

TUGAS AKHIR - KS14 1501

**PEMBANGUNAN SISTEM MONITORING DEBITUR
DENGAN MENGGUNAKAN ARSITEKTUR *THREE-
TIER* (STUDI KASUS : BANK UMUM DI INDONESIA)**

***DEVELOPMENT OF DEBTOR MONITORING
SYSTEM USING THREE-TIER ARCHITECTURE
(CASE STUDY : COMMERCIAL BANKS IN
INDONESIA)***

I GEDE ADITYA WIRATAMA
NRP 5212 100 161

Dosen Pembimbing I
BEKTI CAHYO HIDAYANTO, S.Si., M.Kom.

Dosen Pembimbing II
FEBY ARTWODINI MUQTADIROH S.Kom, MT.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS14 1501

**PEMBANGUNAN SISTEM MONITORING
DEBITUR DENGAN MENGGUNAKAN
ARSITEKTUR *THREE-TIER* (STUDI KASUS :
BANK UMUM DI INDONESIA)**

***DEVELOPMENT OF DEBTOR MONITORING
SYSTEM USING THREE-TIER
ARCHITECTURE (CASE STUDY :
COMMERCIAL BANK IN INDONESIA)***

I GEDE ADITYA WIRATAMA
NRP 5212 100 161

Dosen Pembimbing I
BEKTI CAHYO HIDAYANTO, S.Si., M.Kom.

Dosen Pembimbing II
FEBY ARTWODINI MUQTADIROH, S.Kom, MT.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017

LEMBAR PENGESAHAN

PEMBANGUNAN SISTEM MONITORING DEBITUR DENGAN MENGGUNAKAN ARSITEKTUR *THREE- TIER* (STUDI KASUS : BANK UMUM DI INDONESIA)

TUGAS AKHIR

Disusun untuk memenuhi salah satu syarat
memperoleh gelar Sarjana Komputer
pada

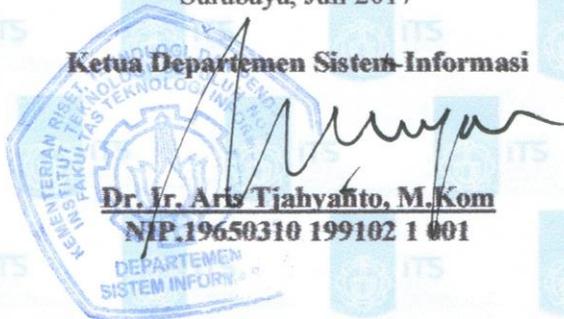
Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

IGEDE ADITYA WIRATAMA
NRP 5212100161

Surabaya, Juli 2017

Ketua Departemen Sistem-Informasi



Dr. Ir. Aris Tjahjanto, M.Kom
NIP.19650310 199102 1 001

LEMBAR PENGESAHAN

PEMBANGUNAN SISTEM MONITORING DEBITUR DENGAN MENGGUNAKAN ARSITEKTUR *THREE- TIER* (STUDI KASUS : BANK UMUM DI INDONESIA)

TUGAS AKHIR

Disusun untuk memenuhi salah satu syarat
memperoleh gelar Sarjana Komputer
pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

I GEDE ADITYA WIRATAMA
NRP 5212100161

Disetujui Tim Penguji: Tanggal Ujian: Juli 2017
Periode Wisuda: September 2017

Bekti Cahyo Hidayanto, S.Si., M.Kom.

(Pembimbing I)

Febby Artwodini Muqtadiroh, S.Kom, MT.

(Pembimbing II)

Dr. Ir. Aris Tjahyanto, M.Kom

(Penguji I)

Arif Wibisono, S.Kom., M.Sc

(Penguji II)

PEMBANGUNAN SISTEM MONITORING DEBITUR DENGAN MENGGUNAKAN ARSITEKTUR THREE- TIER (STUDI KASUS : BANK UMUM DI INDONESIA)

Nama Mahasiswa : I Gede Aditya Wiratama
NRP : 5212100161
Jurusan : Sistem Informasi FTIF – ITS
Dosen Pembimbing I : Bekti Cahyo Hidayanto, S.Si., M.Kom.
Dosen Pembimbing II : Feby Artwodini Muqtadiroh S.Kom, MT.

