemdikbud). Pelaksanaan magang dilakukan selama enam bulan (Agustus 2021 – Februari 2022) dengan divisi penempatan, yaitu Divisi Middleware di bawah naungan IT Application Development. Divisi Middleware mengelola sistem *middleware* yang berperan dalam integrasi antarsistem dan layanan yang dimiliki Bank, termasuk dalam penyediaan layanan Internet Banking.

Selama kegiatan magang, telah dilakukan kontribusi terhadap tujuh proyek yang ada pada Divisi Middleware seperti pada Tabel 1.1.

Tabel 1.1 Aktivitas kegiatan magang sebagai Middleware SA

Kegiatan	Timeline	Rincian Aktivitas
<i>On-boarding</i> Magang	23 Agustus 2021 – 23 September 2021	<ul style="list-style-type: none"> • Penjelasan mengenai profil perusahaan, <i>learning framework</i> perusahaan, dan panduan magang • Menyelesaikan modul pada aplikasi yang disediakan perusahaan terkait pengenalan aturan dan panduan sebagai karyawan • Pembelajaran tentang sistem <i>middleware</i> dan proses proses pertukaran pesan
<i>Analisis Existing Middleware System</i>	Agustus – November 2021	<ul style="list-style-type: none"> • Menganalisis alur transaksi yang sedang berjalan di Middleware • Memberikan solusi/saran terhadap temuan analisis • Membuat visualisasi alur transaksi ke diagram alir • Mendokumentasikan hasil analisis
Pembayaran Tagihan Perusahaan Gas Negara (PGN) pada ATM dan Internet Banking	20 September – 25 September 2021	<p>Dokumen ISO 8583 Message Format</p> <ul style="list-style-type: none"> • Membuat dokumen panduan pertukaran pesan antar sistem di Middleware <p>Dokumen System Design</p> <ul style="list-style-type: none"> • Menerjemahkan spesifikasi teknis ke pengembang • Membuat diagram alir yang sesuai dengan solusi yang akan dibangun • Membuat gambaran umum solusi, cerita pengguna, serta desain fungsional dan teknis <p>Dokumen Test Script</p> <ul style="list-style-type: none"> • Membuat dokumen panduan pengujian yang akan dilakukan oleh Testing Engineer agar dapat menjamin kualitas proyek secara keseluruhan.
Perubahan Bukti Setor Haji	28 September 2021 – 8 Oktober 2021	
Otomasi Jurnal <i>Fee Waiver</i>	19 Oktober 2021 – November 2021	
Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB) pada Internet Banking	6 Desember 2021 – Januari 2022	
Top Up Voucher pada Fitur Perjalanan Internet Banking	6 Desember 2021 – Januari 2022	
Pemesanan Hotel pada Fitur Perjalanan melalui OTA	Januari – Februari 2022	

Laporan akhir ini membahas analisis dan perancangan pengembangan pada kanal Internet Banking pada lembaga keuangan Bank. Internet Banking dipilih menjadi kanal transaksi yang diangkat pada laporan ini karena paling banyak dikerjakan selama kegiatan magang. Terdapat empat dari tujuh proyek yang menggunakan Internet Banking sebagai kanal transaksi, antara lain Pembayaran Tagihan Perusahaan Gas Negara (PGN), Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB), *Top Up Voucher* pada Fitur Perjalanan Internet Banking melalui *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melalui *Online Travel Agent* (OTA).

Selain itu, cakupan analisis dan perancangan lainnya yang dibahas dalam laporan adalah pengembangan *middleware* sebagai aplikasi yang mengintegrasikan Bank dengan seluruh layanan perbankan yang ada di internal dan eksternal Bank. Bank bekerja sama dengan berbagai layanan pembayaran sehingga masyarakat dapat dengan mudah melakukan transaksi, misalnya *Collecting Agent Aggregator* (CAA), *Voucher Agent*, dan OTA. Pada sebuah transaksi, tidak hanya satu sistem saja yang terlibat, namun juga beberapa sistem dan aplikasi yang ada di Bank. Untuk mengintegrasikan aplikasi-aplikasi internal dan eksternal tersebut, Bank menggunakan bantuan piranti tengah/*middleware*. Perancangan sistem integrasi *middleware* ini selanjutnya dapat dijadikan panduan bagi pengembang dan penguji dalam mengembangkan sistem pembayaran tagihan pada Internet Banking.

1.2 Ruang Lingkup

Selama menjalani magang enam bulan sebagai *Middleware SA*, telah dilakukan kontribusi terhadap tujuh proyek. Namun dari tujuh proyek tersebut, hanya empat proyek yang akan dibahas secara detail pada laporan akhir ini, yaitu proyek yang menggunakan Internet Banking sebagai kanal transaksi. Keempat proyek tersebut di antaranya, Pembayaran Tagihan Perusahaan Gas Negara (PGN), Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB), *Top Up Voucher* pada Fitur Perjalanan melalui *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melalui OTA.

1.2.1 Analisis Existing Middleware System

Pada proyek ini dilakukan analisis *existing transaction flow* pada sistem *middleware* yang ada di Bank. Melalui analisis transaksi ditemukan *issues* pada alur transaksi yang saat ini sedang berjalan pada sistem *middleware*. Temuan analisis ini nantinya akan menjadi pertimbangan tim *development* untuk melakukan perbaikan.

1.2.2 Pembayaran Tagihan Perusahaan Gas Negara (PGN)

Proyek pembayaran tagihan PGN pada Internet Banking dan ATM adalah proyek yang bertujuan untuk mengembangkan sistem Bank untuk pembayaran tagihan PGN secara digital melalui Internet Banking dan ATM. Transaksi pembayaran tagihan PGN ini dilakukan secara H2H (*host-to-host*) menggunakan *switching 3rd Party* yang terhubung dengan sistem *middleware*.

1.2.3 Otomasi Jurnal *Fee Waiver*

Proyek Otomasi Jurnal *Fee Waiver* berisi solusi perubahan proses penjurnalan *fee* transaksi yang dibebaskan secara otomatis ketika nasabah melakukan proses transaksi. Proses pejurnalan sebelumnya dilakukan secara manual oleh *related business unit* dan memakan waktu lebih dari satu bulan. Melalui otomasi jurnal *fee waiver*, proses penjurnalan akan menghemat waktu.

1.2.4 Perubahan Cetak Bukti Setor Haji

Pada proyek ini terdapat pengembangan pada sistem *middleware* sehingga dapat mendukung penyesuaian terhadap pencetakan bukti setor haji. Bukti setor haji merupakan bukti atas transaksi setoran awal Biaya Penyelenggaraan Ibadah Haji (BPIH) Reguler yang saat ini telah berjalan di Bank. Perubahan yang dilakukan, yaitu bukti setor haji berupa *textfile* yang semula dicetak pada kertas *pre-printed* menjadi berkas dengan format PDF.

1.2.5 Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB)

Proyek ini mendukung pengembangan sistem *middleware* dalam transaksi pembayaran tagihan PBB (Pajak Bumi dan Bangunan) pada Internet Banking. Saat ini pembayaran tagihan pada Internet Banking belum menyediakan pembayaran untuk tagihan PBB. Transaksi pembayaran tagihan PBB ini dilakukan secara H2H menggunakan *switching 3rd Party* yang terhubung dengan sistem *middleware*.

1.2.6 Top Up Voucher pada Fitur Perjalanan

Pada proyek ini terdapat pengembangan sistem *middleware* yang mendukung penambahan menu untuk *top up voucher* di Internet Banking. Menu *top up voucher* ini menu tambahan pada fitur Perjalanan yang ada di Internet Banking. Transaksi *top up voucher* pada Internet Banking diintegrasikan dengan *3rd Party*, yaitu *Voucher Agent*, yang terhubung langsung dengan sistem *middleware*.

1.2.7 Pemesanan Hotel pada Fitur Perjalanan melalui *Online Travel Agent* (OTA)

Pada proyek ini terdapat pengembangan sistem *middleware* untuk mendukung transaksi pemesanan hotel melalui Internet Banking. Proyek ini bekerja sama dengan pihak ketiga, yaitu *Online Travel Agent* (OTA) yang terhubung dengan Bank. Nasabah dapat memilih hotel sesuai dengan tanggal, ruangan, dan area yang diinginkan, serta dapat melakukan pengecekan status dan riwayat *booking*.

1.3 Tujuan

Laporan akhir ini bertujuan untuk melakukan analisis dan perancangan sistem integrasi *middleware* proses transaksi melalui kanal Internet Banking. Selain itu, laporan akhir ini dapat menjadi panduan pengembang dan penguji dalam melakukan pengembangan dan pengujian proyek.

1.4 Manfaat

Manfaat yang diperoleh dari perancangan sistem integrasi *middleware* melalui kanal Internet Banking, di antaranya:

- a. Mengetahui perancangan sistem dari sisi *middleware* dalam pengembangan sistem pembayaran di perbankan melalui kanal Internet Banking
- b. Mengetahui sistematika perancangan yang dapat dijadikan referensi oleh pengembang dan penguji

1.5 Sistematika Penulisan

Sistematika penulisan berisi gambaran umum laporan akhir. Adapun sistematika penulisan laporan akhir ini sebagai berikut:

- a. BAB I: Pendahuluan
Bab ini berisi latar belakang, ruang lingkup magang, tujuan, manfaat, dan sistematika penulisan laporan.
- b. BAB II: Landasan Teori dan Tinjauan Pustaka
Bab ini berisi landasan teori dan tinjauan pustaka yang dikaji dari penelitian-penelitian yang berhubungan dengan topik laporan.
- c. BAB III: Pelaksanaan Magang
Bab ini membahas pelaksanaan magang dan pembahasan perancangan sistem integrasi *middleware* pada transaksi melalui Internet Banking secara lebih rinci.
- d. BAB IV: Refleksi Pelaksanaan Magang

Bab ini membahas refleksi pelaksanaan magang sebagai Middleware SA berupa relevansi akademik dan pembelajaran magang.

e. BAB V: Kesimpulan dan Saran

Bab ini membahas kesimpulan dari hasil analisis dan perancangan sistem integrasi *middleware* pada proyek yang menggunakan kanal transaksi Internet Banking. Bab ini juga berisi saran yang diberikan bagi peneliti lainnya yang akan melakukan penelitian serupa di masa depan.



BAB II

LANDASAN TEORI DAN TINJAUAN PUSTAKA

Bab ini membahas landasan teori yang digunakan untuk penelitian. Secara umum, laporan akhir yang berjudul “Analisis Perancangan Sistem Integrasi *Middleware* pada Internet Banking” membahas hal-hal terkait *internet banking*, *middleware*, serta ISO 8583. Penggunaan *Waterfall Software Development Life Cycle* (SDLC) dan pendekatan *model-driven analysis* yang diterapkan pada kegiatan magang juga akan dijelaskan pada bab ini. Selain itu, untuk memperkuat laporan, dilakukan tinjauan pustaka terhadap penelitian-penelitian sebelumnya dengan topik terkait dan menganalisis kesenjangannya (*gap analysis*) dengan proyek yang dilakukan.

2.1 Internet Banking

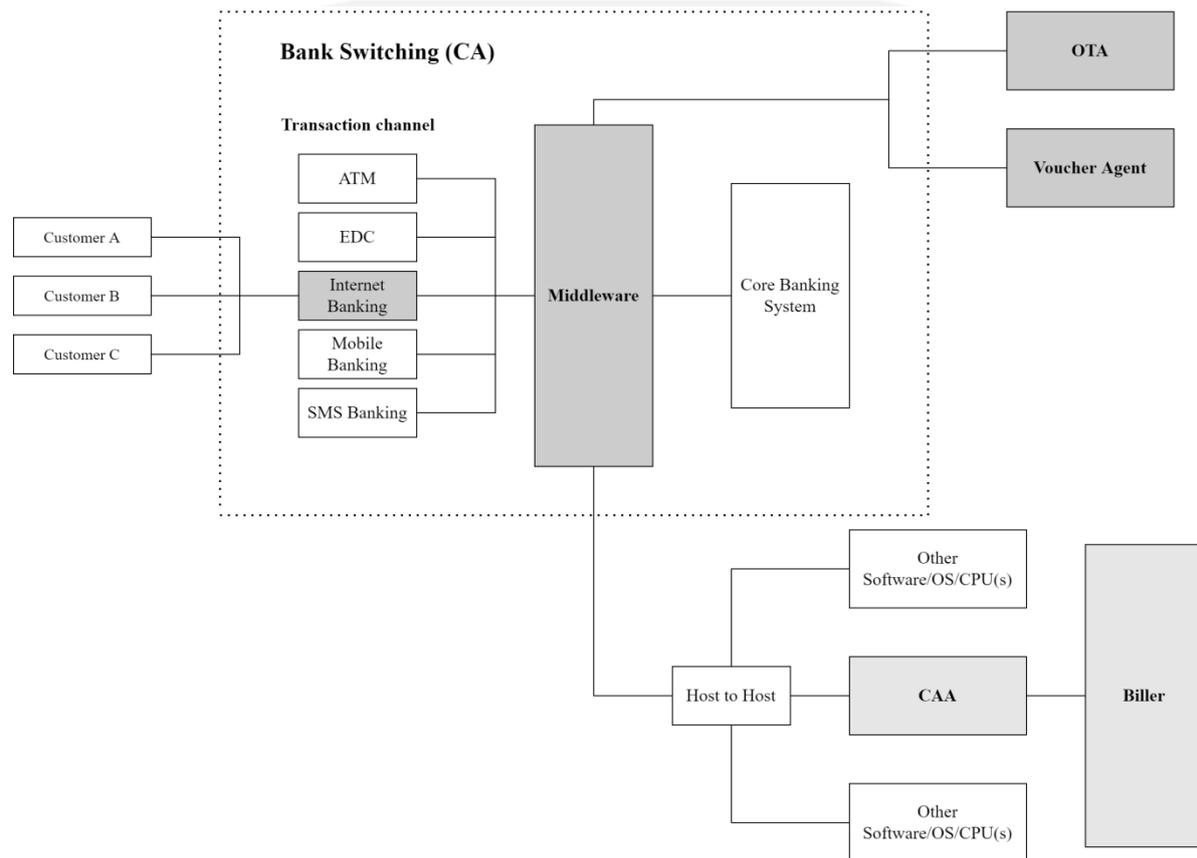
Internet banking merupakan suatu istilah penggunaan internet sebagai saluran pengiriman layanan perbankan. Seiring dengan maraknya penggunaan internet, Bank turut memperluas cakupan layanan perbankan elektroniknya, salah satunya dengan menyediakan layanan Internet Banking. Hadirnya Internet Banking membuat persaingan lembaga keuangan perbankan semakin kompetitif dan transparan (Margaretha, 2015).

Layanan Internet Banking yang dimiliki Bank mendukung masyarakat dalam melakukan transaksi perbankan kapan saja dan di mana saja selama terhubung dengan internet, tanpa harus mendatangi Bank. Selain itu, Internet Banking juga menawarkan berbagai kemudahan, di antaranya: melakukan seluruh transaksi perbankan (transfer, pembayaran tagihan, atau pembelian); mendapatkan informasi terkait saldo, informasi nilai tukar, dan mutasi rekening; melihat ringkasan dan catatan transaksi; serta pembukaan rekening tabungan secara *online*. Internet Banking juga dapat memberikan manfaat terhadap bank yang menjalankan, di antaranya loyalitas nasabah (Heryani dkk., 2020).

2.2 Middleware

Middleware digambarkan sebagai sebuah lapisan perangkat lunak yang berada di antara aplikasi atau lapisan teknologi (Kassab & Darabkh, 2020) yang mengintegrasikan seluruh aplikasi dan proses bisnis yang beroperasi pada suatu *enterprise* (Al-Ghamdi & Saleem, 2014). *Middleware* didefinisikan sebagai sebuah perangkat yang menyediakan interoperabilitas di antara aplikasi dan perangkat yang tidak saling kompetibel (da Cruz dkk.,

2018; Kortuem dkk., 2010) dan menjadi jembatan di antaranya sehingga sistem dapat berjalan secara efisien (Farahzadi dkk., 2018). Pada lembaga keuangan bank, *middleware* akan melakukan integrasi terhadap layanan-layanan dari institusi lain untuk mendukung produk dan layanan perusahaan. *Middleware* menjadi sistem yang umum digunakan oleh perusahaan sesuai dengan standar sistem informasi yang berjalan di setiap perusahaan. Konsep *middleware* pada Bank secara umum dapat dilihat pada Gambar 2.1.



Gambar 2.1 Konsep Middleware sebagai sistem integrasi dalam perbankan

Sumber: Bank (2011)

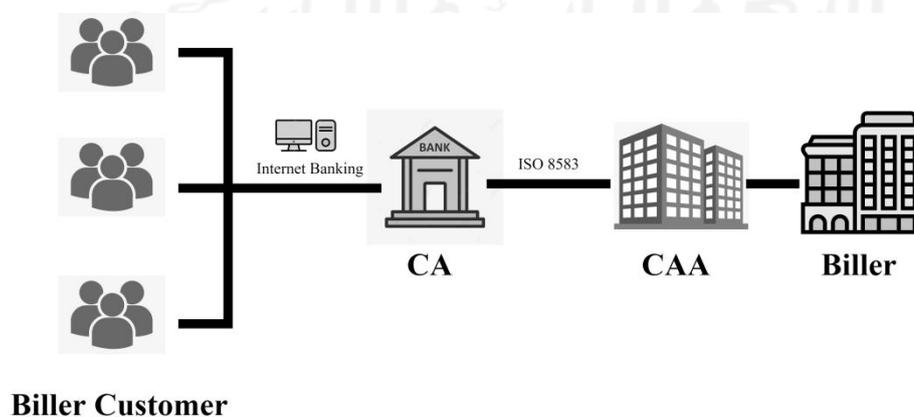
Bank mengadopsi sistem *middleware* sebagai sistem integrasi. Dengan adanya *middleware*, tercipta transparansi pada seluruh jaringan sehingga Bank dapat dengan mudah melakukan interaksi dan kolaborasi ke aplikasi klien dan *server* (Hasibuan & Soecahyadi, 2020). Pada kegiatan perbankan, *middleware* menghubungkan Bank dengan berbagai kanal elektronik yang ada di Bank, Core Banking, hingga pihak eksternal. Pihak eksternal dalam hal ini merupakan pihak yang bekerja sama dengan Bank dalam pembayaran tagihan. Beberapa istilah yang perlu diketahui dalam kerja sama dalam penyediaan layanan Internet

Banking pada laporan ini, antara lain *Collecting Agent (CA)*, *Biller*, *Collecting Agent Aggregator (CAA)*, *Voucher Agent*, dan *Online Travel Agent (OTA)*.

Collecting Agent (CA) adalah pengumpul dana tagihan yang dibayar oleh konsumen. CA dapat berupa perusahaan bank atau non-Bank. Pada analisis dan perancangan sistem integrasi *middleware*, yang menjadi pengumpul dana adalah Bank. Selain CA, terdapat pihak yang memiliki tagihan yang harus dibayar oleh pelanggan, yang disebut *Biller*. Tagihan yang dibayarkan dapat berupa tagihan rutin atau sekali bayar. Namun, CA dalam transaksi pembayaran tagihan tidak berkomunikasi langsung dengan *Biller*. CAA akan menjadi jembatan dalam transaksi pembayaran ini.

Collecting Agent Aggregator (CAA) adalah pihak yang menjadi perantara antara pihak yang melakukan penerimaan pembayaran (CA) dan pihak yang memiliki tagihan (*Biller*). Pada sistem pembayaran yang disediakan oleh Bank, pembayaran tagihan, transfer, dan pembelian saat ini dilakukan melalui berbagai kanal perbankan (*Teller*, *ATM*, *EDC*, *Internet Banking*, atau *Mobile Banking*), aplikasi *online payment*, atau *payment point* lainnya. Namun, akan sulit bagi CA/Bank apabila harus melakukan kegiatan transaksi satu per satu dengan berbagai *Biller* sehingga diperlukan CAA sebagai jembatan yang membantu.

CAA akan membantu komunikasi antara *Biller* dan CA sehingga pembayaran dapat dengan mudah dilakukan. Caranya, yakni dengan melakukan koneksi secara *host-to-host* dengan CA dan *Biller* yang telah memilih *billing system* sehingga dapat diintegrasikan seperti pada Gambar 2.2. Komunikasi antara CA dan CAA tentunya harus disepakati terlebih dahulu. Pada perancangan sistem integrasi *middleware* untuk transaksi *Internet Banking*, CAA menjadi jembatan komunikasi dalam proyek *Pembayaran Tagihan PGN* dan *Pembayaran Tagihan PBB*.