ABSTRAK

Bank berfungsi sebagai lembaga penyalur antara kelompok masyarakat pemilik dana dengan masyarakat yang membutuhkan dana. Bank menghimpun dana dari masyarakat dalam bentuk tabungan, giro, deposito maupun bentuk simpanan yang lain. Dana yang terkumpul tersebut akan disalurkan dalam bentuk pinjaman atau kredit.

Kredit merupakan penyediaan dana atau tagihan yang berdasarkan pada kesepakatan pinjam-meminjam antara penyedia dana dengan pihak peminjam atau yang membutuhkan dana. Kompensasi yang diberikan oleh pihak peminjam dapat berupa bunga, imbalan atau bagi hasil maupun kesepakatan lain yang disetujui bersama antara pihak bank dan peminjam. Dalam mengambil keputusan terkait kredit yang diajukan calon debitur, bank melakukan analisis dari berbagai data.

Sistem monitoring debitur merupakan sistem yang penulis tawarkan untuk menyimpan data secara terpusat sehingga dapat diakses kapan dan di mana saja dengan mudah. Sistem di rancang dengan menggunakan arsitektur three-tier. Hal ini bertujuan agar tampilan aplikasi dapat dikembangkan dalam bahasa lain, namun tetap memiliki aturan yang baku.

Kata Kunci: *Bank, kredit, kredit wholesale, system monitoring, three-tier architecture.*

Halaman ini sengaja dikosongkan

**DEVELOPMENT OF DEBTOR MONITORING
SYSTEM USING THREE-TIER ARCHITECTURE
(CASE STUDY : COMMERCIAL BANK IN
INDONESIA)**

Name : I Gede Aditya Wiratama
NRP : 5212100161
Department : Sistem Informasi FTIF-ITS
Supervisor : Bekti Cahyo Hidayanto, S.Si., M.Kom.
Supervisor : Feby Artwodini Muqtadiroh S.Kom, MT.

ABSTRACT

The Bank serves as a mediator institution between people who own the fund and people who need the funds. Banks collect funds from the public in the form of savings, demand deposits, deposits and other forms of deposits. Funds collected will be distributed in the form of loans or credit.

Credit is the provision of funds or claims based on credit agreements between the fund provider and the debtor. The compensation provided by the debtor may be in the form of interest, reward or profit sharing or other agreement mutually agreed between the bank and the debtor. In making decisions related to loans proposed by prospective debtor, banks conduct analysis of various data.

Authors offer to store data centrally so that can be accessed anytime and anywhere with ease, by using debtor monitoring system. The system is designed using a three-tier architecture. It aims to application's interface can be developed in other languages, but still using the same rule.

Keyword: Bank, credit, wholesale credits, system monitoring, three-tier architecture.

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur penulis haturkan atas kehadiran Tuhan Yang Maha Esa atas segala rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan laporan tugas akhir dengan judul :

“PEMBANGUNAN SISTEM MONITORING DEBITUR DENGAN MENGGUNAKAN ARSITEKTUR *THREE-TIER* (STUDI KASUS : BANK UMUM DI INDONESIA)

Tugas akhir ini merupakan salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Dengan terselesaikannya tugas akhir ini, maka selesai pula studi penulis untuk meraih gelar sarjana yang telah ditempuh selama lima tahun.

Pada kesempatan ini, penulis menyampaikan ucapan terimakasih kepada pihak-pihak yang telah membantu penulis dalam menyelesaikan tugas akhir ini karena semua ini tidak akan pernah luput dari dukungan dan bantuan mereka. Oleh karena itu, rasa terima kasih ini penulis sampaikan kepada:

- 1) Tuhan Yang Maha Esa yang telah memberikan kesehatan dan kesempatan untuk bisa mengerjakan dan menyelesaikan tugas akhir ini.
- 2) Kedua orang tua penulis. Ibu Rn Endah Listiorini dan Bapak I Gede Eka Yudiasa. Yang telah mengizinkan penulis untuk mengambil kuliah yang sesuai dengan minat penulis, dan telah mempercayai penulis untuk dapat menyelesaikan perkuliahan. Serta kesabaran yang sungguh luar biasa untuk menghadapi sikap dan sifat penulis.
- 3) Kedua adik penulis. Luh Rosita Saraswati dan Luh Chinantya Prabawati. Sebab tanpa doa dan dukungan yang mereka berikan penulis tidak dapat menyelesaikan tugas akhir ini
- 4) Bapak Bakti Cahyo Hidayanto yang berperan sebagai pembimbing dalam pengerjaan tugas akhir ini dan juga

sebagai Kepala Laboratorium Infrastruktur dan Keamanan Teknologi Informasi. Penulis tidak dapat menyelesaikan tugas akhir ini tanpa petunjuk yang beliau berikan terutama dalam pengerjaan sistem yang penulis buat.

- 5) Ibu Feby Artwodini Muqtadiroh yang berperan sebagai pembimbing dalam pengerjaan tugas akhir ini. Penulis tidak dapat menyelesaikan tugas akhir ini tanpa petunjuk yang beliau berikan terutama dalam pengerjaan segala dokumen yang penulis buat.
- 6) Seluruh karyawan Jurusan Sistem Informasi yang telah memberikan pelayanan maksimal demi kenyamanan seluruh civitas akademik.
- 7) I Gusti Ayu Okty Prabawati yang telah memberikan dukungan dan doa agar penulis dapat segera menyelesaikan tugas akhir ini.
- 8) Keluarga IKTI yang telah menemani penulis agar tidak merasa kesepian saat mengerjakan tugas akhir di laboratorium.
- 9) Rekan – rekan dari treeva. Natha, Faisal, Ozy, Arya dan Dio. Yang telah memberikan semangat kepada penulis untuk menyelesaikan tugas akhir ini.
- 10) Estu Rizky H yang telah membantu penulis dalam mengerjakan format penulisan tugas akhir.

Dan berbagai pihak yang tidak dapat penulis sebutkan satu per satu namanya disini. Semoga Tuhan membalas semua kebaikan yang telah dilakukan. Penulis menyadari masih banyak yang dapat dikembangkan pada tugas akhir ini, oleh karena itu penulis menerima segala masukan dan kritik yang diberikan. Semoga tugas akhir ini dapat memberikan manfaat.

Surabaya, Juli 2017

Penulis

DAFTAR ISI

ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	3
1.3. Batasan Tugas Akhir	3
1.4. Tujuan Penelitian	4
1.5. Manfaat Penelitian	4
1.6. Relevansi	5
BAB II TINJAUAN PUSTAKA	7
2.1. Studi Sebelumnya	7
2.2. Dasar Teori	9
2.2.1. Bank	9
2.2.2. Kredit	9
2.2.3. Kredit Wholesale	10
2.2.4. Sistem Informasi	11
2.2.4.1. Definisi Sistem Informasi	11
2.2.4.2. Metodologi Waterfall	12
2.2.4.3. Three-tier architecture	12
2.2.4.4. CodeIgniter	14
2.2.4.5. Perbedaan Three-tier dan MVC	14
2.2.4.6. RESTful Web Services	15
2.2.4.7. JavaScript Object Notation (JSON)	16
2.2.4.8. MySQL	16
2.2.4.9. Gambaran Proses Bisnis	16