Gambar 2.2 CAA sebagai jembatan antara CA dan *Biller*

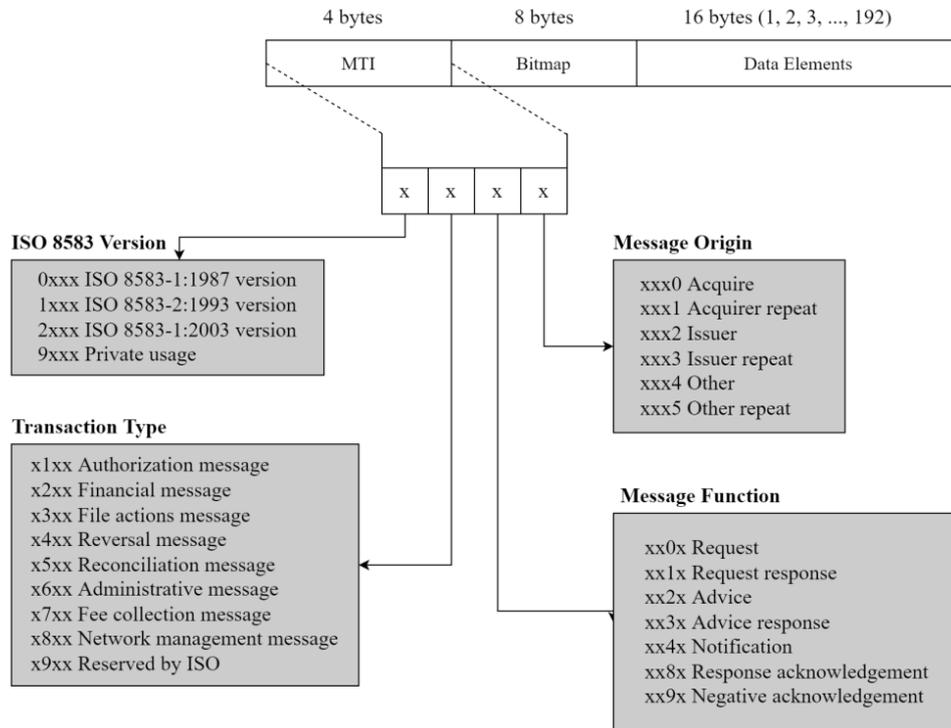
Sumber: Solution (2018)

Middleware pada perancangan sistem integrasi *middleware* melalui Internet Banking juga berkomunikasi dengan *Voucher Agent* dan OTA. Komunikasi ini dilakukan langsung dan tidak membutuhkan jembatan CAA. *Voucher Agent* merupakan agen yang menyediakan voucher yang akan diintegrasikan ke sistem Internet Banking sehingga nasabah bisa melakukan *top up* voucher. Begitupula dengan OTA yang menyediakan berbagai layanan perjalanan *online*, salah satunya pemesanan hotel sehingga nasabah bisa melakukan transaksi tersebut melalui Internet Banking.

2.3 ISO 8583

ISO 8583 adalah standar internasional yang berlaku untuk pertukaran pesan pada transaksi finansial (Standardization, 1987). Dalam prosesnya, ISO 8583 akan mendefinisikan format pertukaran data pada seluruh rantai transaksi di lembaga keuangan sehingga memungkinkan perangkat lunak dan perangkat keras bertukar pesan, baik berupa pesan *request* maupun *response*. Jenis transaksi yang didefinisikan bermacam-macam, misalnya pembelian, pembayaran tagihan, penarikan tunai, pengembalian dana, dll. Melalui ISO 8583, Bank dapat berkomunikasi dengan lembaga di luar Bank, misalnya bank lain, CAA, maupun pihak *interchange* (pihak ketiga yang memiliki kerja sama keuangan) (Kuntoro & Ladjamuddin, 2019). ISO 8583 memiliki tiga komponen utama yang terdiri dari *Message Type Identifier* (MTI), *Bitmap*, dan *Data Elements*. Gambar 2.3 menunjukkan komponen utama pada ISO 8583.

Message Type Identifier (MTI) memuat tipe pesan yang akan dikirimkan/diterima. Melalui *identifier* ini, pesan yang dikirimkan dapat diidentifikasi arah dan server yang melakukan *request/response*. Standar untuk MTI sudah diatur oleh ISO 8583 seperti pada Gambar 2.3. Panjang MTI adalah 4 byte. *Bitmap* memuat informasi mengenai *fields/subfields* yang menunjukkan elemen data yang digunakan. *Bitmap* direpresentasikan dengan 8 byte data biner atau 16 karakter heksadesimal (0-9, A-F). *Bitmap* terdiri dari *bitmap primer* (data 1-64) yang bersifat wajib dan *bitmap sekunder* (data 65-192) yang bersifat opsional. *Data Elements* memuat informasi pesan yang ada dalam transaksi. Setiap elemen data memiliki arti dengan format tertentu yang telah disusun sesuai dengan standar penulisan ISO 8583 seperti pada Tabel 2.1 (Standardization, 1987) sehingga dapat diimplementasikan. Panjang elemen data adalah 16 *byte*.



Gambar 2.3 Komponen utama ISO 8583

Sumber: Mian dkk. (2015)

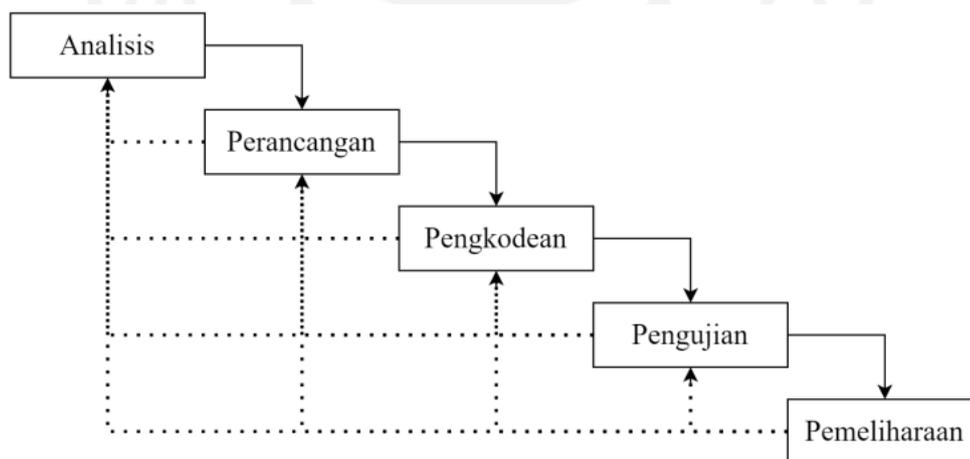
Tabel 2.1 Format elemen data dengan ISO 8583

Format	Deskripsi	Nilai
N	Numerik	
ANS	Alfanumerik dan karakter khusus	
AN	Alfanumerik	
M	<i>Mandatory</i>	
O	<i>Optional</i>	
C	<i>Conditional</i>	
<i>Blank/Space</i>	<i>Bit Off/Should not send</i>	
MM	Bulan	01-12
DD	Hari	01-31
YY	Tahun	<i>Last 2 Digits</i>
Hh	Jam	00-23
Mm	Menit	00-59
Ss	Detik	00-59
LVAR	Panjang Variabel	0-9
LLVAR	Panjang Variabel	00-99
LLLVAR	Panjang Variabel	000-999
LPS	<i>Left Justified Padded With Spaces</i>	<i>Usually for Alphanumeric Data</i>
RPZ	<i>Right Justified Padded With Zeros</i>	<i>Usually for Numeric Data</i>

2.4 Metode dan Pendekatan

Bank sebagai perusahaan perbankan akan terus mengembangkan sistem yang sedang berjalan agar sistem menjadi lebih baik sehingga diperlukan model *Software Development Life Cycle* (SDLC) yang tepat. SDLC adalah metode pengembangan berupa tahapan-tahapan yang dilakukan oleh analis sistem, pengembang, teknisi, dan penguji dalam pengembangan suatu sistem. SDLC bertujuan untuk memastikan proses penerjemahan tujuan sistem ke dalam proses operasional sesuai dengan jadwal dan anggaran. Pada kegiatan magang sebagai Middleware SA, tahapan SDLC yang dilakukan adalah analisis dan perancangan sistem.

Metode yang digunakan dalam pengembangan pada kegiatan magang, yaitu Waterfall SDLC. Waterfall SDLC merupakan sebuah proses pengembangan sistem yang dilakukan secara sekuensial mengalir ke bawah seperti air terjun. Metode Waterfall SDLC sederhana, metedis, serta mudah dipahami dan diimplementasikan sehingga cenderung dapat bertahan dari waktu ke waktu (Kramer, 2018).



Gambar 2.4 Metode Waterfall SDLC

Sumber: Bassil (2015)

Metode Waterfall digagas oleh Dr. Winston W. Royce pada tahun 1970 yang mengemukakan bahwa pengembangan sistem dapat dilakukan secara sistematis dan berurutan. Langkah umum yang menjadi inti dari pengembangan sistem adalah analisis dan pengkodean (Royce, 2021). Namun, langkah ini tidak cukup jika proyek yang dilakukan lebih besar, misalnya dalam skala *enterprise*, sehingga diperlukan langkah yang lebih ekstensif.

Jika digambarkan, secara umum metode Waterfall SDLC terdiri dari proses analisis, perancangan, pengkodean, test, dan pemeliharaan seperti Gambar 2.4. Tahapan analisis dan perancangan merupakan tahapan rekayasa sistem. Setiap tahap pada model Waterfall harus

diselesaikan terlebih dahulu sebelum berpindah ke tahap berikutnya. Hal ini membuat model Waterfall dikatakan bersifat rekursif yang memungkinkan setiap tahap dapat diulang tanpa henti sampai sempurna (Bassil, 2015).

Setiap model SDLC tentunya memiliki tingkat risiko dan kegunaan yang berbeda dalam pengembangan sistem. Ketepatan penggunaan SDLC tergantung pada kebutuhan perusahaan. Beberapa model SDLC lainnya yang bisa dibandingkan dengan Waterfall untuk diimplementasikan ke Divisi Middleware, misalnya model *prototype*, *spiral*, dan *agile*. Untuk lebih jelas, perbandingan antarmodel dapat dilihat pada Tabel 2.2.

Model *waterfall*, seperti namanya, merupakan proses pengembangan sistem yang berurutan. Model Waterfall SDLC menjadi pilihan yang tepat apabila persyaratan sistem yang dibutuhkan jelas dan tidak berubah-ubah (Amlani, 2012). Waterfall SDLC digunakan pada pengembangan proyek yang ada di Divisi Middleware karena persyaratan sistem yang jelas di awal proyek seperti yang tertuang dalam dokumen *feasibility study*.

Model *prototype* cocok digunakan pada pengembangan sistem apabila pengguna akhir memiliki banyak interaksi dengan sistem (Kumar, 2018). Jika diimplementasikan pada sistem yang memiliki sedikit interaksi pengguna, misalnya sistem pemrosesan yang banyak melakukan kalkulasi, prototipe tidak tepat digunakan. Sistem yang tidak banyak berinteraksi dengan pengguna biasanya melalui tahapan pengkodean yang intensif sehingga potensi keuntungan yang akan diperoleh melalui prototipe terlalu kecil (Salve dkk., 2018). Begitu pula dengan sistem *middleware* yang merupakan sistem yang berada di lapisan tengah sebagai jembatan komunikasi dan manajemen transaksi, model *prototype* tidak cocok digunakan karena sistem *middleware* tidak berinteraksi langsung dengan pengguna (nasabah).

Model *spiral* cocok digunakan pada proyek yang besar dengan kompleksitas dan risiko menengah hingga tinggi (Adel & Abdullah, 2015). Jika dilihat dari proyek yang ada di *middleware*, penggunaan model *spiral* kurang tepat karena proyek-proyek yang ada di *middleware* merupakan proyek *enhancement* dari sistem yang telah ada. Selain itu, apabila diimplementasikan ke pengembangan sistem *middleware*, model ini memungkinkan estimasi waktu yang lebih lama dibanding Waterfall karena kompleksnya tahapan yang harus dilalui proyek.

Model *agile* cocok digunakan pada pengembangan dengan persyaratan proyek yang memiliki kecenderungan berubah-ubah sesuai dengan kebutuhan pengguna (Dubey dkk., 2015). Hal ini didukung oleh kemampuan *agile* untuk beradaptasi menanggapi perubahan terhadap pengembangan (Kumar, 2018). Namun, model ini kurang tepat digunakan pada

sistem *middleware* karena persyaratan yang selalu berubah sehingga hampir tidak ada penekanan yang diletakkan pada proses perancangan dan dokumentasi. Tidak adanya penekanan pada proses perancangan dan dokumentasi memungkinkan keluar dari jalur (McCormick, 2012). Pada pengembangan sistem di Bank, setiap kebutuhan proyek secara umum telah didefinisikan kebutuhannya pada dokumen *feasibility study* dan setiap proyek *middleware* biasanya merupakan *enhancement* dari sistem yang telah ada, sehingga diperlukan adanya dokumentasi untuk mengetahui batas pengembangan.

Selain penggunaan metode Waterfall SDLC, pendekatan *model-driven analysis* juga digunakan, terutama pada proses analisis dan perancangan. Pendekatan *model-driven analysis* menekankan visualisasi model dalam penggambaran dan pendokumentasian sistem. Pendekatan *model-driven* akan memudahkan kerja *developer* dengan tugas pengkodean yang kompleks (Deddy dkk., 2021). Tiga jenis teknik yang sering digunakan pada *model-driven analysis*, di antaranya analisis terstruktur (*structured analysis*), rekayasa informasi (*information engineering*), dan analisis berorientasi objek (*object-oriented analysis*) (Mulyani, 2016).

Teknik *model-driven analysis* yang diadopsi pada kegiatan magang adalah analisis terstruktur yang berfokus pada aliran data, regulasi bisnis, dan proses bisnis. Teknik ini merepresentasikan data melalui grafik, diagram, dan DFD (*data flow diagram*). Pada proses di kegiatan magang, Middleware SA melakukan analisis terhadap *feasibility study* proyek yang ada dan difokuskan pada analisis sistem yang akan dirancang, yaitu integrasi *middleware*, untuk menentukan kebutuhan sistem. Setiap detail kebutuhan yang telah dianalisis selanjutnya akan dipetakan, divisualisasikan (misalnya, dengan *data flow sequence diagram*), dan didokumentasikan sehingga dapat menjadi panduan pengembang. Ruang lingkup perancangan melalui pendekatan *model-driven analysis* ini meliputi perancangan *inquiry*, *payment*, dan *reversal transaction*.

Tabel 2.2 Perbandingan antara model SDLC

Model SDLC	Waktu penggunaan	Kelebihan	Kekurangan
<i>Waterfall</i>	<ul style="list-style-type: none"> • Ketika persyaratan proyek jelas dan tidak berubah-ubah • Ketika <i>existing system</i> telah tersedia 	<ul style="list-style-type: none"> • Persyaratan sistem yang dibutuhkan jelas dan tidak berubah • Titik awal dan akhir proyek telah pasti • Kesalahan dan kesalahpahaman setiap tahapan dapat dideteksi dari awal • Membutuhkan dokumen spesifikasi kebutuhan yang berfungsi sebagai pedoman untuk tahap pengembangan dan pengujian • Setiap tahapan bersifat diskrit sehingga setiap tahap dipastikan sempurna sebelum masuk ke tahapan selanjutnya • Model Waterfall SDLC efisien jika anggota tim tersebar di lokasi yang berbeda • Sumber daya yang dibutuhkan lebih rendah 	<ul style="list-style-type: none"> • Setiap tahapan yang telah selesai artinya telah terkunci dan tidak bisa diperbaiki • Asumsi yang salah dapat membuat proyek gagal selesai sesuai dengan tenggat waktu yang telah ditentukan • Tidak fleksibel karena tidak memungkinkan perubahan sesuai kebutuhan klien • Waktu dan biaya dapat terus meningkat jika klien terus menambahkan kebutuhan • Anggota tim yang telah menyelesaikan tanggung jawabnya dalam fasenya cenderung duduk diam, kecuali anggota tim yang berada di bawah fase kerja yang sedang berlangsung
<i>Prototype</i>	<ul style="list-style-type: none"> • Ketika pengguna akhir memiliki banyak interaksi dengan sistem yang akan dirancang • Untuk pengembangan sistem yang banyak membangun antarmuka 	<ul style="list-style-type: none"> • Adanya keterlibatan pengguna dalam siklus pengembangan sehingga pengguna juga mendapatkan pemahaman dan pengalaman tentang sistem yang sedang dikembangkan • Kesalahan dan fungsionalitas yang hilang dapat dideteksi jauh lebih awal • Umpan balik pengguna lebih cepat didapatkan sehingga solusi juga lebih cepat ditemukan 	<ul style="list-style-type: none"> • Secara praktis, metodologi ini dapat meningkatkan kompleksitas sistem karena ruang lingkup sistem meluas seiring dengan umpan balik dari pengguna dan perbaikan • Prototipe yang tidak lengkap menyebabkan aplikasi tidak dapat dirancang secara lengkap sehingga memungkinkan aplikasi tidak dapat digunakan sesuai dengan rancangan • Sistem yang tidak banyak berinteraksi dengan pengguna biasanya melalui tahapan pengkodean yang intensif sehingga potensi keuntungan yang akan diperoleh melalui prototipe terlalu kecil
<i>Spiral</i>	<ul style="list-style-type: none"> • Untuk proyek dengan kompleksitas dan risiko menengah hingga tinggi • Ketika diharapkan 	<ul style="list-style-type: none"> • Adanya jumlah analisis risiko yang tinggi sehingga jika ada risiko yang tidak dapat diatasi dapat diindikasikan dari awal • Adanya persetujuan dan kontrol dokumentasi yang kuat 	<ul style="list-style-type: none"> • Perlunya tenaga yang ahli dalam analisis dan penilaian risiko • Banyaknya dokumentasi membuat pengelolaan proyek menjadi sangat kompleks • Waktu yang diperlukan untuk mengevaluasi risiko

	<p>adanya perubahan signifikan pada proyek</p> <ul style="list-style-type: none"> • Cocok digunakan untuk mengembangkan <i>customized product</i>. 	<ul style="list-style-type: none"> • Fungsionalitas tambahan dapat ditambahkan di kemudian hari • Pemantauan proyek yang mudah dan efektif • Umpan balik yang sering dari pengguna 	<p>proyek kecil dan berisiko rendah terlalu besar</p> <ul style="list-style-type: none"> • Waktu yang dihabiskan untuk tahap perencanaan, pengaturan ulang tujuan, analisis risiko, dan prototipe mungkin sangat lama • Keberhasilan proyek bergantung pada fase analisis risiko
<i>Agile</i>	<ul style="list-style-type: none"> • Ketika persyaratan proyek yang cenderung berubah-ubah sesuai kebutuhan pengguna • Ketika tidak ada gambaran dengan jelas seperti apa produk akhir akan terbentuk • Ketika kecepatan lebih penting daripada keunggulan produk 	<ul style="list-style-type: none"> • Fungsionalitas dapat dikembangkan dengan cepat dan diverifikasi secara berdampingan • Adaptif untuk lingkungan yang berubah secara progresif • Sedikit atau tidak ada perancangan yang dilakukan, <i>development</i> dan <i>deployment</i> dilakukan secara berdampingan 	<ul style="list-style-type: none"> • Dokumentasi sangat kurang sehingga ketergantungan terhadap individu meningkat (misalnya, <i>developer</i> yang mengembangkan sistem) • Sangat bergantung pada interaksi dengan pelanggan sehingga apabila keinginan pelanggan tidak jelas akan menciptakan ambiguitas antara pengembang • Sulit untuk mempertahankan kebutuhan karena salah satu tahapannya sebenarnya tidak benar-benar selesai • Tidak adanya penekanan pada proses perancangan dan dokumentasi memungkinkan proyek keluar dari ruang lingkup

2.5 Tinjauan Pustaka

Terdapat beberapa penelitian yang telah dilakukan sebelumnya mengenai pengembangan sistem pembayaran pada bank yang melibatkan ISO 8583. Penelitian sebelumnya tentang pengembangan sistem pembayaran elektronik PBB secara *host-to-host* menggunakan ISO 8583 pada pemerintah daerah dibahas oleh penelitian (Titi & Sari, 2017) yang menerapkan konsep komunikasi data yang dibangun secara *private* antara Bank BJB dan Pemerintah Daerah Kota Depok. Komunikasi data menggunakan ISO 8583 sebagai jembatan antara *server* melalui masing-masing *switch* diterapkan pada Sistem Online Pelayanan Pembayaran Pajak Bumi dan Bangunan (SOPP-PBB). Pada penelitian ini, tidak dijelaskan mengenai keterlibatan CAA pada komunikasi antara Biller dan CA. Selain itu, pada penelitian ini hanya menyebutkan adanya *inquiry*, *payment*, dan *reversal* tanpa menjelaskan proses bisnisnya.

Penelitian lainnya (Yunandar, 2016) juga mengimplementasikan ISO 8583 untuk pembayaran tagihan mahasiswa Bina Sarana Informatika (BSI) melalui *switching* PT Finnet Indonesia berbasis Advanced Interactive Executive (AIX). AIX merupakan sebuah sistem *middleware* yang dirancang oleh IBM. Pada penelitian ini, PT Finnet Indonesia bertindak sebagai CAA. Informasi transaksi pada penelitian dilakukan dengan format SMS pada telepon seluler berbasis Android. Penelitian ini menggunakan metode *waterfall* dengan metode pengujian integrasi. Selain itu, ISO 8583 juga diterapkan pada proses *inquiry*, *payment*, dan *reversal*. Hasil penelitian yang dilakukan, yaitu ISO 8583 dapat diterapkan dalam suatu *middleware* berbasis AIX untuk menghubungkan *channel*. Namun, sama seperti penelitian yang disebutkan sebelumnya (Titi & Sari, 2017), pada penelitian ini tidak dijelaskan secara detail faktor dan kondisi yang membuat transaksi gagal/*timeout*, serta solusi yang dilakukan.