BAB VI HASIL DAN PEMBAHASAN.....	47
6.1. Lingkungan Uji Coba.....	47
6.2. Requirements Traceability Matrix.....	48
6.3. Pengujian Fungsional.....	53
6.4. Pengujian Keamanan	57
6.4.1. Pengujian pada Logical Layer.....	57
6.4.2. Pengujian pada Presentation Layer	58
6.5. Pengujian compatibility Browser.....	58
6.5.1. Perangkat Desktop	60
6.5.1.1. Mac OSX 10.8 – Google Chrome 48	60
6.5.1.2. Mac OSX 10.8 – Safari 6.2.....	60
6.5.1.3. Mac OSX 10.8 – Firefox 48.....	61
6.5.2. Perangkat Mobile	61
6.5.2.1. Galaxy Note 3 – Android Browser 4.4	61
6.5.2.2. Galaxy Note 3 – Chrome Mobile 58	62
6.5.2.3. Galaxy Note 3 – Firefox Mobile 53	62
6.5.2.4. iPad 3 / 6.1 – Mobile Safari 6.0	63
6.6. Instalasi Sistem Monitoring Debitur.....	63
6.6.1. Data Layer.....	63
6.6.2. Logical Layer	65
6.6.3. Presentation Layer.....	67
BAB VII KESIMPULAN DAN SARAN	71
7.1. Kesimpulan	71
7.2. Saran	71
DAFTAR PUSTAKA	73
BIODATA PENULIS	75
A. LAMPIRAN A DOMAIN MODEL	A-1
B. LAMPIRAN B USE CASE DIAGRAM	B-1
C. LAMPIRAN C USE CASE DESCRIPTION.....	C-1
D. LAMPIRAN D ROBUSTNESS DIAGRAM	D-1

E.	LAMPIRAN E UPDATED DOMAIN MODEL	E-1
F.	LAMPIRAN F SEQUENCE DIAGRAM.....	F-1
G.	LAMPIRAN G CLASS DIAGRAM.....	G-1

DAFTAR GAMBAR

Gambar 2.1 Metodologi Waterfall	12
Gambar 2.2 Arsitektur yang akan digunakan penulis	13
Gambar 2.3 Proses kredit pada bank	17
Gambar 3.1 Tahap Pelaksanaan Tugas Akhir	19
Gambar 5.1 Alur proses bisnis pada sistem	43
Gambar 5.2 Alur pada Presentation Layer	44
Gambar 5.3 Alur pada Logical Layer.....	45
Gambar 6.1 Hasil pengujian keamanan logical layer.....	57
Gambar 6.2 Hasil pengujian pada presentation layer.....	58
Gambar 6.3 Hasil Pengujian Pada Google Chrome 48	60
Gambar 6.4 Hasil pengujian pada Safari 6.2.....	60
Gambar 6.5 Hasil pengujian pada Firefox 48.....	61
Gambar 6.6 Hasil pengujian pada Android Browser 4.4.....	61
Gambar 6.7 Pengujian Pada Chrome Mobile 58.....	62
Gambar 6.8 Hasil pengujian pada Firefox Mobile 53	62
Gambar 6.9 Hasil pengujian pada Mobile Safari 6.0	63
Gambar 6.10 Buat database baru.....	63
Gambar 6.11 Import Database Simode	64
Gambar 6.12 Struktur database	64
Gambar 6.13 file – file simode-ws	65
Gambar 6.14 ubah config	65
Gambar 6.15 file config simode-ws	65
Gambar 6.16 ubah database	66
Gambar 6.17 pengaturan database.....	66
Gambar 6.18 Upload file simode	67
Gambar 6.19 Ubah file config.....	67
Gambar 6.20 config pada presentation layer.....	68
Gambar 6.21 Halaman login simode.....	68
Gambar 6.22 Tampilan bila berhasil login.....	69

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka (Studi Literatur)	7
Tabel 5.1 Spesifikasi perangkat keras	35
Tabel 5.2 Spesifikasi perangkat lunak.....	35
Tabel 5.3 Daftar API Web Service.....	38
Tabel 6.1 Spesifikasi perangkat keras	47
Tabel 6.2 Spesifikasi perangkat lunak.....	47
Tabel 6.3 Pengujian fungsional	53
Tabel 6.4 Hasil pengujian compatibility browser	59

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Pada bab ini akan dijelaskan mengenai gambaran umum terkait tugas akhir yang berisi latar belakang masalah, rumusan masalah, batasan pengerjaan tugas akhir, tujuan, dan manfaat tugas akhir.