Penelitian lainnya (Nur Utomo, 2017) merancang sebuah format ISO 8583 untuk sistem pembayaran Brilink yang dikembangkan oleh Bank Rakyat Indonesia (BRI). Penelitian ini melibatkan CAA sebagai pihak ketiga yang menjembatani BRI dan Biller terkait. Pada penelitian ini dijelaskan secara rinci spesifikasi ISO 8583 mulai dari *inquiry*, *payment*, dan *reversal*. Namun, pada penelitian yang akan dilakukan pada laporan akhir ini, isi data dari spesifikasi tidak dijelaskan sedetail itu dan hanya sesuai dengan standar dari ISO. Hal ini dikarenakan sifatnya yang *confidential* dan *private*, serta menyangkut keterlibatan Bank dan CAA terkait. Secara jenis transaksi yang dijelaskan (*inquiry*, *payment*, dan *reversal*) hampir

sama dengan yang dijelaskan pada laporan akhir ini. Namun, pada penelitian (Nur Utomo, 2017) tidak dijelaskan informasi seputar pengujian. Ringkasan hasil tinjauan pustaka dapat dilihat pada Tabel 2.3.

Tabel 2.3 Hasil tinjauan pustaka

No.	Aspek yang ditinjau	Peneliti dan Tahun		
		Yunandar, 2016	Titi & Sari, 2017	Nur Utomo, 2017
1.	Tema Penelitian	Pengembangan sistem pembayaran tagihan mahasiswa BSI	Pengembangan sistem pembayaran elektronik PBB	Perancangan format ISO 8583 pada sistem pembayaran
2.	Metode	Model proses Waterfall dengan metode pengujian integrasi	Model arsitektur <i>3-tier</i>	
3.	Kanal transaksi	Aplikasi SMS Banking berbasis Android	Pembayaran elektronik secara <i>online</i> (kanal tidak disebutkan secara khusus)	Brilink (kanal tidak disebutkan secara khusus)
4.	CAA	✓		✓
5.	ISO 8583	✓	✓	✓
6.	<i>Inquiry transaction</i>	✓	✓	✓
7.	<i>Payment transaction</i>	✓	✓	✓
8.	Solusi saat transaksi gagal	<i>Reversal</i>	<i>Reversal</i>	<i>Reversal</i>
9.	Pengujian	✓		
10.	Pembeda	<ul style="list-style-type: none"> - Implementasi ISO 8583 dijelaskan dengan rinci - Tidak dijelaskan kondisi penyebab <i>reversal</i> - Proses dilakukan hingga pengujian 	<ul style="list-style-type: none"> - Tidak dijelaskan proses yang terjadi saat transaksi <i>inquiry, payment, dan reversal</i> - Tidak dijelaskan kondisi dan penyebab <i>reversal</i> 	<ul style="list-style-type: none"> - Informasi bit yang bersifat privat dijelaskan secara rinci - Tidak ada informasi pengujian

Keterangan: ✓ = Terdapat adanya keterlibatan aspek yang ditinjau pada penelitian

BAB III

PELAKSANAAN MAGANG

Bab ini membahas proses pelaksanaan magang sebagai Middleware SA di Bank. Selama magang enam bulan, telah dilakukan kontribusi terhadap tujuh proyek. Pada bab ini, akan dibahas empat dari tujuh proyek tersebut. Proyek yang akan dibahas secara detail adalah proyek yang menggunakan Internet Banking sebagai kanal transaksi, di antaranya yaitu proyek Pembayaran Tagihan Perusahaan Gas Negara (PGN), Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB), *Top Up Voucher* pada Fitur Perjalanan melalui *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melalui OTA.

Proses pelaksanaan yang dibahas meliputi manajemen proyek, analisis kebutuhan proyek, dan perancangan sistem. Analisis kebutuhan proyek membahas proses analisis kebutuhan proyek dari keempat proyek, sedangkan untuk perancangan sistem akan dibedakan berdasarkan pihak ketiga yang terlibat, CAA atau OTA. Proyek Pembayaran Tagihan Perusahaan Gas Negara (PGN) dan Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB) melibatkan CAA sebagai jembatan antara Bank dan Biller, *Top Up Voucher* pada Fitur Perjalanan melibatkan *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melibatkan OTA sebagai pihak ketiga.

3.1 Manajemen Proyek

Manajemen proyek pada pengembangan proyek pada perusahaan tempat magang secara umum dilakukan secara *hybrid*, yaitu menggunakan kerangka kerja *agile* dan *waterfall*. Namun, untuk Divisi Middleware, kerangka kerja yang digunakan, yaitu *waterfall*. Kerangka kerja *waterfall* yang digunakan oleh Middleware ini secara umum terdiri atas analisis, perancangan, pengembangan, dan pengujian. Estimasi setiap proses biasanya akan diperinci pada dokumen *feasibility study* yang dimiliki oleh setiap proyek. Middleware SA bertanggung jawab atas proyek pada tahap analisis dan perancangan dengan estimasi pengerjaan setiap proyek 2-3 minggu. Pembagian dan deskripsi setiap *role* dalam Middleware dipaparkan pada Tabel 3.1.

Setiap memulai proyek baru, dilakukan *meeting* dan diskusi melalui Microsoft Teams bersama pemangku kepentingan terkait, yang terdiri dari *system analyst*, *testing engineer*, *developer*, serta pemangku kepentingan dari divisi lain yang bekerja sama dalam proyek

tersebut. Pada *meeting* tersebut, akan dijelaskan latar belakang, lingkup, dan tujuan proyek. Middleware SA akan bertanya secara detail kepada para pemangku kepentingan untuk memastikan tidak adanya miskomunikasi terhadap proyek sehingga sistem yang akan dirancang sesuai dengan yang diharapkan. Setelah semua pemangku kepentingan memahami proyek, Middleware SA berkewajiban untuk melakukan proses analisis dan perancangan sistem yang dituangkan ke dalam dokumen System Design.

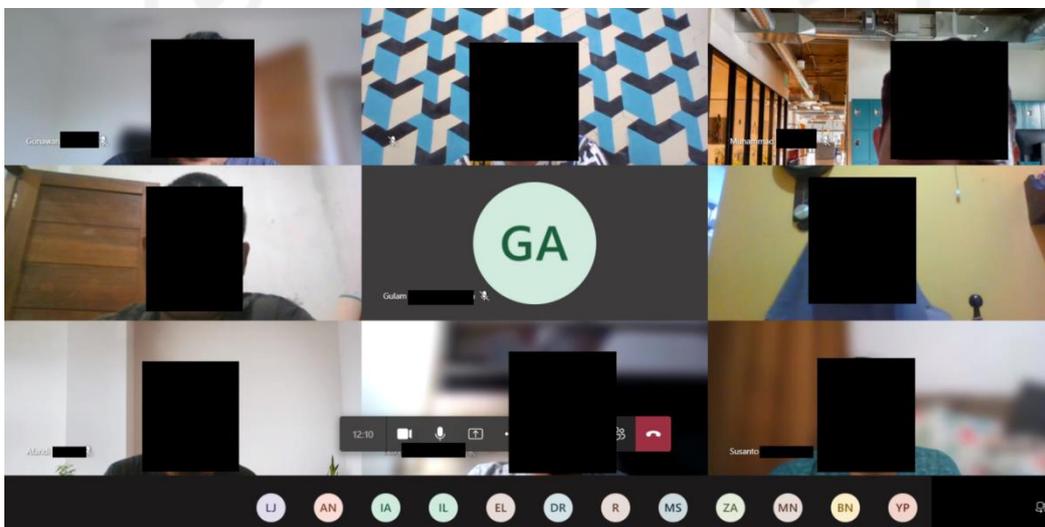
Tabel 3.1 Deskripsi pekerjaan pada Divisi Middleware

No.	Role Pekerjaan	Deskripsi
1.	IT Application Development Head	Kepala divisi penempatan yang bertanggung jawab atas seluruh aktivitas pada divisi.
2.	IT Application Development Lead	Mengarahkan tim dalam perancangan, pengembangan, dan pengujian aplikasi. Selain itu, biasanya juga membantu pemegang dalam melakukan komunikasi ke pihak eksternal.
3.	IT Application System Analyst	Melakukan analisis kebutuhan, perancangan sistem, pendokumentasian, serta jembatan komunikasi antara pengembang dan penguji.
4.	IT Application Developer	Melakukan pengembangan dan pengintegrasian sistem.
5.	IT Application Testing Engineer	Melakukan pengujian sistem untuk memastikan proyek berjalan dengan baik.

Setelah rapat perdana proyek, biasanya akan dilakukan rapat rutin untuk membahas proyek setiap seminggu sekali (*weekly meeting*). Rapat mingguan ini dilakukan bersama pengembang dan penguji yang ada di proyek. Pada rapat ini, pengembang akan memberikan reviu dan umpan baliknya terhadap proses pengerjaan dokumen ISO Message Format dan System Design. Jika dokumen System Design sudah disetujui oleh pengembang, selanjutnya dilakukan *circulate approval* terhadap dokumen System Design ke pemangku kepentingan terkait, mulai dari internal Middleware (sebagai IT Developer) dan eksternal (IT Infrastructure dan IT Project Manager). Dokumen ISO Message Format tidak memerlukan *circulate approval*.

Setelah proses *circulate approval* dokumen System Design selesai, proyek masuk ke proses pengembangan. Saat pengembangan mulai dilakukan, biasanya Middleware SA mulai melakukan perancangan dokumen skenario pengujian. Jika telah selesai, skenario pengujian akan diberikan ke penguji seraya menunggu proyek bisa dilakukan pengujian. Tanggung jawab seorang pemegang sebagai Middleware SA hanya sampai proses tersebut. Ketika dokumen ISO Message Format, System Design, dan skenario pengujian telah selesai, mentor (Senior Middleware SA) akan memberikan proyek baru.

Selain rapat mingguan bersama tim yang terkait dalam proyek yang sedang berjalan, ada juga rapat harian (*daily meeting*) bersama tim Middleware. Namun, pemegang tidak diwajibkan hadir pada rapat ini sehingga hanya diundang melalui Teams beberapa kali pertemuan. Tangkapan layar rapat bersama tim Middleware dapat dilihat pada Gambar 3.1. Adapun selama kegiatan juga difasilitasi *checkpoint meeting* yang dilaksanakan setiap Kamis atau Jumat. *Checkpoint meeting* merupakan fasilitas yang diberikan dari kegiatan magang untuk berdiskusi dengan mentor. Mentor juga akan memberikan kritik dan saran tentang performa kita pada minggu tersebut. Pada kegiatan ini, biasanya dihadiri oleh IT Application Head dan Senior Middleware SA sebagai mentor.



Gambar 3.1 Kegiatan rapat bersama tim Middleware

Sistem manajemen proyek pada Middleware masih dilakukan secara manual menggunakan email, misalnya untuk proses reviu dokumen. Layanan email yang digunakan adalah Microsoft Outlook, tetapi pada pemegang diberikan Blackberry Access yang hanya bisa dibaca tanpa disunting dan disimpan di penyimpanan internal (*read-only*). Hal ini dikarenakan semua dokumen bersifat *confidential*. Untuk penyimpanan bersama, dilakukan menggunakan Microsoft OneDrive. Namun, pemegang juga tidak diberi akses ke Microsoft OneDrive.

3.2 Analisis Kebutuhan Proyek

Analisis kebutuhan sistem dilakukan berdasarkan analisis dari dokumen *feasibility study*. *Feasibility study* berisi kelayakan sistem bank untuk melakukan perencanaan proyek. Lingkup yang dibahas pada *feasibility study* berisi kelayakan keseluruhan sistem yang akan

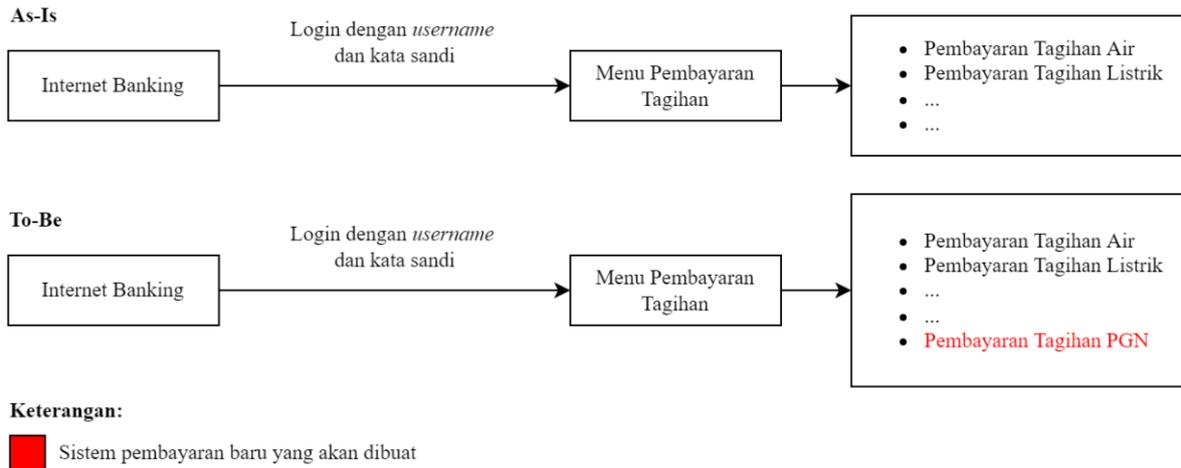
dibangun. Melalui pendekatan *model-driven analysis*, hasil analisis akan dipetakan dan kemudian menjadi panduan untuk membangun sebuah perancangan sistem proyek (*system design*).

3.2.1 Proyek Pembayaran Tagihan Perusahaan Gas Negara (PGN)

Pada proyek ini, dilakukan penambahan fitur pembayaran tagihan PGN pada Internet Banking. Menu pembayaran tagihan yang dimiliki bank belum memiliki fitur untuk melakukan pembayaran tagihan PGN sehingga dilakukan menu pembayaran baru. Sebelum melakukan perancangan sistem integrasi pada *middleware* dilakukan penggambaran cerita pengguna seperti pada Tabel 3.2. Pengembangan sistem pembayaran tagihan PGN sebenarnya merupakan pengembangan dari menu pembayaran tagihan yang telah dimiliki sistem Bank. Jika digambarkan, aliran bisnis yang akan terjadi seperti pada Gambar 3.2.

Tabel 3.2 Cerita pengguna fitur pembayaran tagihan PGN

<i>As a/an...</i>	<i>I want...</i>	<i>So that...</i>
Nasabah	Internet Banking memiliki fitur pembayaran tagihan PGN pada menu pembayaran tagihan.	Nasabah dapat membayar tagihan PGN setelah login ke Internet Banking dan masuk ke menu pembayaran tagihan.
Nasabah	Memilih <i>source account</i> yang diinginkan.	Nasabah dapat melakukan pembayaran sesuai dengan <i>source account</i> yang diinginkan.
CA	Melakukan validasi ID Pelanggan yang dimasukkan oleh nasabah.	Nasabah dapat melakukan transaksi dengan ID Pelanggan yang benar.
CA – Internet Banking	Menampilkan <i>popup error</i> apabila ID Pelanggan yang dimasukkan salah.	Nasabah memperhatikan <i>error</i> yang dapat muncul jika ID Pelanggan salah dan memasukkan ID Pelanggan yang benar.
CA – Internet Banking	Mengirimkan notifikasi email atau SMS setelah transaksi selesai dilakukan.	Nasabah dapat mengetahui status dan detail transaksi.



Gambar 3.2 Aliran bisnis sistem pembayaran tagihan PGN

3.2.2 Proyek Pembayaran Tagihan Pajak Bumi dan Bangunan

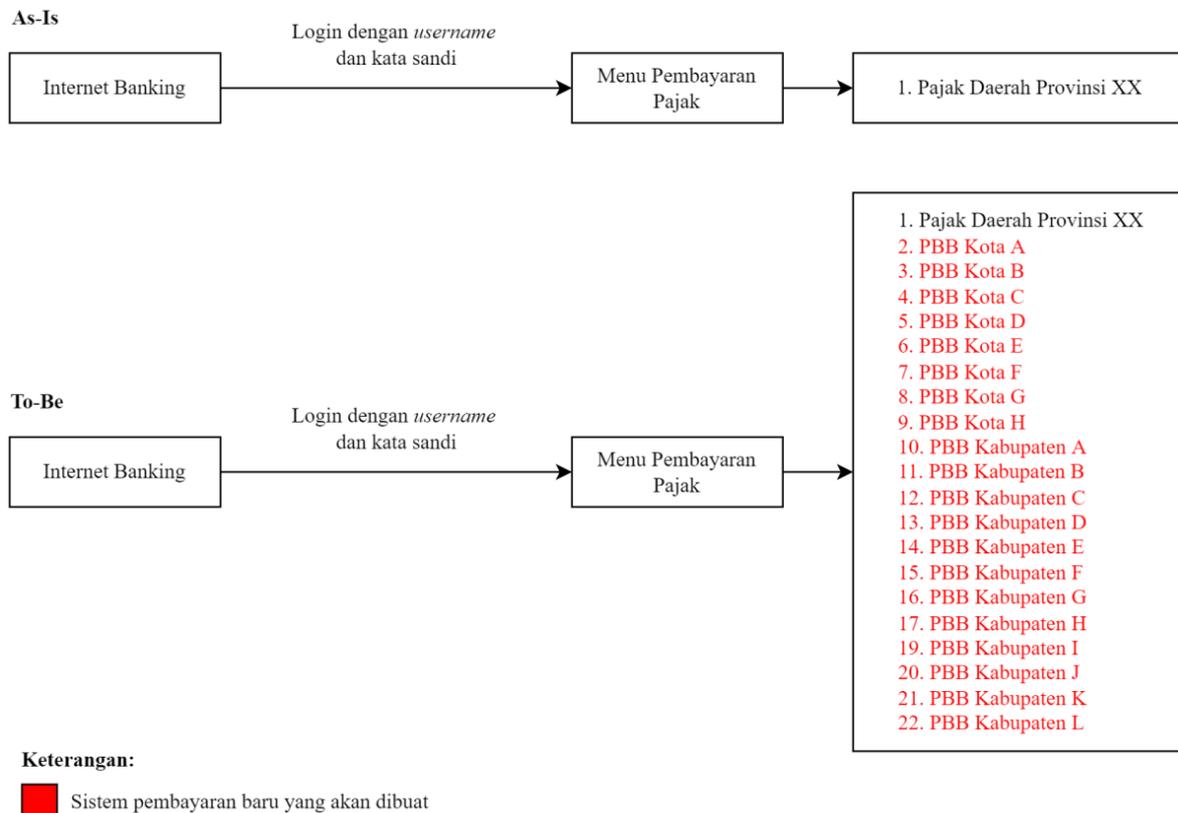
Proses analisis pada proyek pembayaran tagihan PBB dilakukan dengan menganalisis *feasibility study* proyek. Fitur pembayaran pajak pada Internet Banking yang dimiliki Bank belum mendukung pembayaran PBB sehingga akan dilakukan pengembangan fitur yang telah ada, yakni menambahkan layanan pembayaran PBB pada menu pembayaran pajak di Internet Banking. Analisis kebutuhan proyek dapat dapat dijabarkan dalam cerita pengguna secara umum seperti pada Tabel 3.3.

Tabel 3.3 Cerita pengguna pengembangan fitur pembayaran tagihan PBB

<i>As a/an...</i>	<i>I want...</i>	<i>So that...</i>
Nasabah	Internet Banking memiliki fitur pembayaran tagihan PBB pada menu pembayaran tagihan pajak.	Nasabah dapat membayar tagihan PBB setelah login ke Internet Banking dan masuk ke menu pembayaran tagihan pajak.
CA	Melakukan validasi Nomor Objek Pajak (NOP) yang dimasukkan oleh nasabah.	Nasabah dapat melakukan transaksi dengan NOP yang benar.
CA – Internet Banking	Menampilkan <i>popup error</i> apabila NOP yang dimasukkan salah.	Nasabah memperhatikan <i>error</i> yang dapat muncul jika NOP salah dan memasukkan NOP yang benar.
	Mengirimkan notifikasi email atau SMS setelah transaksi selesai dilakukan.	Nasabah dapat mengetahui status dan detail transaksi.

. Pembayaran pajak pada Internet Banking yang dimiliki Bank sebelumnya telah memiliki fitur untuk pembayaran pajak daerah. Fitur ini kemudian dikembangkan dengan menambahkan fitur pembayaran PBB yang terdiri dari 20 fitur yang mewakili 20 daerah

kabupaten/kota yang ada di Indonesia. Pengembangan fitur tagihan PBB ini difokuskan pada Internet Banking, Middleware System, dan Core Banking System. Aliran bisnis secara sederhana dapat dilihat pada Gambar 3.3.



Gambar 3.3 Aliran bisnis sistem pembayaran tagihan PBB

Analisis *feasibility study* selanjutnya difokuskan pada pengembangan sistem *middleware*. Pada perancangan sistem integrasi *middleware* untuk pembayaran tagihan PBB ini, kanal yang akan dilakukan pengembangan adalah Internet Banking. *Middleware* akan menambahkan parameter baru dalam sistemnya sehingga dapat mendukung proses pembayaran tersebut. Transaksi pembayaran PBB akan menggunakan sistem *switching* CAA secara H2H sebagai *3rd Party* yang terkoneksi dengan Bank dan Biller.