1.1. Latar Belakang Masalah

Berdasarkan Undang – Undang pokok perbankan tahun 1998, bank berfungsi sebagai lembaga penyalur antara kelompok masyarakat pemilik dana dengan masyarakat yang membutuhkan dana. Bank merupakan bisnis yang bersifat kepercayaan dan berkaitan dengan ekonomi negara. Bank harus mampu membuat kelompok masyarakat pemilik dana percaya bahwa dana yang telah di himpun oleh pihak bank dalam bentuk tabungan, deposito maupun simpanan lainnya akan selalu aman. Sehingga, bank memiliki dana untuk dipinjamkan kepada kelompok masyarakat yang membutuhkan dana. [1]

Kredit merupakan penyediaan dana atau tagihan yang berdasarkan pada kesepakatan pinjam-meminjam antara penyedia dana dengan pihak peminjam atau yang membutuhkan dana atas dasar kepercayaan. Dalam hal ini, penyedia dana adalah bank sehingga mewajibkan pihak peminjam dana untuk mengembalikan dana dalam kurun waktu beserta kompensasi yang telah ditentukan. Kompensasi yang diberikan oleh pihak peminjam dapat berupa bunga, imbalan atau bagi hasil maupun kesepakatan lain yang disetujui bersama antara pihak bank dan peminjam. [1]

Pertumbuhan kredit di Indonesia relatif besar bila dibandingkan dengan negara lain di Asia. Pada Tahun 2016 terjadi pertumbuhan kredit mendekati kisaran 9 persen. Dimana hal tersebut sesuai dengan target dari bank sentral yaitu pertumbuhan kredit berkisar antara 7 hingga 9 persen. [2] Dengan pertumbuhan yang cukup besar tersebut, kredit menjadi salah satu aktivitas utama dalam bisnis perbankan. [1]

Terdapat beberapa segmen dalam kredit, salah satunya adalah kredit korporasi. Kredit korporasi diberikan dalam jumlah yang relatif besar dan persyaratan yang kompleks. Hal tersebut menyebabkan muncul kebutuhan untuk melakukan pengawasan yang sangat ketat terhadap calon debitur. [1]

Dalam mengambil keputusan terkait kredit yang diajukan calon debitur bank melakukan analisis. Bank mengumpulkan seluruh data dan informasi terkait calon debitur untuk di analisis. Untuk analisis kredit korporasi, setidaknya terdapat beberapa data yang diperlukan oleh bank yaitu identitas calon debitur, data dan informasi keuangan, daftar agunan atau jaminan dari pihak calon debitur, hasil IDI Bank Indonesia, informasi mengenai grup calon debitur, laporan hasil kunjungan ke perusahaan calon debitur. [1]

Data dan informasi merupakan hal yang sangat penting Karena informasi merupakan aset bagi suatu organisasi atau bisnis. Dalam studi kasus ini data dan informasi yang dimiliki oleh perusahaan adalah data dan informasi milik calon debitur. Data dan informasi tersebut harus dilindungi. Perlindungan terhadap data dan informasi terdiri dari beberapa aspek, yaitu *confidentiality*, *integrity*, dan *availability*. [3]

Saat ini Bank Mandiri masih menggunakan *spreadsheet* sebagai sarana untuk menyimpan data dan informasi debitur. Untuk saling bertukar data dan informasi yang ada staff Bank Mandiri menggunakan media penyimpanan yang dimiliki. Hal ini menyebabkan kesulitan ketika harus mencari dimana posisi data dan informasi terkakhir dari suatu debitur berada. Serta menyebabkan mudahnya data untuk tersebut ke pihak yang tidak berkepentingan. Padahal saat ini Bank Mandiri sudah memiliki server yang terpusat yang terhubung dengan jaringan internal di seluruh gedung milik Bank Mandiri yang terdapat di Indonesia.