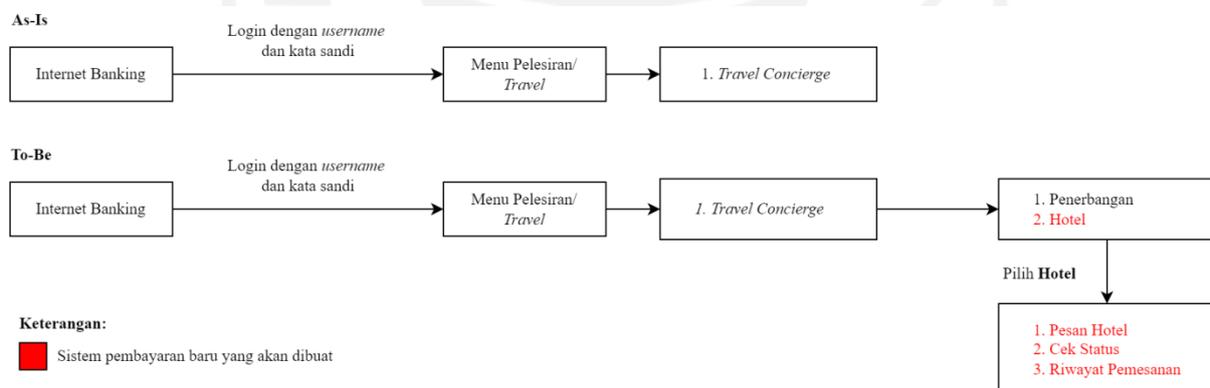
3.2.3 Proyek Pemesanan Hotel pada Fitur Perjalanan melalui OTA

Fitur Perjalanan pada Internet Banking menyediakan layanan pemesanan dan pembayaran tiket pesawat. Untuk menambah pengalaman pengguna yang lebih baik, dilakukan penambahan menu pemesanan hotel pada fitur tersebut. Pengembangan layanan pemesanan hotel ini terintegrasi dengan salah satu *Online Travel Agent* (OTA) yang ada di Indonesia. Cerita pengguna pada proyek ini dapat dilihat pada Tabel 3.4.

Tidak seperti proyek lainnya yang melibatkan CAA, pada proyek ini tidak melibatkan CAA. Middleware nantinya akan berkomunikasi langsung dengan OTA sebagai *3rd Party* tanpa bantuan CAA. Jika digambarkan, aliran bisnis pada proyek pemesanan hotel pada fitur perjalanan digambarkan pada Gambar 3.4.

Tabel 3.4 Cerita pengguna layanan pemesanan hotel pada fitur perjalanan

<i>As a/an...</i>	<i>I want...</i>	<i>So that...</i>
Nasabah	Internet Banking memiliki fitur pemesanan hotel pada fitur perjalanan.	Nasabah dapat memilih hotel sesuai dengan waktu, ruangan, dan area yang diinginkan, serta dapat melakukan pengecekan status dan riwayat pemesanan.
CA – Internet Banking	Menampilkan data dan status hotel yang terintegrasi dengan OTA.	Nasabah mendapat data dan status <i>hotel</i> yang terbaru/ <i>updated</i> sebelum melakukan proses transaksi.
	Mengirimkan notifikasi email atau SMS setelah transaksi selesai dilakukan.	Nasabah dapat mengetahui status dan detail transaksi.



Gambar 3.4 Aliran bisnis pemesanan hotel pada fitur perjalanan

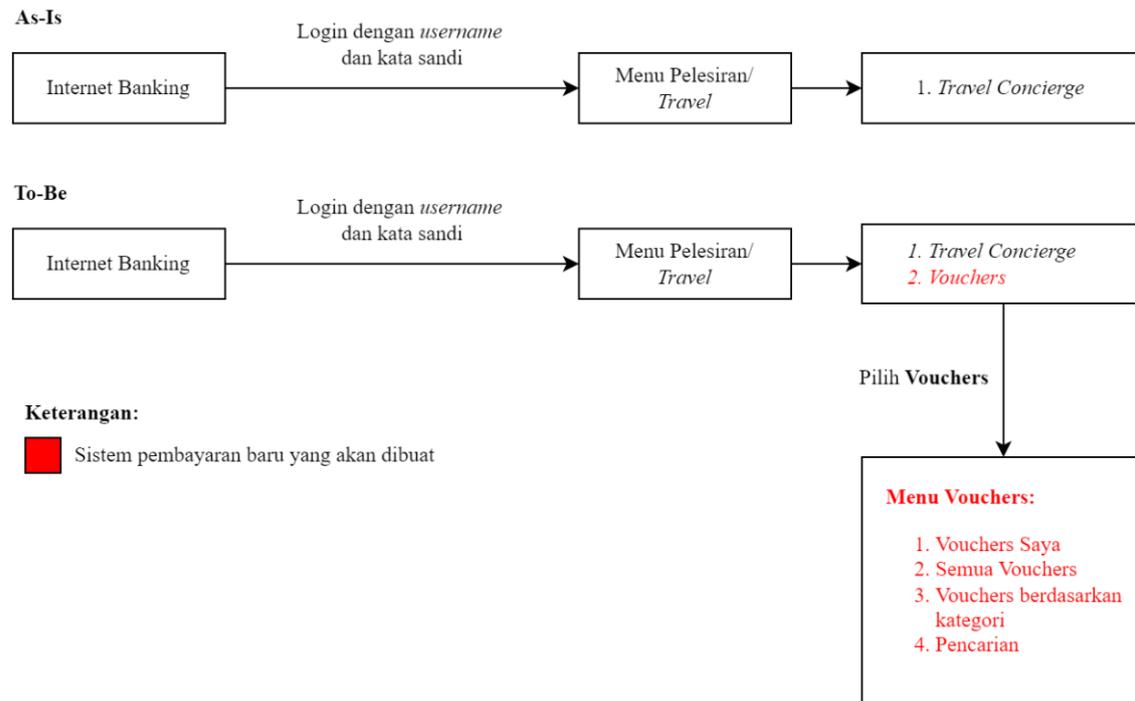
3.2.4 Proyek *Top Up Voucher* pada Fitur Perjalanan melalui *Voucher Agent*

Pada proyek ini, terdapat penambahan fitur pada Internet Banking yang dimiliki Bank untuk memfasilitasi nasabah dalam melakukan transaksi menggunakan voucher. Nasabah dapat melakukan *top up* voucher, melakukan instalasi *voucher* setelah proses *top up*, melihat *voucher*, menghapus *voucher*, serta melakukan sinkronisasi antara *voucher* yang dimiliki pada Internet Banking dan Mobile Banking. Secara umum, cerita pengguna pada proyek *top up voucher* digambarkan pada Tabel 3.5.

Penambahan fitur *top up voucher* melibatkan beberapa sistem yang ada di Bank. Namun, yang akan dibahas secara detail hanya sistem *middleware* sebagai sistem integrasi. *Middleware* pada proyek ini juga akan melakukan integrasi ke *Voucher Agent* sebagai pihak ketiga yang menyediakan *voucher*. Pengembangan akan difokuskan pada Internet Banking, Middleware System, dan Core Banking System. Fitur *top up voucher* sebenarnya merupakan pengembangan dari fitur perjalanan yang sebelumnya telah ada pada Internet Banking yang dimiliki Bank. Terdapat penambahan menu pada fitur perjalanan yang apabila digambarkan aliran bisnisnya seperti pada Gambar 3.5.

Tabel 3.5 Cerita pengguna *top up voucher* pada fitur perjalanan

<i>As a/an...</i>	<i>I want...</i>	<i>So that...</i>
Nasabah	Internet Banking memiliki menu baru untuk transaksi voucher yang terhubung ke <i>Voucher Agent</i> .	Nasabah dapat melakukan: <ul style="list-style-type: none"> - Pembelian dan <i>top up voucher</i> - Lihat voucher dan riwayat transaksi voucher - Hapus voucher
CA – Internet Banking	Internet Banking memiliki data <i>voucher</i> yang sinkron dengan yang dimiliki <i>Voucher Agent</i> (data selalu terbaru/ <i>updated</i>).	Nasabah mendapat data stok <i>voucher</i> yang terbaru/ <i>updated</i> sebelum melakukan proses transaksi.
	Internet Banking dapat memberikan validasi apabila voucher telah habis.	Nasabah mengetahui bahwa terjadi kegagalan transaksi karena stok voucher telah habis.
	Internet Banking memiliki fitur pencarian voucher	Nasabah dapat melakukan pencarian <i>voucher</i> berdasarkan nama atau kategori



Gambar 3.5 Aliran bisnis sistem *top up* voucher pada fitur perjalanan

3.3 Perancangan Sistem

Setelah proses analisis selesai, selanjutnya adalah tahap perancangan. Perancangan sistem yang dilakukan meliputi perancangan *inquiry transaction*, *payment transaction*, *reversal transaction*, dan skenario pengujian. Secara umum, perancangan sistem yang dilakukan untuk proyek-proyek yang menggunakan Internet Banking secara alur proses integrasi *inquiry* dan *payment* adalah sama. Perbedaannya terletak pada persyaratan yang dibutuhkan antara CA dan CAA/OTA/*Voucher Agent*, data yang dikirim dan diterima, serta skenario *reversal* dan pengujian.

Proyek yang membutuhkan perancangan transaksi *inquiry* hanya Pembayaran Tagihan PGN dan Pembayaran Tagihan PBB. Hal ini dikarenakan proyek ini membutuhkan informasi tagihan dari CAA. Selain itu, pada transaksi *payment* dan *reversal* nantinya akan dibedakan berdasarkan pihak ketiga yang terlibat, yaitu CAA, OTA atau *Voucher Agent*. Proyek Pembayaran Tagihan Perusahaan Gas Negara (PGN) dan Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB) melibatkan CAA sebagai jembatan antara Bank dan Biller, Pemesanan Hotel pada Fitur Perjalanan melibatkan OTA, dan *Top Up Voucher* pada Fitur Perjalanan melibatkan *Voucher Agent*. Rincian lebih jelas dapat dilihat pada Tabel 3.6.

Tabel 3.6 Rincian perancangan setiap proyek melalui Internet Banking

Proyek	Rincian Perancangan				3 rd Party		
	<i>Inquiry</i>	<i>Payment</i>	<i>Reversal</i>	<i>Testing</i>	CAA	OTA	<i>Voucher Agent</i>
Pembayaran Tagihan Perusahaan Gas Negara (PGN)	✓	✓	✓	✓	✓		
Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB)	✓	✓	✓	✓	✓		
Top Up Voucher pada Fitur Perjalanan		✓	✓	✓			✓
Pemesanan Hotel pada Fitur Perjalanan		✓	✓	✓		✓	

Keterangan: ✓ = Ya

3.3.1 Perancangan Transaksi *Inquiry*

Tahap pertama perancangan adalah melakukan perancangan pada transaksi *inquiry*. Transaksi *inquiry* hanya dirancang untuk transaksi pembayaran tagihan PGN dan PBB. Untuk proyek *Top Up Voucher* pada Fitur Perjalanan melalui *Voucher Agent* dan Pemesanan Hotel pada Fitur Perjalanan melalui OTA, tidak perlu dilakukan perancangan *inquiry*. Hal ini dikarenakan kedua transaksi ini bukan merupakan transaksi pembayaran tagihan (*bill payment*), melainkan pembelian/belanja (*purchase*). Pada transaksi *inquiry* pembayaran tagihan PGN dan PBB, Bank akan melakukan pengecekan tagihan atas transaksi yang akan dilakukan nasabah ke CAA. Tagihan ini sebelumnya telah ada dan harus dibayar oleh nasabah. Pengecekan tagihan ini untuk menjaga kesesuaian data antara Bank, CAA, dan Biller. Pada *Top Up Voucher* pada Fitur Perjalanan dan Pemesanan Hotel pada Fitur Perjalanan melalui OTA, tidak ada tagihan rutin yang harus dibayar nasabah.

Pada prosesnya, Bank akan meminta informasi tagihan (*bill inquiry*) ke CAA dengan mengirimkan pesan dengan tipe 0200 (Financial Transaction Request) ke CAA. CAA harus mengirimkan kembali pesan dengan tipe 0210 (Financial Transaction Response) ke Bank sebagai respon dari *request inquiry*. Pada umumnya, pertukaran pesan pada transaksi *inquiry* telah diatur oleh ISO 8583 seperti pada Tabel 3.7. Nantinya, yang menjadi pembeda adalah Processing Code pada bit #3 serta Additional Data pada bit #48 dan bit #62 yang berisi informasi *inquiry* yang bersifat privat.

Processing Code pada bit #3 berisi kode proses pertukaran pesan. Setiap transaksi pada Bank memiliki kode proses yang berbeda-beda sebagai *identifier* setiap transaksi. Kode proses ini biasanya hanya dirancang untuk transaksi *inquiry* dan *payment*.

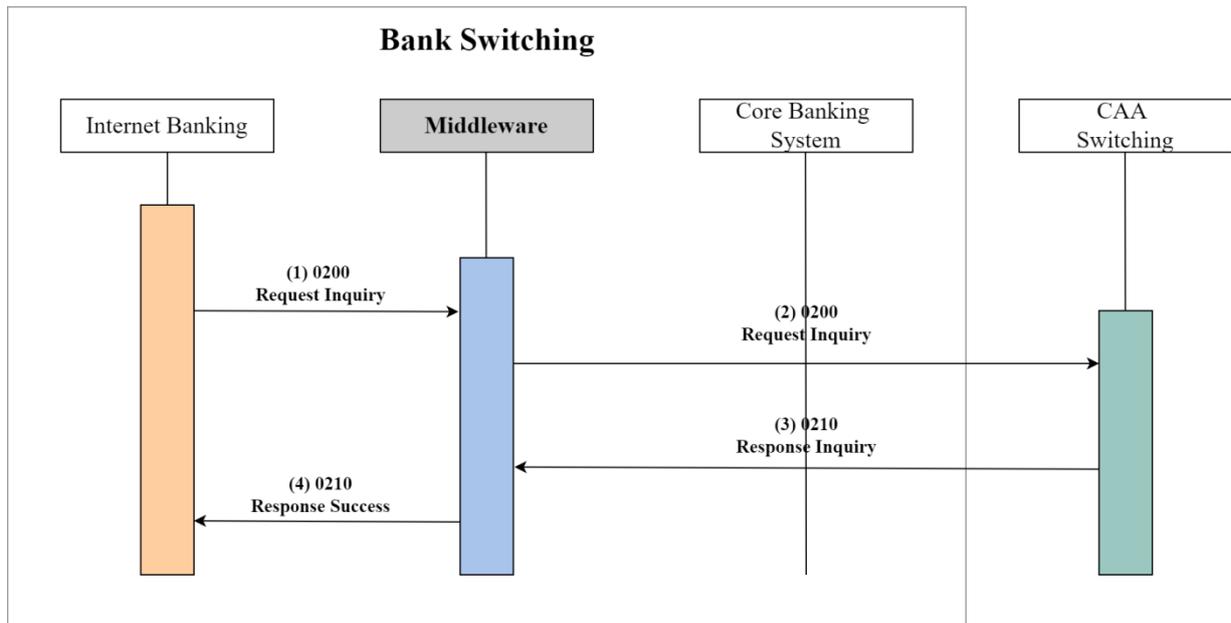
Tabel 3.7 Informasi data yang dipertukarkan pada transaksi *inquiry*

Bit#	Elemen Data	Format	Atribut	0200	0210
	Message Type			M	M
	Primary bit map		b-64	M	M
2	Primary Account Number	LLVAR	n...16	M	M
3	Processing Code		n-6	M	M
4	Transaction Amount		n-12	M	M
7	Transmission Date and Time	MMDDhhmmss	n-10	M	M
11	System Trace Audit Number		n-6	M	M
13	Local Transaction Date	MMDD	n-4	M	M
15	Settlement Date	MMDD	n-4	M	M
18	Merchant Type		n-4	M	M
32	Acquiring Institution ID Code	LLVAR	n-4	M	M
37	Retrieval Reference Number		an-12	M	M
39	Response Code		n-2	M	M
41	Card acceptor terminal ID		an-16	M	M
48	Additional Data	LLLVAR	an...999	M	M
49	Transaction Currency Code		n-3	M	M
59	Channel ID		an-3	M	M
62	Additional Data	LLLVAR	an...999	M	M
63	Institution Code	LLLVAR	n-5	M	M
102	Source Account Number	LLLVAR	n-19	M	M

Keterangan: Informasi format elemen data dapat dilihat pada Tabel 2.1

Additional Data pada bit #48 dan bit #62 memuat informasi tagihan yang dikirimkan oleh CAA sebagai respon dari *request inquiry*. Informasi tagihan yang dikirimkan untuk setiap proyek tentunya berbeda, tergantung data yang diperlukan untuk melakukan proses transaksi. Oleh karena sifatnya yang *private*, informasi data yang dipertukarkan pada elemen data bit #3, bit #48, dan bit #62 tidak bisa ditampilkan pada laporan ini.

Format data dalam Additional Data harus sesuai dengan kesepakatan antara Bank, CAA, dan Biller, sehingga biasanya dalam pertukaran data antara CAA dan Biller (di luar Bank), elemen data #3, #48, dan #62 juga digunakan untuk menyimpan informasi tagihan. *Field* ini nantinya juga akan disimpan dalam basis data CAA dan dapat diambil untuk keperluan *reporting* transaksi antara Bank dan Biller. Pada perancangan integrasi Middleware, informasi tagihan pada bit #3, bit #48, dan bit #62 perlu dirinci dalam dokumen ISO Message Format sehingga dapat menjadi panduan *developer* dalam pengembangan.



Gambar 3.6 Alur proses transaksi *inquiry*

Alur proses *inquiry* dapat dilihat pada Gambar 3.6. Proses transaksi *inquiry* akan dimulai saat nasabah melakukan proses transaksi melalui Internet Banking. Sebelum melakukan pembayaran, Internet Banking akan mengirimkan *request inquiry* dengan kode 0200 ke Middleware untuk mendapatkan informasi tagihan. Middleware akan meneruskan *request* tersebut ke CAA. Kemudian, CAA akan mengirimkan *response* berupa informasi tagihan yang harus dibayar oleh nasabah dengan kode 0210 ke Middleware. Middleware meneruskan *response* tersebut ke Internet Banking dan akan ditampilkan ke layar nasabah.

3.3.2 Perancangan Transaksi *Payment*

Perancangan sistem integrasi *middleware* untuk transaksi pembayaran melalui Internet Banking melibatkan *core banking system* yang ada pada Bank. Core Banking berperan dalam pencatatan transaksi nasabah mulai dari nasabah membuka rekening hingga menutupnya. Pada proses ini, *core banking* akan berperan dalam proses pengkreditan dan pendebitan dana nasabah. Pada pembayaran tagihan PBB dan PGN, setelah memperoleh informasi tagihan melalui proses *inquiry*, selanjutnya masuk ke transaksi pembayaran. Namun, untuk *top up voucher* dan pemesanan hotel pada fitur perjalanan, langsung masuk ke proses transaksi pembayaran. Pertukaran pesan dalam format ISO 8583 untuk transaksi *payment* secara umum digambarkan pada Tabel 3.8.

Seperti halnya proses *inquiry*, pada proses *payment* juga akan dilakukan pertukaran Additional Data bit #48 dan bit #62 yang berisi informasi tagihan/pembayaran dan bersifat *private*. Alur proses *payment* dijelaskan oleh Gambar 3.7. Internet Banking pada prosesnya akan melakukan *request payment* ke Middleware. Middleware akan meneruskan *request* tersebut dengan melakukan *request posting* ke Core Banking untuk melakukan pendebitan dana nasabah. Pendebitan dana nasabah oleh Core Banking ini bertujuan untuk mengurangi saldo nasabah yang digunakan untuk pembayaran. Selain pendebitan dana nasabah, Core Banking juga akan melakukan pencatatan transaksi nasabah. Setelah transaksi yang dilakukan sukses, Core Banking akan mengirimkan *response posting success* dengan kode respon 0210 ke Middleware.

Tabel 3.8 Informasi data yang dipertukarkan pada transaksi *payment*

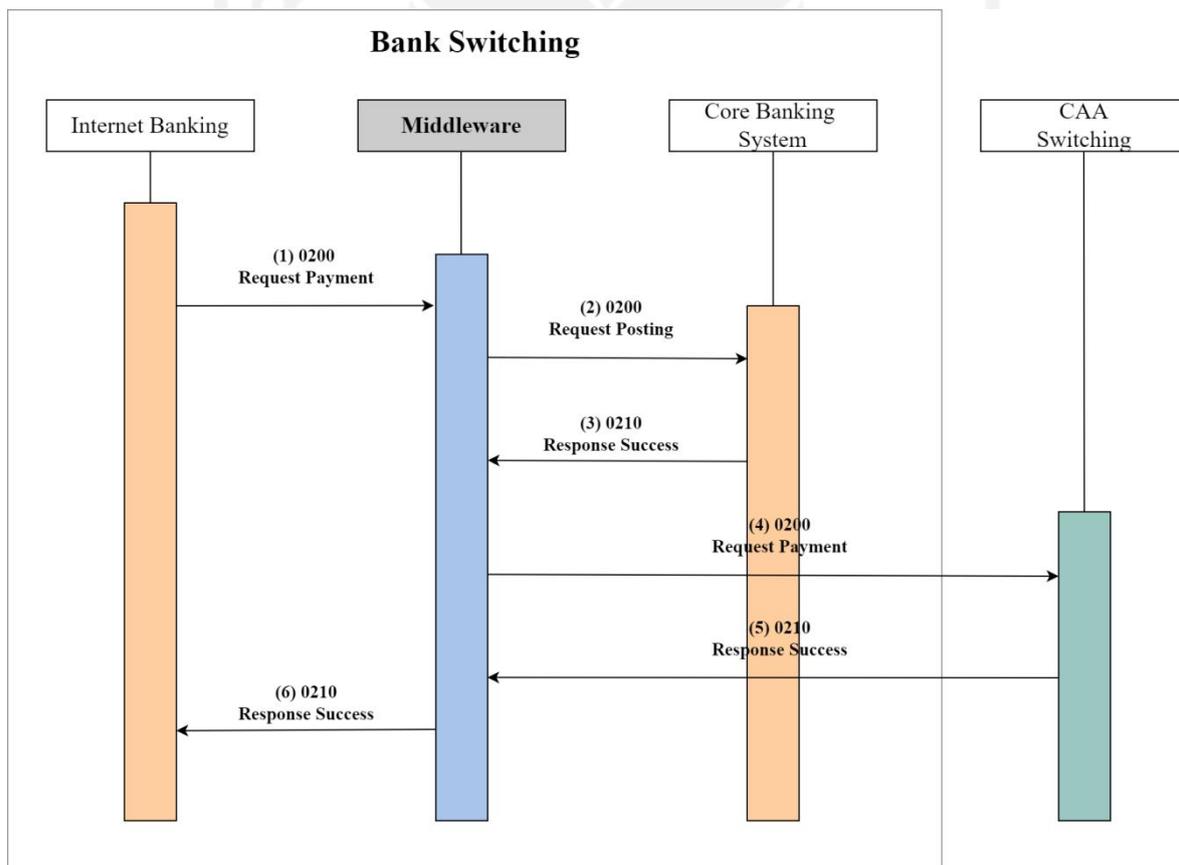
Bit#	Elemen Data	Format	Atribut	0200	0210
	Message Type			M	M
	Primary bit map		b-64	M	M
2	Primary Account Number	LLVAR	n...16	M	M
3	Processing Code		n-6	M	M
4	Transaction Amount		n-12	M	M
7	Transmission Date and Time	MMDDhhmmss	n-10	M	M
11	System Trace Audit Number		n-6	M	M
13	Local Transaction Date	MMDD	n-4	M	M
15	Settlement Date	MMDD	n-4	M	M
18	Merchant Type		n-4	M	
32	Acquiring Institution ID Code	LLVAR	n-4	M	M
37	Retrieval Reference Number		an-12	M	M
39	Response Code		n-2		M
41	Card acceptor terminal ID		an-16	M	M
48	Additional Data	LLLVAR	an...999	M	M
49	Transaction Currency Code		n-3	M	M
59	Channel ID		an-3	M	M
62	Additional Data	LLLVAR	an...999	M	M
63	Institution Code	LLLVAR	n-5	M	M
102	Source Account Number	LLLVAR	n-19	M	

Keterangan: Informasi format elemen data dapat dilihat pada Tabel 2.1

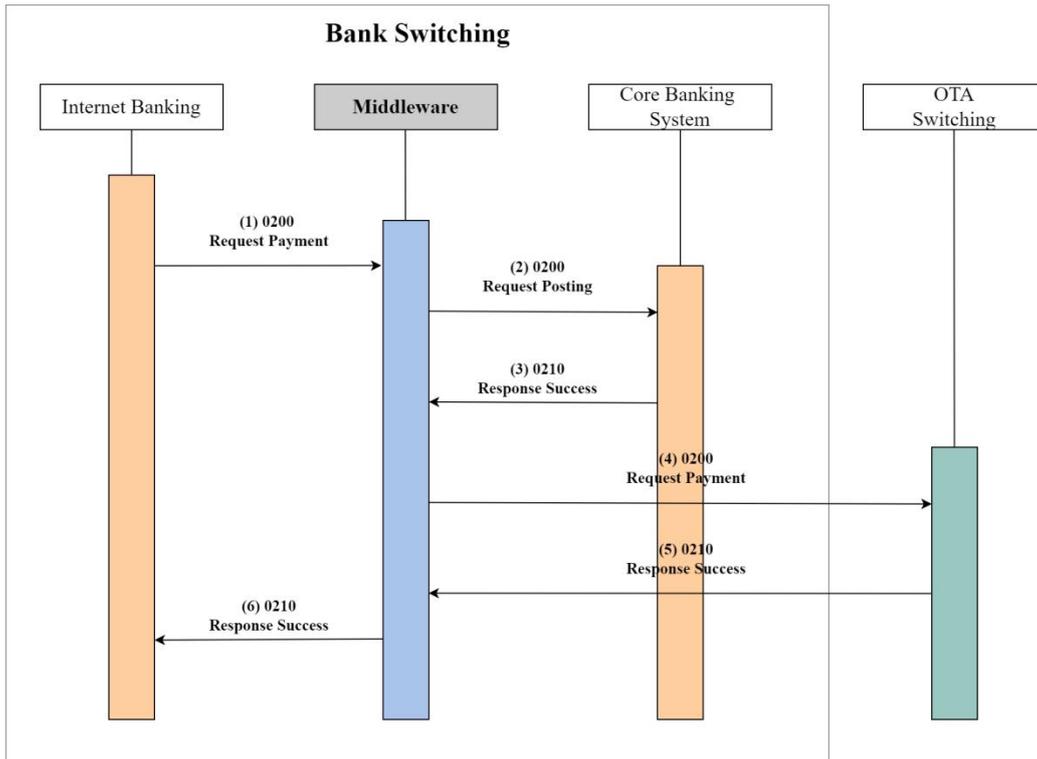
Selanjutnya, setelah menerima respon sukses dari Core Banking, Middleware akan mengirimkan *request payment* dengan kode 0200 ke CAA untuk melakukan kredit dana yang

dibayarkan oleh nasabah. Dana yang dikredit inilah yang nantinya akan diproses oleh CAA sampai ke Biller. Setelah proses kredit berhasil di CAA, CAA akan mengirimkan *response payment success* ke Middleware dengan dengan kode respon 0210. Middleware akan meneruskan respon tersebut ke Internet Banking untuk kemudian ditampilkan ke layar ke nasabah.

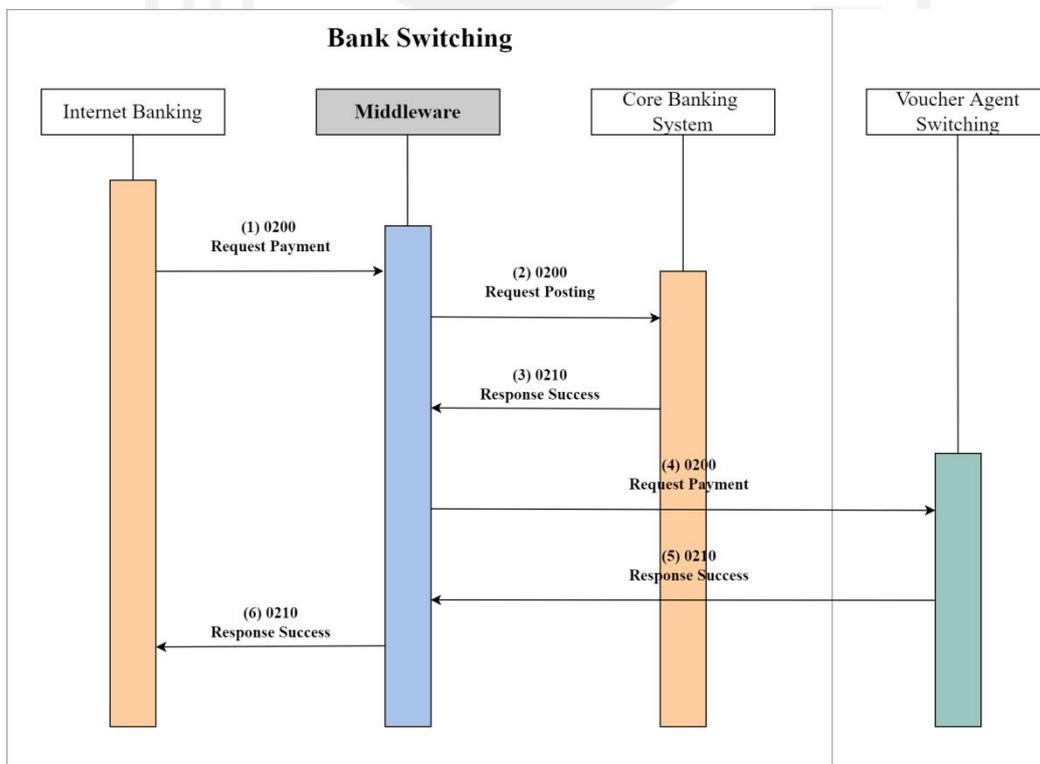
Pada proyek Pemesanan Hotel pada Fitur Perjalanan melalui OTA, ada sedikit perbedaan dalam alur proses yang dirancang. Proyek ini tidak melibatkan CAA sebagai pihak ketiga yang menjembatani Bank dan Biller, melainkan langsung terhubung dengan salah satu OTA yang ada di Indonesia. Begitupula dengan *Top Up Voucher* pada Fitur Perjalanan, pihak ketiga yang terlibat adalah *Voucher Agent*. Secara umum, tidak ada alur transaksi *payment* yang berbeda antara yang melalui CAA, OTA, maupun *Voucher Agent*. Untuk lebih jelas, alur proses integrasi *middleware system* untuk transaksi *payment* Pemesanan Hotel pada Fitur Perjalanan melalui OTA dapat dilihat pada Gambar 3.8 dan pada *Top Up Voucher* pada Fitur Perjalanan dapat dilihat pada Gambar 3.9.



Gambar 3.7 Alur proses transaksi *payment* pembayaran tagihan PBB dan PGN



Gambar 3.8 Alur proses transaksi *payment* untuk pemesanan hotel pada fitur perjalanan melalui OTA



Gambar 3.9 Alur proses transaksi *payment* untuk *top up* voucher melalui *Voucher Agent*

3.3.3 Perancangan Skenario *Reversal*

Pada realisasinya, tidak semua transaksi akan berhasil dilakukan. Transaksi pembayaran bisa saja gagal karena *timeout*, tidak direspon, bahkan ditolak oleh CAA, OTA, *Voucher Agent*, atau Core Banking. Penyebabnya beragam, mulai dari rekening tujuan yang salah, jaringan internet tidak stabil, gangguan pada aplikasi yang digunakan, dan lain sebagainya. Oleh karena itu, perlu sebuah mekanisme apabila transaksi tidak berhasil dilakukan. Transaksi *reversal* merupakan skema pengembalian dana sebagai solusi apabila transaksi tidak berhasil dilakukan. Tujuannya, untuk melindungi dana nasabah dari transaksi yang sebenarnya tidak berhasil sehingga tidak ada saldo nasabah yang berkurang. Untuk transaksi *reversal*, dilakukan perancangan alur *reversal* apabila transaksi gagal/ditolak oleh *3rd Party* atau Core Banking, serta transaksi *timeout*/tidak direspon oleh CAA, OTA, atau Core Banking.

Namun, tidak semua transaksi dapat dilakukan *reversal* apabila tidak berhasil dilakukan, terutama jika gagal/*timeout* terjadi pada *3rd Party* yang terlibat. Hal ini kembali lagi ke persyaratan proyek dan kesepakatan antara CA dan *3rd Party*. CAA mendukung *reversal* hanya jika transaksi *timeout*, sedangkan OTA dan *Voucher Agent* tidak mendukung adanya *reversal*. Rincian lebih lengkap terkait *reversal* dapat dilihat pada Tabel 3.9.

Tabel 3.9 Rincian *reversal* setiap proyek

Proyek	Pembayaran Tagihan PGN	Pembayaran Tagihan PBB	Pemesanan Hotel pada Fitur Perjalanan	Top Up Voucher pada Fitur Perjalanan
<i>3rd Party</i>	CAA		OTA	<i>Voucher Agent</i>
<i>Reversal transaction</i>				
<i>Timeout/No Response 3rd Party</i>	<i>Core Banking</i>	✓	✓	✓
	<i>3rd Party</i>	✓	✓	
<i>Failed/Rejected 3rd Party</i>	<i>Core Banking</i>	✓	✓	✓
	<i>3rd Party</i>			
<i>Timeout/No Response Core Banking</i>	<i>Core Banking</i>	✓	✓	✓
	<i>3rd Party</i>			
<i>Failed/Rejected Core Banking</i>	<i>Core Banking</i>	✓	✓	✓
	<i>3rd Party</i>			

Keterangan: ✓ = Ya

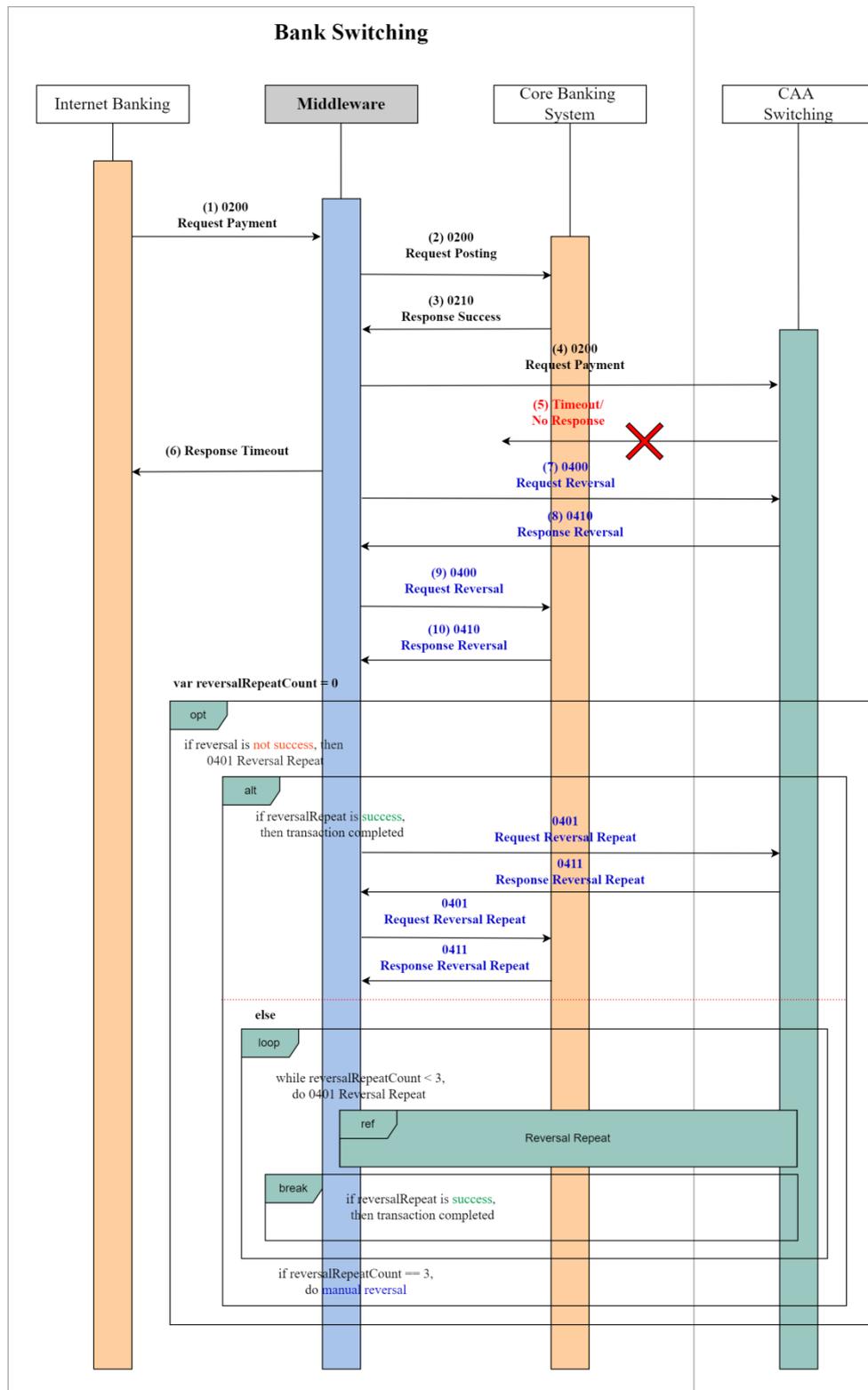
Transaksi Pembayaran melalui Internet Banking *Timeout/No Response* CAA

Transaksi pembayaran melalui Internet Banking yang *timeout/no response* dari CAA hanya terjadi untuk proyek Pembayaran Tagihan PBB dan Pembayaran Tagihan PGN karena proses transaksi pada proyek ini melibatkan CAA. Jika terjadi *timeout/no response* dari CAA, transaksi tidak berhasil dilakukan. Hal ini dikarenakan transaksi melewati tenggang waktu transaksi (*timeout*) atau transaksi tidak mendapatkan respon dari CAA. Sebagai solusi untuk kasus ini, Middleware akan melakukan *request reversal* terhadap pembayaran tagihan yang telah dilakukan sebelumnya ke CAA dan Core Banking untuk mengembalikan dana nasabah.

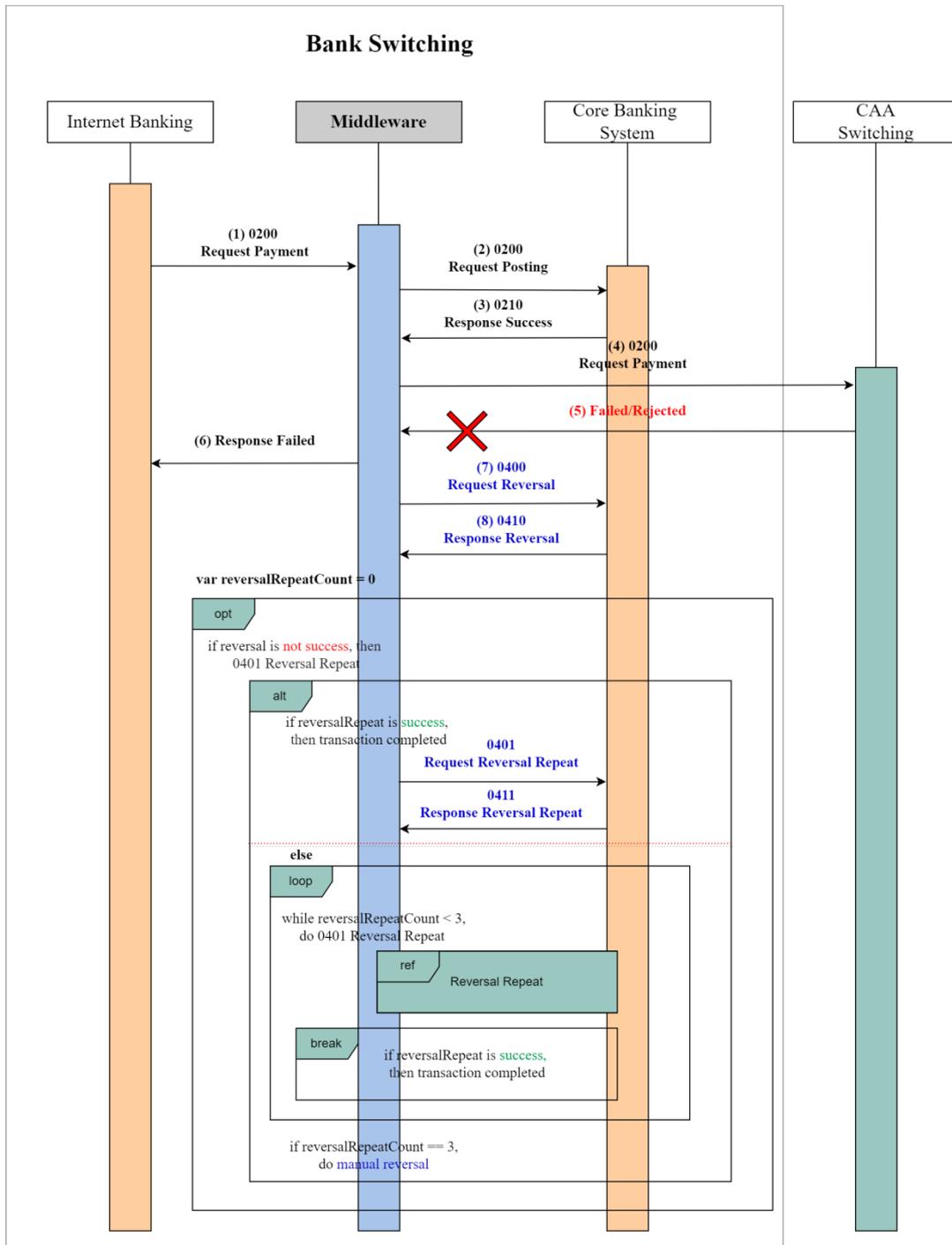
Setelah Middleware tidak mendapatkan respon dari CAA atau telah melewati batas tenggang yang diberikan, Middleware akan mengirim kode respon *timeout* atau *no response* ke Internet Banking untuk diteruskan ke nasabah. Setelah memberikan informasi tersebut, Middleware akan secara otomatis melakukan *request reversal* dengan tipe pesan 0400 ke CAA. *Reversal* ke CAA berfungsi untuk membatalkan pengkreditan dana nasabah di CAA. Oleh karena proses yang dilakukan *timeout/no response*, tidak dapat dipastikan dana berhasil dikredit ke CAA atau tidak sehingga perlu dilakukan *reversal*. Untuk memastikan proses *reversal* berhasil, Middleware membutuhkan pesan dengan tipe 0410 (Reversal Response) dari CAA.

Selanjutnya, setelah menerima respon *reversal* dari CAA, Middleware kembali mengirimkan *request reversal* ke Core Banking dengan tipe 0420. *Reversal* ini diperlukan untuk membatalkan pendebitan kas nasabah di Core Banking sehingga dana nasabah dapat dikembalikan ke rekening. Proses ini juga memerlukan *reversal response* tipe data 0430 dari Core Banking. Transaksi *reversal* dikatakan berhasil apabila dana berhasil dikembalikan sesuai skema mekanisme *reversal*. Proses *reversal* saat transaksi *timeout* dari CAA dapat dilihat pada Gambar 3.10.