Dari apa yang telah penulis sebutkan sebelumnya, dapat disimpulkan terdapat beberapa masalah yang dialami pihak Bank Mandiri dalam proses memberikan kredit kepada segment

korporasi. Masalah pertama adalah, bagaimana setiap bagian dari Bank Mandiri yang berkaitan dengan pengambilan keputusan kredit bisa mendapatkan data dan informasi terakhir terkait debitur dalam waktu segera. Masalah kedua adalah bagaimana agar data dan informasi terkait calon debitur Bank Mandiri tidak tersebar kepada pihak yang tidak memiliki kepentingan. Dan masalah yang terakhir adalah bagaimana melakukan pengasawan terhadap debitur dengan mudah dan cepat.

Untuk menyelesaikan permasalahan yang telah disebutkan, penulis memiliki suatu gagasan untuk membuat sistem berbasis website. Sistem yang ingin penulis buat merupakan sistem yang akan membantu pihak Bank Mandiri untuk menyimpan semua data dan informasi terkait calon debitur yang ada dalam satu tempat sehingga data dan informasi terakhir dapat di akses dengan segera dan dari manapun dengan aman. Dan untuk dapat memanfaatkan server milik Bank Mandiri, maka penulis harus menyesuaikan dengan arsitektur server yang ada, yaitu arsitektur tiga tingkat atau *three-tier architecture*.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, adapun rumusan masalah yang dapat diuraikan dalam tugas akhir ini adalah sebagai berikut:

1. Apa saja kebutuhan pengguna dalam berbagi data dan informasi terkait debitur, sehingga proses pengawasan debitur dapat dilakukan dengan lebih mudah dan cepat?
2. Bagaimana agar aplikasi dapat diterapkan pada jaringan yang saat ini tersedia?

1.3. Batasan Tugas Akhir

Beberapa batasan yang diberlakukan dalam pengerjaan tugas akhir ini adalah :

1. Sistem dibuat berdasarkan pada proses yang saat ini ada di Bank Mandiri Regional VIII.

2. Sistem hanya berfungsi sebagai alat untuk membantu mengambil keputusan bukan sebagai alat untuk pengambil keputusan.
3. Sistem yang dibuat hanya akan berbasiskan website.
4. Pengguna harus berada di jaringan yang terhubung dengan server.

1.4. Tujuan Penelitian

Tujuan pengerjaan tugas akhir ini adalah membuat aplikasi yang dapat membantu pihak Bank Mandiri Regional VIII dalam melakukan *monitoring* debitur. Adapun aplikasi tersebut memiliki beberapa fitur yakni sebagai berikut:

1. Melakukan pemisahan data pada tiap CMB.
2. Menyimpan data debitur, yang terdiri dari rincian debitur, *ots*, *financial ratio*, dan *covenant*.
3. Melakukan perhitungan pada *financial ratio*.

1.5. Manfaat Penelitian

Tugas akhir ini diharapkan dapat memberikan manfaat antara lain adalah sebagai berikut :

1. Bagi Bank Mandiri Regional VIII hasil penelitian ini akan dapat digunakan sebagai solusi dalam beberapa masalah yang terdapat dalam penyimpanan data debitur dan pengambilan keputusan terkait debitur.
2. Bagi akademisi hasil penelitian ini dapat memberikan manfaat sebagai salah satu acuan dan informasi tambahan serta perbandingan bagi peneliti lain, khususnya yang melakukan penelitian terhadap permasalahan yang serupa.
3. Bagi penulis penelitian ini dapat meningkatkan wawasan penulis terkait sistem monitoring debitur, dan dalam pembuatan aplikasi berbasiskan *three-tier architecture*.