Apabila transaksi *reversal* tetap tidak memperoleh respon dari CAA, Middleware akan mengirimkan pesan dengan tipe 0401 (Reversal Repeat Request). Proses *reversal repeat* berhasil apabila Middleware menerima pesan 0411 dari CAA dan Core Banking. Jika dalam tiga kali *request, reversal repeat* tetap tidak berhasil, Middleware akan berhenti mengirimkan *request reversal* dan proses *reversal* dana nasabah akan dilakukan secara manual.



Gambar 3.10 Alur transaksi *timeout/no response* dari CAA



Gambar 3.11 Alur transaksi gagal/ditolak oleh CAA

Transaksi Pembayaran melalui Internet Banking *Rejected/Failed* CAA

Transaksi abnormal selanjutnya, yaitu apabila transaksi gagal atau ditolak oleh CAA. Transaksi ini juga hanya terjadi untuk proyek Pembayaran Tagihan PBB dan Pembayaran Tagihan PGN. Ketika kasus gagal/ditolak oleh CAA terjadi, Middleware tidak perlu mengirimkan *request reversal* ke CAA. Hal ini dikarenakan transaksi sudah jelas ditolak/gagal sehingga tidak ada dana yang dikredit di CAA yang harus di-*reversal*. Namun, karena dana telah didebit oleh Core Banking melalui kas nasabah, dana tersebut perlu dikembalikan ke Core Banking melalui proses *reversal*.

Setelah mendapatkan informasi ditolak atau gagal oleh CAA, Middleware akan meneruskan informasi tersebut ke Internet Banking untuk diteruskan ke nasabah. Selanjutnya, Middleware akan mengirimkan *request reversal* untuk mengembalikan dana nasabah ke Core Banking melalui tipe data 0400 (Reversal Request). Core Banking harus mengirimkan respon 0410 (Reversal Response) ke Middleware sebagai bukti bahwa transaksi *reversal* pada Core Banking telah berhasil dilakukan dan dana telah berhasil dikembalikan ke Core Banking. Untuk alur yang lebih jelas dapat dilihat pada Gambar 3.11.

Namun, seperti yang telah dijelaskan sebelumnya, tidak semua transaksi *reversal* selalu berhasil. Apabila transaksi *reversal* tidak memperoleh respon dari Core Banking, Middleware akan mengirimkan pesan dengan tipe 0401 (Reversal Repeat Request). Proses *reversal repeat* berhasil apabila Middleware menerima pesan 0411 dari Core Banking. Jika dalam tiga kali *request*, *reversal repeat* tetap tidak berhasil, Middleware akan berhenti mengirimkan *request reversal* dan proses *reversal* dana nasabah akan dilakukan secara manual.

Transaksi Pembayaran melalui Internet Banking *Timeout/No Response* OTA atau *Voucher Agent*

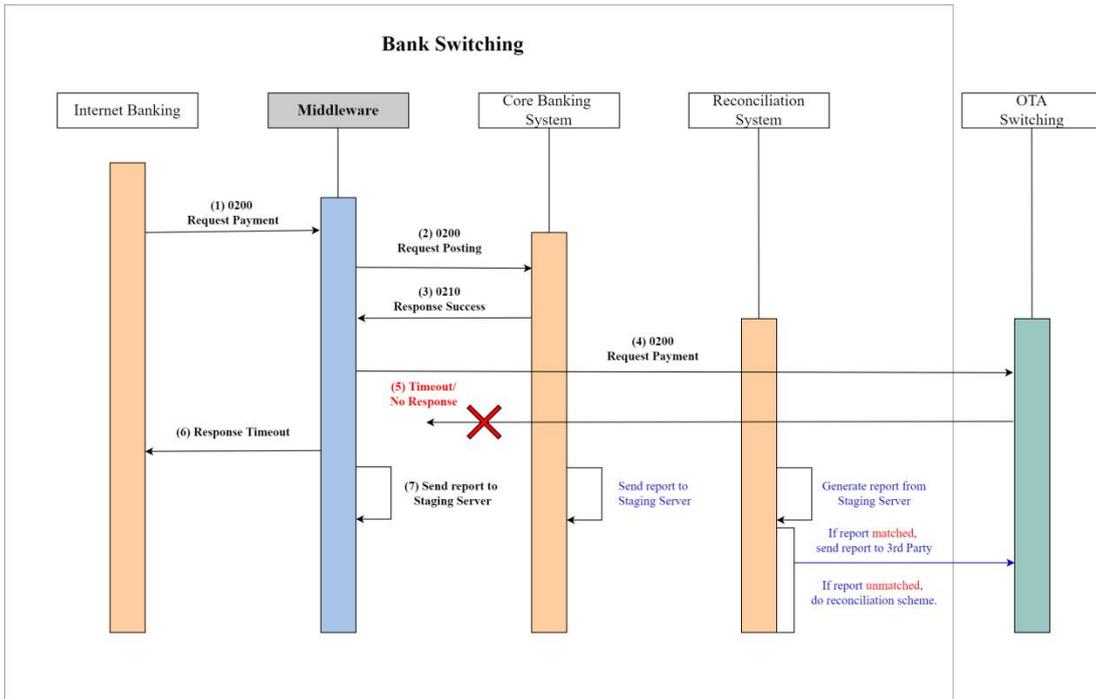
Transaksi pembayaran melalui Internet Banking dapat juga mengalami *timeout/no response* dari OTA seperti pada Gambar 3.12 atau *Voucher Agent* seperti pada Gambar 3.13. Hal ini terjadi pada transaksi Pemesanan Hotel pada Fitur Perjalanan melalui OTA atau *Top Up Voucher* melalui *Voucher Agent*. Transaksi ini melibatkan OTA atau *Voucher Agent* sebagai 3rd Party yang terhubung dengan Internet Banking.

Tidak seperti transaksi yang melibatkan CAA, OTA dan *Voucher Agent* tidak mendukung adanya *reversal* sehingga tidak ada *reversal* yang dilakukan ke OTA atau *Voucher Agent* apabila transaksi *timeout* atau tidak direspon oleh OTA. Hal ini dikarenakan kesepakatan antara Bank dan OTA atau *Voucher Agent* tidak mendukung adanya *reversal*.

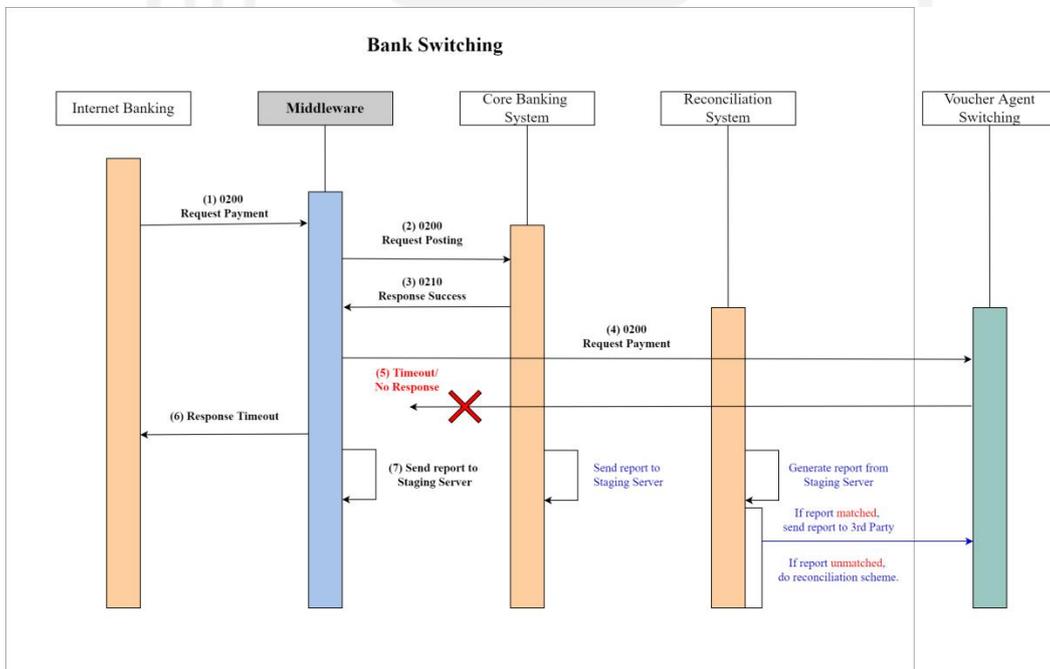
Nantinya, transaksi tidak berhasil karena *timeout* atau tidak direspon statusnya akan menjadi *suspect*, yang artinya belum jelas berhasil atau tidak sehingga harus dilakukan pengecekan lebih lanjut melalui rekonsiliasi. Proses rekonsiliasi akan ditangani oleh sistem *reconciliation* dan bukan wewenang dari sistem *middleware* lagi.

Apabila kasus *timeout* dari OTA terjadi, Middleware juga tidak bisa melakukan *request reversal* ke Core Banking dengan kode 0400. Hal ini dikarenakan status *suspect* yang dimiliki transaksi. Oleh karena tidak adanya *reversal* dari OTA atau *Voucher Agent*, tidak ada kejelasan dana berhasil dikredit di OTA atau *Voucher Agent*, sehingga transaksi yang telah berjalan tidak bisa dibatalkan oleh Core Banking. Dana yang telah didebit dari Core Banking harus tetap ditahan sampai proses rekonsiliasi dilakukan.

Apabila terjadi kasus *timeout/no response* dari OTA atau *Voucher Agent*, Middleware akan meneruskan informasi tersebut ke Internet Banking untuk diinformasikan ke nasabah. Untuk menginformasikan ke Reconciliation System bahwa telah terjadi *timeout*, Middleware dan Core Banking akan mengirimkan *report* ke *staging server* agar dapat di-*generate* oleh Reconciliation System. Reconciliation System nantinya akan membandingkan data antara Bank dan OTA/*Voucher Agent*. Data Bank dalam hal ini akan menjadi *source of truth* untuk proses penyelesaian. Setelah proses pencocokan data selesai, Reconciliation System dapat menentukan data telah sesuai atau tidak. Apabila memang tidak ada kesesuaian data antara OTA/*Voucher Agent* dan Bank, maka Reconciliation System dapat melakukan skema rekonsiliasi dan pelaporan sesuai dengan kewenangan.



Gambar 3.12 Alur transaksi pemesanan hotel pada fitur perjalanan *timeout/no response* dari OTA



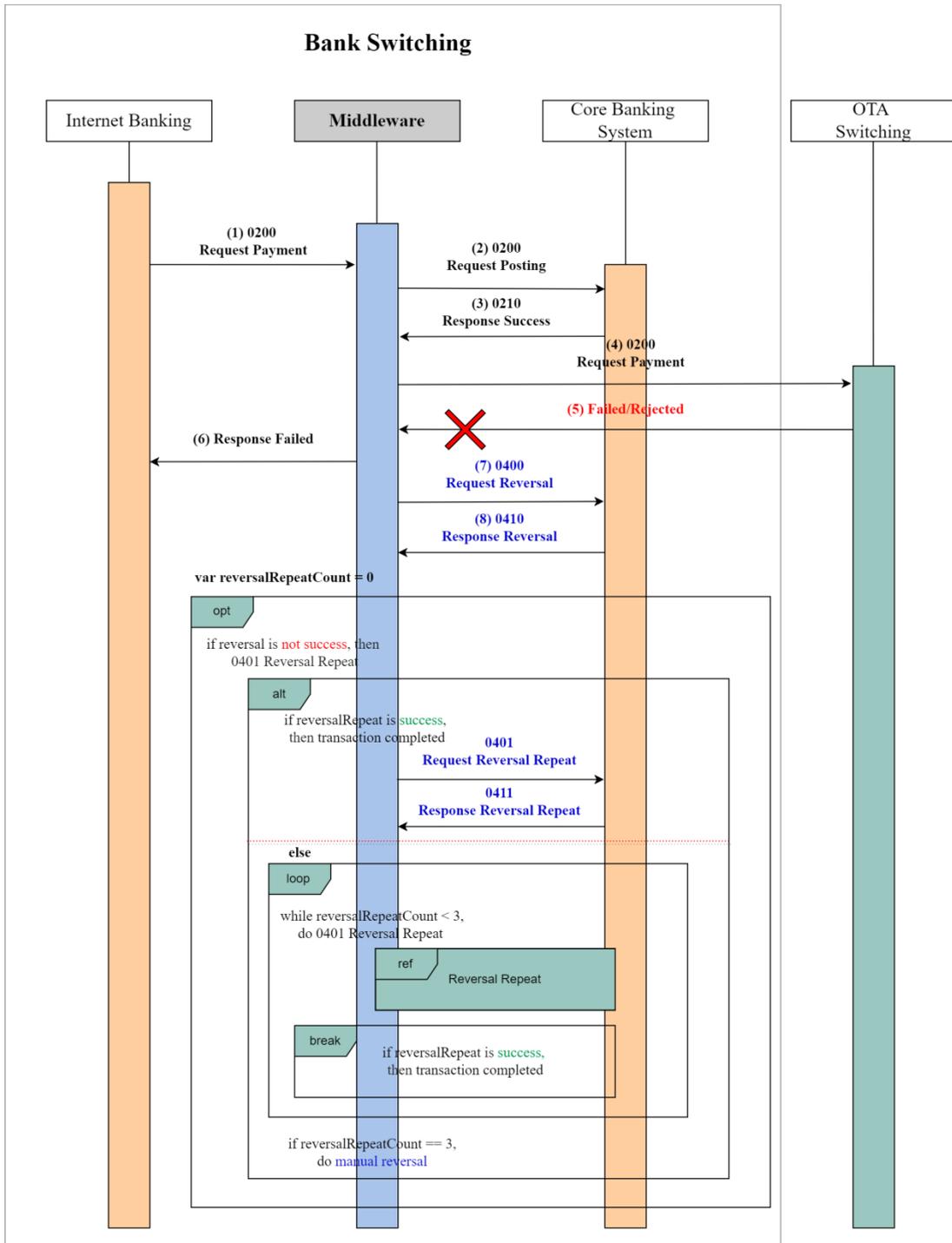
Gambar 3.13 Alur transaksi *top up* voucher pada fitur perjalanan *timeout/no response* dari Voucher Agent

Transaksi Pembayaran melalui Internet Banking *Rejected/Failed* OTA atau *Voucher Agent*

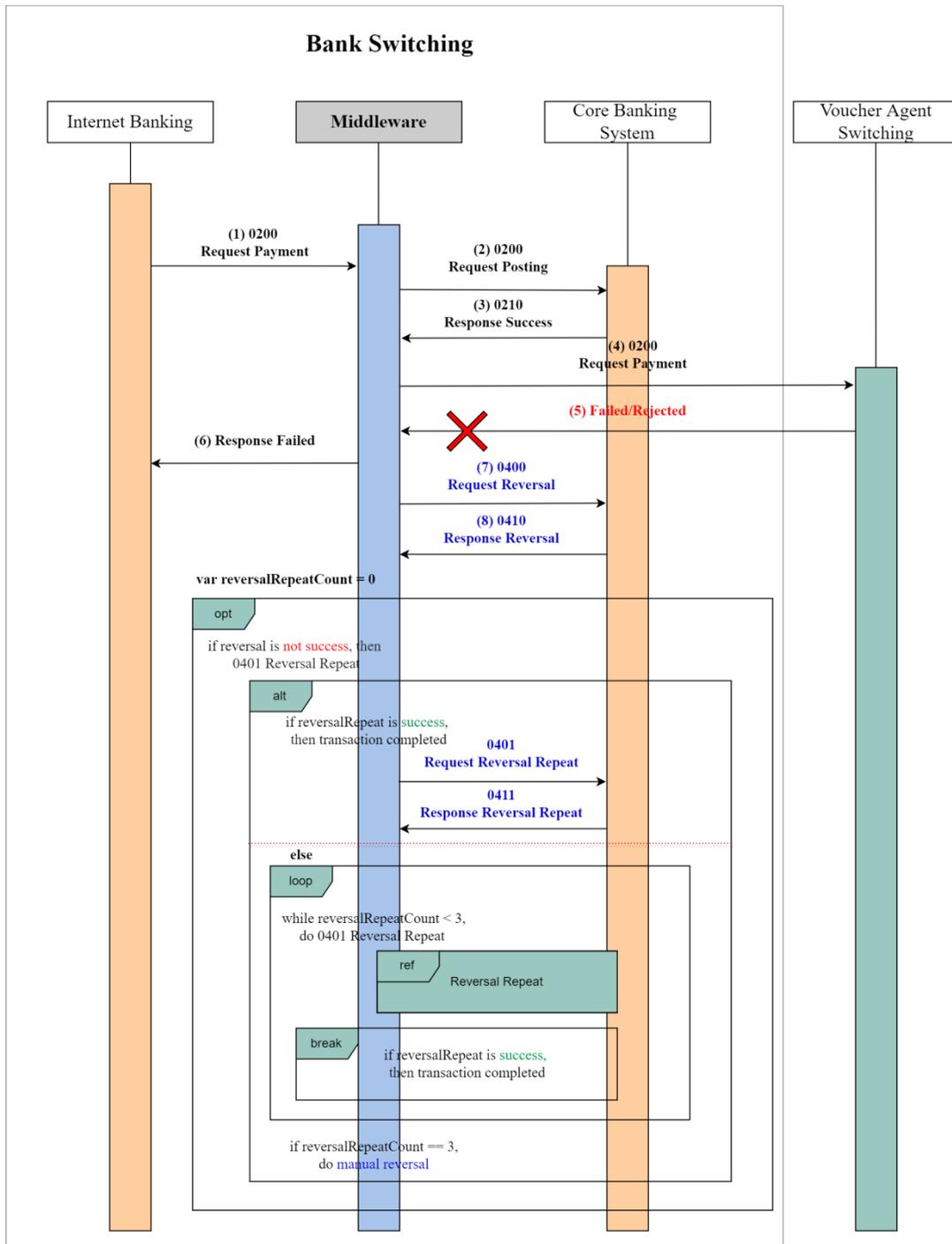
Selain mengalami *timeout/no response* dari OTA atau *Voucher Agent*, transaksi juga bisa ditolak atau memang gagal di OTA atau *Voucher Agent*. Untuk mengatasi transaksi yang gagal/ditolak oleh OTA atau *Voucher Agent*, perlu dilakukan *reversal* pada Core Banking. Mekanisme *reversal* untuk transaksi yang gagal/ditolak oleh OTA terjadi pada Pemesanan Hotel pada Fitur Perjalanan melalui OTA seperti pada Gambar 3.14. Untuk transaksi *Top Up Voucher* pada Fitur Perjalanan yang ditolak *Voucher Agent* dapat dilihat pada Gambar 3.15.

Reversal pada kasus ini hanya dilakukan di Core Banking. Middleware tidak perlu melakukan *request reversal* ke OTA karena transaksi sudah jelas gagal/ditolak oleh OTA sehingga tidak ada dana yang dikredit di OTA. Pada prosesnya, setelah menerima informasi ditolak oleh OTA, Middleware akan mengirimkan *request reversal* dengan kode 0400 ke Core Banking. Core Banking akan melakukan pembatalan pendebitan dana nasabah dan pencatatan transaksi. Setelah proses *reversal* berhasil di Core Banking, Core Banking akan mengirimkan respon *reversal* dengan kode 0410 ke Middleware.

Apabila transaksi *reversal* tidak memperoleh respon dari Core Banking, Middleware akan mengirimkan pesan dengan tipe 0401 (*Reversal Repeat Request*). Proses *reversal repeat* berhasil apabila Middleware menerima pesan 0411 dari Core Banking. Jika dalam tiga kali *request, reversal repeat* tetap tidak berhasil, Middleware akan berhenti mengirimkan *request reversal* dan proses *reversal* dana nasabah akan dilakukan secara manual.