1.6. Relevansi

Tugas akhir ini memiliki keterkaitan atau relevansi dengan ketentuan topik tugas akhir pada Jurusan Informasi, yakni sebagai berikut:

- Topik pada tugas akhir ini adalah terkait implementasi aplikasi monitoring debitur dengan menggunakan *three-tier architecture* sehingga masih berkaitan dengan topik yang berada pada Laboratorium Infrastruktur dan Keamanan Teknologi Informasi.
- Tugas akhir ini layak dijadikan sebagai tugas akhir pada tingkat S1, karena tugas akhir ini dapat memberikan solusi dalam pemecahan masalah terkait pengawasan data debitur yang digunakan sebagai dasar dalam pengambilan keputusan.
- Tugas akhir ini berkaitan dengan beberapa mata kuliah seperti Desain Manajemen Jaringan dan Keamanan Aset Informasi sehingga layak untuk dijadikan topik sebagai tugas akhir pada Departemen Sistem Informasi

Halaman ini sengaja dikosongkan

BAB II TINJAUAN PUSTAKA

Pada bab ini dijelaskan mengenai teori-teori terkait yang bersumber dari buku, jurnal, ataupun artikel yang berfungsi sebagai dasar dalam melakukan pengerjaan tugas akhir agar dapat memahami konsep atau teori penyelesaian permasalahan yang ada.

2.1. Studi Sebelumnya

Penelitian tugas akhir ini menggunakan metode yang dikembangkan pada penelitian sebelumnya. Sementara untuk obyek penelitian dan tujuan penelitian, hasil studi penelitian sebelumnya telah dijelaskan pada bab pendahuluan.

Berikut akan dipaparkan beberapa penelitian sebelumnya terkait dengan permasalahan yang diangkat dalam pengerjaan tugas akhir. Beberapa metode dari penelitian sebelumnya akan dijadikan acuan serta pendukung dalam pengerjaan tugas akhir ini.

Tabel 2.1 Tinjauan Pustaka (Studi Literatur)

No	Judul, Penulis, Tahun	Metodologi	Keterkaitan dengan Tugas Akhir
1.	Perancangan dan Pembuatan Sistem Informasi Manajemen Keuangan Sub Bagian Perbendaharaan, Studi Kasus Pemerintahan Kabupaten Malang [4]	Pengembangan sistem informasi dengan desain sistem three-tier	Sistem Informasi Sub Bagian Perbendaharaan dengan sistem <i>three-tier</i>

	(Bilqis Amaliah, Khakim Ghozali, Tri Agung Wahyu Handrian:JUTI)		
2.	<p>Pengembangan Sistem Informasi Perparkiran Menggunakan Arsitektur Three-tier [5]</p> <p>(Yuli Fauziah, Dessyanto Boedi Prasetyo:Seminar Nasional Informatika 2015)</p>	Penerapan three-tier pada pembuatan sistem informasi	Sistem informasi yang efisien, yang dapat mempermudah aktivitas perusahaan
3.	<p>Implementasi Model Waterfall pada Pengembangan Sistem Informasi Alumni SMKN 1 Jenangan Ponorogo [6]</p> <p>(Eka Nur Ahmad Romadhoni, Triyanna Widiyaningtyas, Utomo Pujiyanto:Seminar Nasional Informatika 2015)</p>	Melakukan pengembangan Sistem Informasi dengan model Waterfall	<p>Model waterfall dapat membantu pengembangan sistem informasi alumni</p> <p>Dengan sistem waterfall kebutuhan dapat terdefinisi secara jelas</p>

2.2. Dasar Teori

Dasar teori menjelaskan konsep-konsep atau teori yang sekiranya dibutuhkan untuk lebih memahami Tugas Akhir ini.

2.2.1. Bank

Bank merupakan suatu lembaga keuangan yang berfungsi sebagai lembaga penyalur antara kelompok masyarakat pemilik dana dengan masyarakat yang membutuhkan dana atau dapat disebut sebagai *financial intermediary*. Pemilik dana menyimpan dana yang dimiliki di bank dalam bentuk tabungan, deposito maupun simpanan lainnya. Dana yang telah di kumpulkan tersebut disalurkan kepada masyarakat yang membutuhkan dana dalam bentuk kredit. Hal ini sesuai dengan Undang - Undang No. 10 tahun 1998, dimana bank adalah badan usaha yang menghimpun dana dari masyarakat dalam bentuk simpanan dan menyalurkan kepada masyarakat dalam bentuk kredit dan/atau bentuk - bentuk lainnya dalam rangka meningkatkan taraf hidup rakyat. [7]