Gambar 3.14 Alur transaksi pemesanan hotel pada fitur perjalanan gagal/ditolak oleh OTA



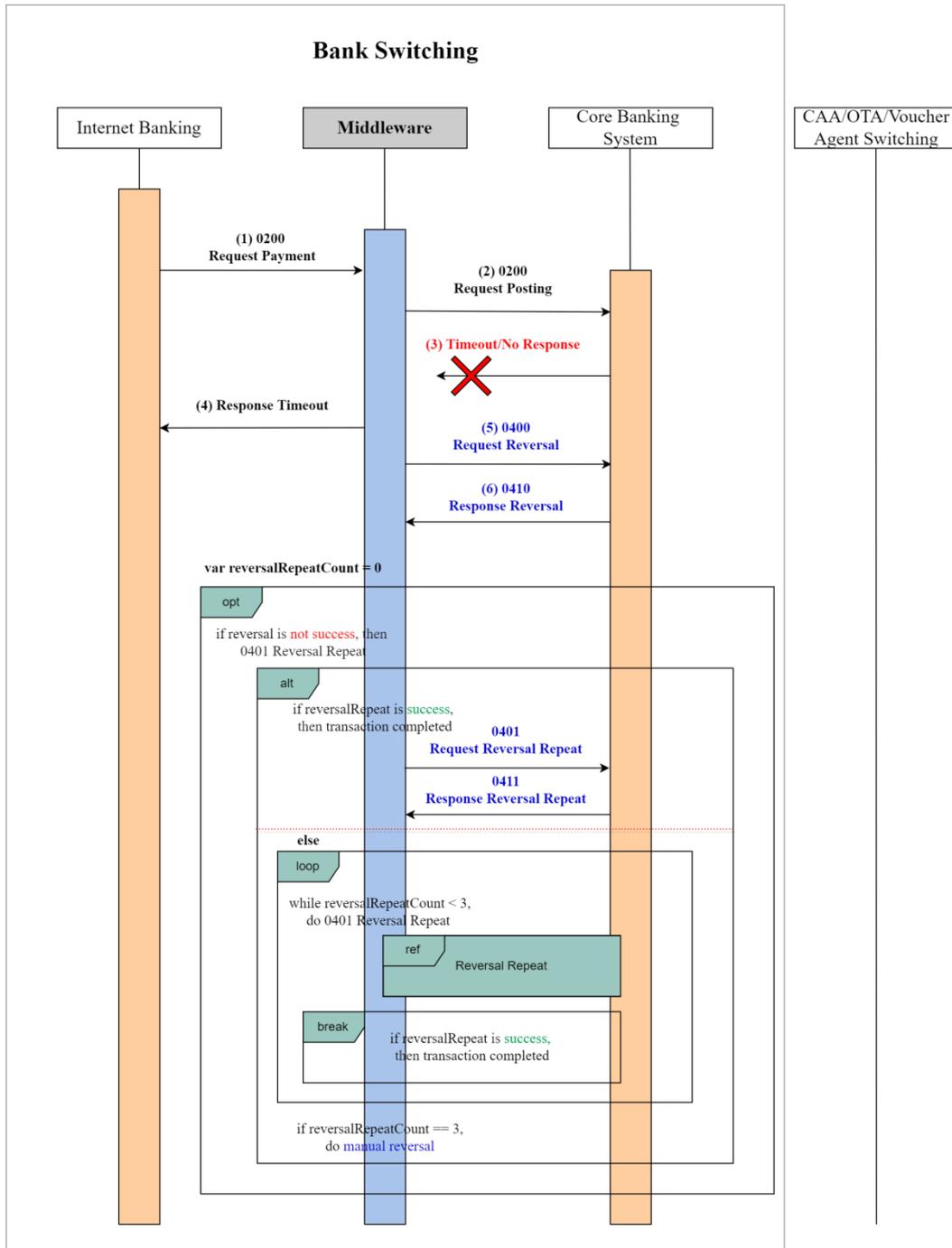
Gambar 3.15 Alur transaksi *top up* voucher fitur perjalanan gagal/ditolak oleh *Voucher Agent*

Transaksi Pembayaran melalui Internet Banking *Timeout/No Response* Core Banking

Tidak hanya CAA, OTA, ataupun *Voucher Agent*, proses yang terjadi pada Core Banking juga dapat menemui kendala, salah satunya terjadi *timeout* atau Core Banking tidak memberikan respon terhadap *request* yang dikirimkan Middleware. Hal ini menyebabkan Middleware tidak tahu dana nasabah telah didebit oleh Core Banking atau belum. Selain itu, apabila dana telah berhasil didebit, saldo nasabah akan berkurang padahal transaksi tidak berhasil. Untuk menghindari kesalahan pencatatan transaksi seperti yang disebutkan, harus dilakukan *reversal* pada Core Banking. CAA, OTA, ataupun *Voucher Agent* dalam hal ini tidak terlibat karena proses transaksi pembayaran telah menemui kendala sebelum sempat diproses oleh CAA, OTA, ataupun *Voucher Agent*.

Mekanismenya, setelah Middleware melewati batas tenggang transaksi dan tetap tidak ada respon dari Core Banking, Middleware akan meneruskan *response timeout* ke Internet Banking untuk diinformasikan ke nasabah bahwa transaksi tidak berhasil dilakukan karena *timeout/no response*. Middleware selanjutnya harus menjalankan mekanisme mengirimkan *request reversal* ke Core Banking untuk mengembalikan dana nasabah dengan mengirimkan *request reversal* dengan tipe pesan 0400. Setelah dana yang didebit di Core Banking berhasil dikembalikan, Core Banking harus memberikan respon 0410 (Reversal Response) sebagai bukti bahwa proses *reversal* di Core Banking berhasil.

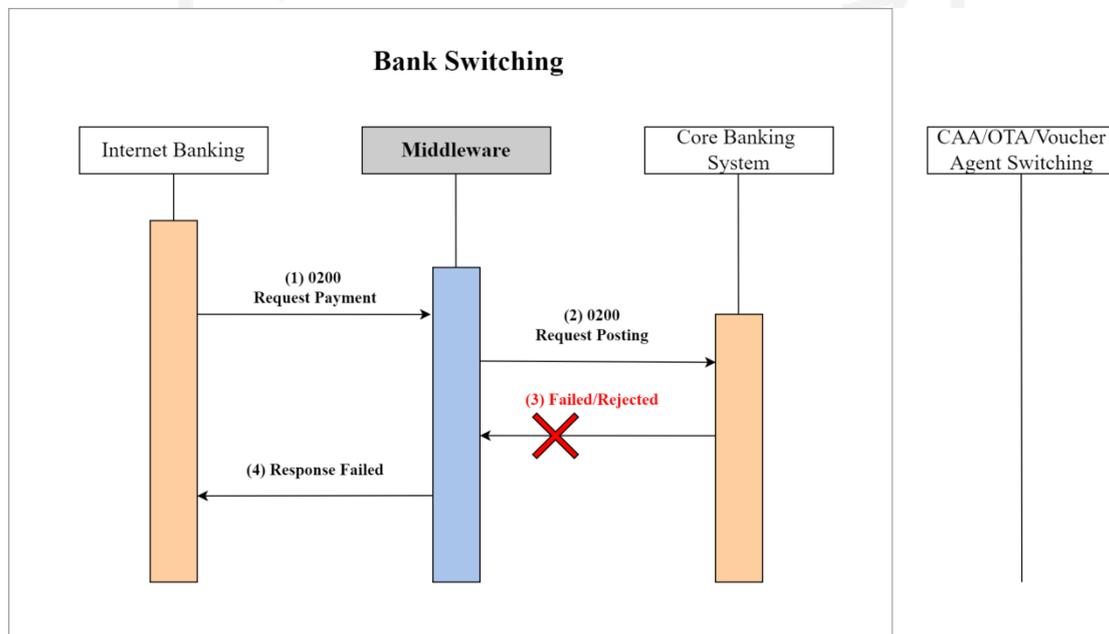
Apabila transaksi *reversal* tidak memperoleh respon dari Core Banking, Middleware akan mengirimkan pesan dengan tipe 0401 (Reversal Repeat Request). Proses *reversal repeat* berhasil apabila Middleware menerima pesan 0411 dari Core Banking. Jika dalam tiga kali *request, reversal repeat* tetap tidak berhasil, Middleware akan berhenti mengirimkan *request reversal* dan proses *reversal* dana nasabah akan dilakukan secara manual. Proses mekanisme *reversal* ketika transaksi *timeout* atau tidak memperoleh respon oleh Core Banking dapat dilihat pada Gambar 3.16.



Gambar 3.16 Alur transaksi *timeout/no response* dari Core Banking

Transaksi Pembayaran melalui Internet Banking *Rejected/Failed* Core Banking

Selain *timeout* dan *no response*, transaksi juga dapat ditolak atau gagal di Core Banking. Apabila hal ini terjadi, tidak ada *reversal* yang harus dilakukan. Middleware tidak perlu mengirimkan *request reversal* ke Core Banking karena transaksi sudah jelas gagal atau ditolak sehingga tidak ada pendebitan dana nasabah pada Core Banking. CAA, OTA, ataupun *Voucher Agent* juga tidak terlibat untuk kasus ini karena dana belum sempat diproses di CAA, OTA, ataupun *Voucher Agent*. Ketika Core Banking mengirimkan sinyal gagal/menolak, Middleware hanya perlu menyampaikan respon gagal/ditolak ke Internet Banking untuk ditampilkan ke nasabah. Alur dapat dilihat pada Gambar 3.17.



Gambar 3.17 Alur transaksi gagal/ditolak oleh Core Banking

3.3.4 Perancangan Skenario Pengujian

Pada proses perancangan skenario pengujian, dilakukan pengembangan kasus uji yang bertujuan untuk menjamin kualitas proyek secara keseluruhan. Skenario pengujian umumnya dibuat menggunakan Microsoft Excel. Skenario ini berisi kolom aksi yang akan dilakukan oleh Testing Engineer, hasil yang diharapkan, hasil yang diperoleh, serta keterangan sukses/gagal. Kolom hasil dan keterangan sukses/gagal akan diisi oleh Testing Engineer sebagai pihak yang melakukan pengujian. Hasil yang diperoleh akan diisi berupa aksi-aksi yang diperoleh ketika dilakukan pengujian, serta akan dibandingkan dengan hasil yang diharapkan. Apabila hasil yang diperoleh sesuai dengan yang diharapkan, transaksi sukses.

Apabila yang terjadi sebaliknya, dapat diartikan transaksi gagal. Terdapat dua tipe pada skenario pengujian, yakni *positive test* dan *negative test*. *Positive test* mendeskripsikan skenario jika aplikasi berjalan dengan benar (normal/tidak error), sedangkan *negative test* mendeskripsikan yang sebaliknya.

Setiap proyek tentunya memiliki skenario pengujian yang berbeda-beda, tergantung kebutuhan pengujian setiap proyek. Namun, secara umum, untuk transaksi melalui Internet Banking dapat diuji melalui sukses/tidaknya sistem yang berjalan sesuai aliran bisnis yang telah dirancang. Setiap skenario mulai dari *inquiry*, *payment*, dan *reversal* akan dilakukan pengujian. Lebih lengkapnya, transaksi yang dilakukan pengujian akan diperinci lagi berdasarkan *source of funds* (SoF).. SoF adalah sumber dana nasabah yang digunakan, yaitu *Current Account* (CA) atau giro, *Savings Account* (SA) atau tabungan, serta *Credit Card* (CC) atau kartu kredit.

3.4 Hasil Analisis dan Perancangan Sistem

Setiap proyek yang telah melalui analisis dan perancangan tentunya harus didokumentasikan sebagai syarat tahapan selanjutnya, serta menjadi panduan pengembang. Hasil tahapan analisis dan perancangan sistem setiap proyek dituang ke dalam tiga dokumen, yaitu dokumen ISO Message Format, System Design, dan Test Script. Dokumen hasil analisis dan perancangan sistem yang telah melalui tahapan *circulate approval* dari internal Middleware (sebagai IT Developer) dan eksternal (IT Infrastructure dan IT Project Manager) menjadi indikator keberhasilan laporan akhir. Hal ini dapat diartikan dokumen dapat masuk ke tahap pengembangan sehingga dapat menjadi panduan bagi tahap pengembang dan penguji. Keempat dokumen proyek, yaitu proyek Pembayaran Tagihan Perusahaan Gas Negara (PGN), Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB), *Top Up Voucher* pada Fitur Perjalanan melalui *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melalui *Online Travel Agent* (OTA) telah berhasil melalui *circulate approval* dan dapat dilanjutkan ke proses pengembangan.

3.4.1 Dokumen ISO Message Format

Dokumen ISO Message Format ditulis berdasarkan ISO 8583. Dokumen ini berisi format dan spesifikasi yang dibutuhkan dalam pertukaran pesan sistem *middleware* yang akan menjadi panduan bagi *programmer* dalam pembangunan sistem dan tentunya harus disesuaikan dengan dokumen teknis CAA/OTA. Pada perancangannya, biasanya Middleware SA dibantu oleh IT Application Developer. Namun, ada beberapa proyek yang dokumen ISO

Message Format-nya sudah dirancang sebelum mulai proyek. Hal itu dikarenakan ada proyek yang hanya membutuhkan *enhancement* pada proses bisnisnya atau *handover project*, sehingga tidak berdampak pada perancangan ISO Message Format.

Spesifikasi yang dijabarkan pada dokumen ISO Message Format berupa spesifikasi pertukaran data pada proses *inquiry*, *payment*, hingga *reversal*. Additional Data pada bit #48 dan/atau bit #62 yang bersifat *private* juga dijabarkan pada dokumen ini. Additional Data pada bit #48 dan bit #62 inilah yang akan menjadi inti dari perancangan ISO Message Format. Namun, perincian Additional Data pada bit #48 dan/atau bit #62 setiap proyek tidak akan dijabarkan pada laporan akhir ini karena sifatnya yang *private* dan *confidential*.

3.4.2 Dokumen System Design

Setiap proyek yang telah melalui tahap analisis kebutuhan dan perancangan sistem integrasi *middleware* pada transaksi melalui Internet Banking selanjutnya akan dituangkan dalam dokumen *system design*. Rancangan pada *system design* terdiri dari gambaran umum solusi, desain fungsional, desain teknis, cerita pengguna, dan diagram alir sistem. Hasil perancangan dari ISO Message Format yang telah dibuat turut dimasukkan ke dalam dokumen *system design*.

Gambaran Umum Solusi (*Solution Overview*)

Bagian ini memuat gambaran umum solusi yang akan dibangun dan dijelaskan secara ringkas, tetapi lengkap. Bagian ini biasanya berisi nama proyek, latar belakang pembangunan proyek, dan solusi yang ditawarkan.

Desain Fungsional dan Teknis (*Functional & Technical Design*)

Bagian ini menjelaskan secara rinci solusi yang ditawarkan. Bagian desain ini akan memaparkan komponen cerita pengguna, desain fungsional, dan desain teknis. Pada cerita pengguna, biasanya dijabarkan kebutuhan proyek, ruang lingkup pengembangan, hingga pengembangan yang dilakukan.

Pada desain fungsional, akan dijabarkan diagram alir proses, desain antarmuka sistem, deskripsi desain layar, dan desain sistem pelaporan (*report system*). Untuk dokumen *system design* pada integrasi *middleware*, penjabaran desain fungsional biasanya digambarkan melalui diagram alir, misalnya rancangan skenario *inquiry*, *payment*, dan *reversal*. Selain itu, hasil dari ISO Message Format proyek juga dijabarkan pada bagian ini.

Pada desain teknis, terdapat beberapa hal yang biasanya dilakukan, mulai dari menjabarkan program/modul lain, *database*, *screen*, *interface*, hingga *report* yang akan terpengaruh bila dilakukan pengembangan (jika ada). Namun, pada umumnya, perancangan yang dilakukan hanya sampai pada perancangan desain fungsional.

Desain Keamanan dan Kontrol (*Security and Control Design*)

Pada bagian ini dijabarkan pertimbangan keamanan dan kontrol termasuk matriks keamanan untuk berbagai fungsi, layar, dan lain-lain yang dapat dilakukan oleh kelompok pengguna, aplikasi, jejak audit, log transaksi, dan periode penyimpanan yang berbeda. Namun, biasanya Middleware SA tidak merancang hingga ke bagian ini.

Desain Pencadangan dan Retensi Data (*Backup and Data Retention Design*)

Bagian ini mencantumkan data yang perlu dicadangkan sebagai langkah preventif bila terjadi masalah ke depannya, jenis pencadangan, frekuensi, lokasi, dan periode penyimpanan. Akan tetapi, Middleware SA tidak merancang hingga ke bagian ini.

3.4.3 Dokumen Test Script

Dokumen Test Script merupakan penjabaran hasil dari rancangan skenario pengujian. Skenario pengujian ini akan menjadi panduan pengujian yang akan dilakukan oleh Testing Engineer. Biasanya, peran Middleware SA hanya merancang skenario pengujian kemudian didiskusikan bersama Testing Engineer. Hasil rancangan skenario juga akan dikirim ke IT Application Developer untuk dilakukan rewiu. Apabila dari Testing Engineer dan IT Application Developer telah menyetujui rancangan yang diajukan, rancangan skenario pengujian bisa digunakan oleh Testing Engineer setelah proyek masuk ke tahap pengujian.

BAB IV

REFLEKSI PELAKSANAAN MAGANG

Bab ini menjelaskan refleksi pelaksanaan magang yang terdiri dari relevansi akademik dan pembelajaran magang. Relevansi akademik berkaitan dengan kesenjangan antara teori pada Bab II dibandingkan dengan pelaksanaan di lapangan, sedangkan pembelajaran magang memaparkan manfaat, tantangan, dan hambatan yang diperoleh selama proses analisis dan perancangan sistem integrasi *middleware*.

4.1 Relevansi Akademik

Subbab ini menganalisis kesenjangan antara landasan teori yang telah dibahas di Bab II dengan pelaksanaan di lapangan selama magang. Beberapa hal yang akan dibahas, yaitu terkait pengembangan integrasi *middleware*, perancangan ISO Message Format, serta manajemen proyek dan implementasi Waterfall SDLC.

4.1.1 Pengembangan Integrasi *Middleware*

Mempelajari sistem *middleware* merupakan hal yang menantang karena sejauh ini belum ada mata kuliah di Program Studi Informatika – Program Sarjana yang berhubungan dekat dan membahas sistem *middleware*. Pada pelaksanaan di lapangan, ternyata ruang lingkup teknologi informasi sangat luas. Berdasarkan pengalaman magang, diketahui bahwa *middleware* merupakan sistem yang umum dimiliki *enterprise*, terlebih dalam penerapannya dalam sistem perbankan. Setiap perusahaan bisa jadi memiliki model, metode, hingga aplikasi *middleware* yang berbeda.

Tanggung jawab seorang *Middleware SA* dalam pengembangan integrasi *middleware* selama kegiatan magang mencakup pembuatan dokumen ISO 8583 Message Format, System Design, dan Test Script. Pada pengembangan yang dituangkan pada setiap dokumen, setiap sistem yang dijabarkan sebenarnya lebih rinci daripada yang dijabarkan pada dokumen laporan akhir ini. Sistem *middleware* yang ada di Bank berbagai macam. Begitu pula sistem *core banking* dan mitra *switching 3rd Party* yang diintegrasikan oleh sistem *middleware*. Namun, karena keterbatasan akses dan sifatnya yang *confidential*, Bank sebagai mitra magang tidak memperkenankan untuk menjelaskan sistem secara rinci (misalnya, penyebutan nama aplikasi, modul-modul yang dimiliki aplikasi, deskripsi *software* dan *hardware*, dan lain sebagainya yang mendeskripsikan dengan jelas aplikasi dan perangkat yang dimiliki

Bank). Oleh karena itu, penjabaran pada laporan akhir ini dilakukan secara *high-level* dan *se-general* mungkin, serta turut disesuaikan dengan teori-teori yang ditemukan dari berbagai sumber.

4.1.2 Perancangan ISO Message Format

Sebelum melakukan perancangan sistem, biasanya Middleware SA perlu merancang dokumen ISO Message Format. Dokumen ini berisi spesifikasi ISO 8583 yang dibutuhkan *middleware* dalam pertukaran data antarsistem, mulai dari MTI hingga pemetaan *bitmap*. Secara teori, jika dilihat pada Gambar 2.3, komponen MTI yang dimiliki ISO 8583 memiliki panjang 4 *bytes* dengan *byte* pertama akan diisi dengan versi ISO 8583, *byte* kedua tipe transaksi, *byte* ketiga fungsi pesan, dan *byte* keempat diisi dengan asal pesan. Tipe pesan pada *byte* kedua bisa didefinisikan hingga sembilan jenis pesan. Namun, dalam pengembangan pada sistem *middleware*, paling tidak hanya ada tiga tipe pesan yang dibutuhkan sistem *middleware*, yaitu x2xx, x4xx, dan x8xx. Rincian tipe pesan yang biasa digunakan pada sistem *middleware* dijabarkan pada Tabel 4.1.

Tabel 4.1 Klasifikasi tipe pesan pada MTI

Tipe	Definisi	Kegunaan	Asal	Tujuan
0200	Financial Transaction Request	<i>Inquiry & Payment</i>	CA	CAA
0210	Financial Transaction Response	<i>Inquiry & Payment</i>	CAA	CA
0400	Reversal Request	<i>Reversal transaction</i>	CA	CAA
0410	Reversal Response	<i>Reversal transaction</i>	CAA	CA
0800	Network Management Request	<i>Sign-on, Sign-off, Cut-off, Echo test</i>	CA	CAA
0810	Network Management Response	<i>Sign-on, Sign-off, Cut-off, Echo test,</i>	CAA	CA

Apabila dianalisis dari proyek-proyek yang dikerjakan di *middleware*, secara garis besar integrasi yang ada di *middleware* lebih sering memerlukan tiga tipe pesan dalam pemrosesan transaksi. Hal ini dikarenakan sistem manajemen transaksi yang dimiliki divisi Middleware hanya mendukung tiga seri pesan tersebut untuk mengintegrasikan antarsistem yang ada di internal dan eksternal Bank. Untuk tipe pesan lain, tidak digunakan pada transaksi yang dijalankan sistem *middleware*, misalnya untuk seri x5xx berkaitan dengan rekonsiliasi digunakan oleh *reconciliation system* yang ada di Bank.

Lebih rinci lagi, sebenarnya ada klasifikasi tipe pesan yang digunakan untuk transaksi antar modul pada sistem *middleware*, yaitu *identifier* 9xxx yang digunakan secara privat pada sistem *middleware*. Namun, pada pelaksanaan magang, Middleware SA hanya

mempelajarinya sebatas keperluan pengetahuan saja, serta tidak diberi akses atau proyek yang berkaitan dengan komunikasi antarmodul tersebut.

4.1.3 Manajemen Proyek dan Implementasi Waterfall SDLC

Berdasarkan pemaparan pada subbab manajemen proyek pada bab sebelumnya, telah dijelaskan bahwa metode SDLC yang digunakan Bank dalam pengembangan sistem adalah *agile* dan *waterfall*. Namun, untuk Divisi Middleware, metode yang digunakan hanya Waterfall SDLC. Apabila dianalisis berdasarkan kelebihan dan kekurangan metode Waterfall SDLC pada pemaparan teoritis pada Bab II, terdapat kesesuaian antara kajian teori dan penerapan metode Waterfall SDLC yang digunakan pada pengembangan sistem pada Divisi Middleware. Setiap proyek yang dikerjakan pada Divisi Middleware memiliki persyaratan sistem yang jelas. Hal ini dikarenakan setiap proyek yang dibangun merujuk pada dokumen *feasibility study* yang telah dirancang sebelumnya. *Feasibility study* berisi ringkasan proyek dan kelayakan keseluruhan sistem bank dalam pengembangan proyek. Pada dokumen *feasibility study* juga telah dipaparkan titik awal, akhir, dan estimasi pengerjaan proyek setiap divisi.

Setelah perilisian dokumen *feasibility study*, setiap divisi bisa mulai menganalisis dan merancang kebutuhannya masing-masing. *Feasibility study* digunakan oleh setiap sistem analisis pada setiap divisi, tidak hanya Middleware, dalam melakukan analisis dan perancangan sistem. Melalui persyaratan yang jelas dari dokumen *feasibility study* (secara keseluruhan proyek), Middleware SA dapat melanjutkan analisis dan perancangan yang difokuskan pada sistem *middleware* dan didokumentasikan pada dokumen ISO Message Format, System Design, dan Testing Script.

Kesesuaian lainnya antara landasan teori dan penerapan di lapangan adalah anggota tim yang telah menyelesaikan fasenya atau belum masuk ke fasenya cenderung duduk diam. Hal ini dapat dikaji dalam sebuah proyek yang sedang berjalan. Sesuai dengan sifat Waterfall SDLC yang sekuensial, ketika Middleware SA sedang merancang dokumen ISO Message Format, System Design, maupun Test Script, anggota proyek lainnya tidak/belum bergerak dalam proyek yang sama, kecuali dalam kegiatan diskusi dan konsultasi (pemberian umpan balik dan review). Misalnya, pengembang yang menunggu rancangan dari Middleware SA dan pengujian yang menunggu pengembangan sistem dari pengembang. Sebaliknya, ketika Middleware SA telah selesai dalam melakukan analisis dan perancangan suatu proyek, serta dokumen telah selesai dilakukan *circulate approval* dan masuk ke tahap pengembangan,

sebenarnya ia hanya akan duduk diam sampai proyek selesai (peran dalam proyek telah selesai).

Namun, hal ini tidak terlalu jelas terlihat karena proyek yang ada pada divisi Middleware cukup banyak sehingga setiap analis sistem, pengembang, dan penguji tidak hanya memegang satu proyek dalam satu waktu (paralel). Selain itu, pada praktik di lapangan, ketika satu proyek telah selesai dirancang oleh Middleware SA, mentor akan memberikan proyek baru atau langsung diarahkan ke proyek yang sebelumnya telah diberikan sehingga tidak ada kesempatan untuk duduk diam. Hal inilah yang menjadi kesenjangan antara kajian teori dengan praktik di lapangan.

Berdasarkan kajian teori, idealnya sebuah pengembangan sistem yang menggunakan Waterfall SDLC bergerak secara sekuensial mulai dari tahapan analisis, perancangan, pengkodean, pengujian, dan pemeliharaan. Namun, pengalaman yang dirasakan sebagai Middleware SA, hanya sampai di tahapan analisis dan perancangan hingga tahap *circulate approval* selesai. Pada proses setelahnya, Middleware SA tidak lagi dilibatkan sehingga tidak ada pengalaman dalam mengawasi jalannya proyek hingga selesai. Proses pengawasan dan pengendalian proyek selanjutnya dilakukan oleh Middleware SA lainnya yang secara hierarki berada di atas pemegang.

4.2 Pembelajaran Magang

Berbagai manfaat, hambatan, dan tantangan telah diperoleh selama kegiatan magang, baik teknis maupun nonteknis. Hal ini terutama dalam aktivitas analisis dan perancangan sistem integrasi *middleware* yang mengasah keilmuan terkait pengembangan sistem informasi perbankan, kemampuan analisis, serta komunikasi verbal dan nonverbal.

4.2.1 Analisis dan Perancangan Sistem *Middleware* pada Internet Banking

Terjun langsung ke proyek dalam skala profesional tentunya memerlukan adaptasi, baik terkait pengetahuan maupun keterampilan teknis dan nonteknis. Sebelum memulai kegiatan magang, dilakukan pelatihan bersama vendor terkait aplikasi *middleware*, mulai dari pengenalan modul-modul yang ada di aplikasi *middleware*, hingga alur sebuah transaksi. Bank sebagai lembaga keuangan memiliki ribuan tipe transaksi dan setiap transaksi tersebut memiliki alur yang berbeda-beda. Mulai dari sistem dan pemangku kepentingan yang terlibat, alur transaksi, hingga skenario pengembalian dana apabila transaksi tidak berhasil dilakukan. Pengalaman menjadi Middleware SA memberikan manfaat, salah satunya keilmuan tentang sistem perbankan yang kompleks.

Bersamaan dengan pengerjaan proyek, pada awal kegiatan magang, dilakukan analisis *existing middleware system* untuk menambah pemahaman pengetahuan terkait sistem *middleware*. Melalui analisis *existing middleware system*, ditemukan berbagai kejanggalaan proses bisnis yang saat ini sedang berjalan di Bank. Misalnya, tidak ditemukan adanya skema pengembalian dana yang harusnya dilakukan saat transaksi gagal. Padahal jika dianalisis dari *file* parameter yang ada di *middleware*, transaksi tersebut harus dilakukan *reversal* jika transaksi gagal. Kejanggalaan-kejanggalaan ini akan dicatat oleh Middleware SA dan divisualisasikan dengan diagram sekuensial. Hasil analisis ini dapat menjadi masukan untuk Middleware dan vendor dalam melakukan perbaikan proses bisnis sistem sehingga Bank dapat meningkatkan performa pelayanan ke nasabah.

Kanal transaksi yang terlibat dalam integrasi *middleware* sebenarnya ada banyak, misalnya ATM, EDC, Internet Banking, Mobile Banking, hingga SMS Banking. Tidak hanya pengelolaan fungsi dan integrasi transaksi, pada kegiatan magang juga didapatkan pengetahuan bahwa peran *middleware* dalam kegiatan perbankan mencakup kliring, pelaporan transaksi, pemeliharaan informasi kartu dan klien, serta persiapan dan otentikasi transaksi. Adapun dari tujuh proyek yang diberikan selama kegiatan magang, empat di antaranya merupakan pengembangan fitur transaksi melalui kanal Internet Banking sehingga topik ini dipilih menjadi laporan akhir. Pada proses analisis dan perancangan sistem pada Internet Banking, ditemukan bahwa terdapat benang merah alur proses bisnis antarproyek. Secara umum, pada alur transaksi *payment* setiap transaksi *payment* terdapat kemiripan mengenai sistem-sistem yang dilalui oleh sebuah transaksi, yang membedakan adalah proses *inquiry* serta skenario *reversal* yang akan memengaruhi rancangan skenario pengujian.

Proyek Pembayaran Tagihan PGN dan Pembayaran Tagihan PBB memiliki kesamaan alur proses bisnis dalam proses *reversal*, meskipun CAA yang terlibat antara keduanya berbeda. Hal ini kedua Biller yang bekerja sama dengan CAA mendukung adanya *reversal*. Adanya *reversal* membuat transaksi tidak perlu melalui tahapan rekonsiliasi jika transaksi berstatus *suspect* sebab apabila status transaksi *suspect*, terdapat skenario *reversal* yang akan dijalankan. Benang merah proyek juga dapat terlihat pada proyek *enhancement* pada Fitur Perjalanan, yaitu Pemesanan Hotel melalui OTA dan *Top Up Voucher* melalui *Voucher Agent*. Bank, melalui sistem *middleware*, sama-sama berkomunikasi secara langsung dengan kedua agen tersebut tanpa perlu *aggregator* seperti CAA. Kedua agen, yakni OTA dan *Voucher Agent* sama-sama tidak menyediakan *reversal* sehingga apabila transaksi tidak berhasil karena *timeout*, harus dilakukan rekonsiliasi.

Melalui analisis terhadap semua transaksi yang sedang berjalan di Bank serta analisis dan perancangan terhadap empat proyek integrasi *middleware* pada kanal Internet Banking, ditemukan bahwa proses bisnis setiap transaksi berbeda. Setiap sistem, kanal, *source of funds*, serta tipe transaksi juga berbeda. Sistem dan modul pada *middleware* yang terlibat berbeda, begitu pula dengan sistem dan modul Core Banking. Tidak hanya itu, status gagal maupun *suspect* pada setiap transaksi juga memiliki perlakuan yang berbeda, ada yang menyediakan skenario *reversal*, ada pula yang tidak sehingga harus melalui proses rekonsiliasi. Middleware berperan dalam setiap perintegrasian transaksi-transaksi tersebut sehingga setiap sistem dan aplikasi yang dimiliki Bank dapat berkomunikasi.

Pada proses perancangan sistem, terdapat hambatan yang ditemui, yaitu apabila terdapat perubahan persyaratan proyek pada dokumen *feasibility study* ketika dokumen telah melalui proses *circulate approval*. Seorang Middleware SA harus menunggu hingga versi terbaru *feasibility study* terbit serta melakukan diskusi dan perubahan pada dokumen-dokumen sesuai dengan persyaratan terbaru. Selain itu, hambatan lainnya terjadi apabila terdapat perbedaan pemahaman dengan rekan di internal maupun eksternal divisi terkait perancangan yang telah dilakukan sehingga proses *approval* diinterupsi. Apabila memang yang dituang dalam dokumen salah, dokumen harus direvisi. Proses revisi menjadi tantangan karena proses *circulate approval* akan diulang dari awal sementara proyek harus tetap berjalan sesuai dengan *timeline*.

4.2.2 Keahlian Teknis dan Non-Teknis

Selama kegiatan magang, telah dilakukan koordinasi dengan berbagai pemangku kepentingan dalam menetapkan ruang lingkup proyek, tujuan sistem, dan persyaratan proyek. Kontribusi lainnya yang diberikan adalah menganalisis, merancang, serta mengatur spesifikasi yang dibutuhkan proyek, yang dituangkan dalam bentuk pemetaan data dan diagram alur dalam bahasa teknis yang dapat diikuti oleh pengembang dan penguji. Proses ini tentunya memerlukan keahlian seperti berpikir analitis, serta berkomunikasi secara verbal (diskusi proyek dan presentasi) dan nonverbal (pendokumentasian).

Dokumen teknis dan spesifikasi mengasah kemampuan berpikir analitis dalam perancangannya. Seorang Middleware SA harus menganalisis setiap persyaratan dan langkah proses bisnis yang akan dirancang sehingga sesuai dengan tujuan proyek. Keahlian teknis dibutuhkan dalam pembuatan dokumen, misalnya ketika membuat diagram. Dalam menentukan diagram yang tepat pada dokumen System Design, diperlukan pengetahuan

mengenai diagram yang biasa digunakan dalam dokumen teknis serta kemampuan untuk merancangya sesuai dengan kebutuhan proyek. Hal ini terutama agar pengembang dan penguji dapat mengikuti persyaratan yang telah dirancang pada dokumen tersebut.

Setiap proyek yang dikerjakan umumnya menggunakan diagram sekuensial dan *flowchart*. Diagram sekuensial umumnya digunakan pada perancangan integrasi *middleware* antarsistem di Bank, sedangkan *flowchart* lebih banyak digunakan dalam proyek-proyek *enhancement* proses bisnis, misalnya pada proyek Perubahan Bukti Setor Haji dan Otomasi Jurnal *Fee Waiver*.



BAB V

PENUTUP

5.1 Kesimpulan

Analisis dan perancangan sistem integrasi *middleware* pada kanal Internet Banking memudahkan pengembang dan penguji dalam melakukan pengembangan dan pengujian sistem. Perancangan sistem ini melibatkan beberapa pemangku kepentingan, di antaranya CA, CAA, OTA, dan *Voucher Agent*. Komunikasi ketiga pemangku kepentingan tersebut harus disepakati bersama. Pihak CA, dalam hal ini Bank, berkomunikasi dengan CAA, OTA, maupun *Voucher Agent* menggunakan standar ISO 8583. Sistem dirancang menggunakan metode SDLC Waterfall melalui pendekatan *model-driven analysis*, dengan tahapan yang dilakukan di antaranya: analisis kebutuhan sistem, perancangan *inquiry*, *payment*, dan *reversal transaction*, serta perancangan skenario pengujian.

Laporan Akhir “Analisis dan Perancangan Sistem Integrasi *Middleware* pada Internet Banking” telah berhasil mencapai tujuan dengan indikator keberhasilan, yaitu setiap dokumen yang dibuat telah berhasil melalui tahapan *circulate approval* untuk dilakukan pengembangan sehingga dapat menjadi panduan bagi pengembang dan penguji. Terdapat empat proyek yang dilakukan analisis dan perancangan pada laporan akhir ini, di antaranya Pembayaran Tagihan Perusahaan Gas Negara (PGN), Pembayaran Tagihan Pajak Bumi dan Bangunan (PBB), *Top Up Voucher* pada Fitur Perjalanan melalui *Voucher Agent*, dan Pemesanan Hotel pada Fitur Perjalanan melalui *Online Travel Agent* (OTA). Adapun dalam proses analisis dan perancangan, terdapat benang merah antarproyek, yaitu alur bisnis *payment* yang sama, ada tidaknya proses *inquiry*, serta ada tidaknya *reversal* dari *3rd Party*.

5.2 Saran

Adapun untuk pengembangan sistem yang lebih baik, saran yang diberikan terhadap analisis dan perancangan sistem, yaitu perlu dilakukan analisis dan perancangan keamanan serta kontrol sistem dari sisi *Middleware* pada dokumen System Design setiap proyek. Pertimbangan perancangan keamanan ini bertujuan untuk melindungi data nasabah pada sisi *Middleware*, serta sebagai salah satu langkah preventif apabila terjadi masalah di masa depan.

DAFTAR PUSTAKA

- Adel, A., & Abdullah, B. (2015). A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model. *IJCSI International Journal of Computer Science Issues*, *12*(1), 106–111. https://www.academia.edu/10793943/A_Comparison_Between_Three_SDLC_Models_Waterfall_Model_Spiral_Model_and_Incremental_Iterative_Model
- Al-Ghamdi, A. A. M., & Saleem, F. (2014). Enterprise application integration as a middleware: Modification in data & process layer. *Proceedings of 2014 Science and Information Conference, SAI 2014, April*, 698–701. <https://doi.org/10.1109/SAI.2014.6918263>
- Amlani, R. D. (2012). Advantages and limitations of different SDLC models. *International Journal of Computer Applications & Information Technology*, *1*(3), 6–11.
- Bank. (2011). *Internal Document (telah didefinisikan secara high-level)*.
- Bassil, Y. (2015). A Simulation Model for the Spiral Software Development Life Cycle. *International Journal of Innovative Research in Computer and Communication Engineering*, *03*(05), 3823–3830. <https://doi.org/10.15680/ijircc.2015.0305013>
- da Cruz, M. A. A., Rodrigues, J. J. P. C., Sangaiah, A. K., Al-Muhtadi, J., & Korotaev, V. (2018). Performance evaluation of IoT middleware. *Journal of Network and Computer Applications*, *109*, 53–65. <https://doi.org/10.1016/j.jnca.2018.02.013>
- Deddy, R., Lumbantobing, H., Situmorang, C., Silalahi, S., & Pasaribu, M. S. (2021). Pendekatan Model Driven Development. *Jurnal Nasional Teknik Informatika dan ElektrO (JURNALTIO)*, *02*(01), 1–9.
- Dubey, M. A., Jain, M. A., & Mantri, M. A. (2015). Comparative Study: Waterfall V/S Agile Model. *International Journal of Engineering Sciences & Research Technology*, *4*(3), 70–75.
- Farahzadi, A., Shams, P., Rezazadeh, J., & Farahbakhsh, R. (2018). Middleware technologies for cloud of things: a survey. *Journal of Digital Communications and Networks*, *4*(3), 176–188. <https://doi.org/10.1016/j.dcan.2017.04.005>
- Hasibuan, B., & Soecahyadi, L. R. (2020). *Perancangan Middleware dalam Pelayanan Publik* (F. F. Putra (ed.); 1 ed.). Sahid University Press.
- Heryani, I. I. P., Simanjuntak, M., & Maulana, A. (2020). Perilaku Penggunaan Internet

- Banking Sebagai Alat Transaksi. *Jurnal Aplikasi Manajemen dan Bisnis*, 6(1), 86–95.
- Kassab, W., & Darabkh, K. A. (2020). A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations. *Journal of Network and Computer Applications*, 163(September 2019), 102663. <https://doi.org/10.1016/j.jnca.2020.102663>
- Kortuem, G., Kawsar, F., Sundramoorthy, V., & Fitton, D. (2010). Smart objects as building blocks for the internet of things. *IEEE Internet Computing*, 14(1), 44–51. <https://doi.org/10.1109/MIC.2009.143>
- Kramer, M. (2018). Lifecycle: An Analyses Based on the Waterfall Model. *Review of Business & Finance Studies*, 9(1), 77–84.
- Kumar, M. (2018). A Comparative Study of Universally Accepted SDLC Models for Software Development. *International Journal of Scientific Research in Science and Technology*, 4(5), 1084–1092. www.ijrst.com
- Kuntoro, N., & Ladjamuddin, S. M. (2019). Rancangan Aplikasi Simulasi Host-to-Host Transaksi ATM Bersama dengan ISO 8583. 8(1), 16–22.
- Margaretha, F. (2015). Dampak Electronic Banking terhadap Kinerja Perbankan Indonesia. In *Jurnal Keuangan dan Perbankan* (Vol. 514, Nomor 3). <http://jurkubank.wordpress.com>
- McCormick, M. (2012). Waterfall and Agile Methodology. *MPCS Inc*, 8/9/2012, 1–8.
- Mian, A. N., Hameed, A., Khayyam, M. U., Ahmed, F., & Beraldi, R. (2015). *Enhancing Communication Adaptability Between Payment Card Processing Networks*. March, 1–7.
- Mulyani, S. (2016). *Metode Analisis dan Perancangan Sistem* (A. Zulvani (ed.); 2 ed.). ABDI SISTEMATIKA.
- Nur Utomo, A. (2017). Rancang Bangun Format Pesan ISO 8583 Sistem Host-to-Host untuk Collection Agent Aggregator (CAA) pada Kerjasama Brilink Bank BRI. *Jurnal Rekayasa Informasi*, 6(2). www.aryonurutomo.blogspot.com
- Perset, K. (2010). The Economic and Social Role of Internet Intermediaries. In *Organisation for Economic Co-operation and Development (OECD)*. <http://www.oecd.org/dataoecd/49/4/44949023.pdf>
- Pikkarainen, T., Pikkarainen, K., Karjaluoto, H., & Pahnla, S. (2004). Consumer acceptance of online banking: An extension of the technology acceptance model. *Internet Research*, 14(3), 224–235. <https://doi.org/10.1108/10662240410542652>
- Royce, W. W. (2021). Managing the Development of Large Software Systems (1970). In H. R. Lewis (Ed.), *Ideas That Created the Future: Classic Papers of Computer Science*

(hal. 321–332). <https://doi.org/10.7551/mitpress/12274.003.0035>

- Sailaja, P., & Thamodaran, V. (2016). Customer Perception on Security Systems in E-Banking Services with special references to ATM. *Indian Journal of Commerce & Management Studies*, VII(6), 66–72.
- Salve, S. M., Samreen, S. N., & Valmik, N. K. (2018). A Comparative Study on Software Development Life Cycle Models. *International Research Journal of Engineering and Technology (IRJET)*, 05(02), 696–699.
- Sanjaya, R. P., Chulafak, G. A., & Yudiantika, A. R. (2015). *Dukungan Sistem Operasi*. <https://www.yumpu.com/id/document/view/51486264/resume-teknik-elektro-ugm>
- Solution, T. D. (2018). *Billing Payment Aggregator*. 1–2. https://www.telkomdigitalsolution.com/uploads/2018/08/Katalog_Produk_7.1_Billing-Payment-Aggregator.pdf
- Standardization, I. O. for. (1987). *ISO 8583: 1987 Bank Card Originated Messages - Interchange Message Specifications - Content for Financial Transactions*. www.iso.org
- Syarif, A. F., Ngesti Basuki, P., & Wijaya, A. F. (2018). Analisa Kinerja Sistem Informasi / Teknologi Informasi pada PT. Bank Central Asia Menggunakan Kerangka IT Balanced Scorecard. *Jurnal Sistem Informasi (JSI)*, 10 No. 1(1), 1491–1502. <http://ejournal.unsri.ac.id/index.php/jsi/index>
- Titi, R., & Sari, K. (2017). Terapan Pembayaran Elektronik PBB Dengan Host To Host Iso 8583 Pada Pemerintahan Daerah. *Jurnal Ilmiah Teknik Informatika (FORMAT)*, 6 No. 2, 47–56.
- Yunandar, R. T. (2016). Implementasi ISO 8583 Untuk Host To Host Mahasiswa Bina Sarana Informatika melalui Channelbiller PT Finnet Indonesia Berbasis AIX. *Jurnal Suara Wawasan Sukabumi (SWABUMI): Ilmu Komputer, Manajemen, dan Sosial*, IV No.2, 97–113.