2.2.2. Kredit

Kredit merupakan salah satu kegiatan utama bank. Kredit merupakan penyediaan dana bagi pihak yang membutuhkan dana atas persetujuan Bersama antara bank dan pihak peminjam dana. Bank melakukan aktivitas kredit dengan tujuan untuk menguntungkan kedua belah pihak, yaitu pihak bank dan pihak debitur. Karena kredit merupakan salah satu kegiatan utama bank, maka bank berusaha untuk melakukan ekspansi kredit yaitu dengan menjual berbagai jenis produk kredit. [1]

Dalam melakukan penyaluran dana terdapat tiga jenis produk kredit yang ditawarkan oleh bank, yaitu kredit konsumen, kredit ritel dan kredit wholesale. Kredit consumer merupakan kredit yang diberikan pada pembelian yang bersifat konsumtif, seperti rumah, kendaraan, dan lain-lain. Kredit ritel adalah kredit yang diberikan kepada perorangan maupun badan

usaha dan digunakan untuk kegiatan usaha. Kredit wholesale memiliki kesamaan dengan kredit ritel, yaitu digunakan untuk kegiatan usaha. Tetapi, kredit wholesale memiliki ukuran pinjaman yang lebih besar, serta hanya diberikan pada usaha berbadan hokum. [1]

2.2.3. Kredit Wholesale

Kredit wholesale dapat juga disebut sebagai kredit segmen korporasi. Kredit pada segmen korporasi memiliki karakteristik dan struktur pembiayaan yang kompleks. Pada umumnya kredit pada segment ini berdasarkan pada arus kas dan fasilitas atau jaminan sesuai kebutuhan bisnis masing-masing calon debitur. [1]

Kredit yang diberikan kepada debitur pada segmen korporasi memiliki jumlah yang besar. Karena jumlah kredit yang besar tersebut, bank melakukan pengawasan yang sangat ketat terhadap calon debitur. Selain pengawasan yang ketat, bank juga membutuhkan waktu yang cukup lama daripada memproses kredit yang diberikan kepada reetail maupun konsumen. [1]

Adapun data yang diawasi oleh pihak bank terdiri dari empat kelompok data, yaitu data debitur, data OTS (On The Spot), data *financial ratio*, data *covenant*. Data debitur merupakan kumpulan data terkait profil perusahaan, seperti nama perusahaan, grup debitur, dan lainnya. OTS adalah data yang di dapat saat pihak bank melakukan verifikasi atau datang langsung ke perusahaan calon debitur untuk mendapatkan kondisi sebenarnya dari perusahaan. *Financial ratio* adalah data keuangan pihak calon debitur yang diambil pada waktu tertentu, bisa diambil tiap bulan, tiap tiga bulan, tiap enam bulan, atau tahunan sesuai dengan kebutuhan pihak bank. *Covenant* merupakan persyaratan yang ditentukan oleh bank dan disetujui calon debitur untuk melakukan dan/atau tidak melakukan tindakan tertentu.

Pada *financial ratio* terdapat perhitungan beberapa data, yaitu nilai kecukupan, nilai *current assets*, nilai *current liabilities*, nilai *current ratio* dan nilai *Debt service coverage*.

Berikut ini adalah rumus untuk menghitung nilai – nilai tersebut:

$$\text{Nilai Kecukupan} = \frac{\text{stok/persediaan} + \text{piutang} + \text{kas}}{\text{kredit}}$$

$$\begin{aligned} \text{Nilai CA} = & \text{Cash} + \text{Trade Receivables} \\ & + \text{Other Receivables} + \text{Inventory} \\ & + \text{Prepaid Expenses} + \text{Prepaid Taxes} \\ & + \text{Sundry CA} \end{aligned}$$

$$\begin{aligned} \text{Nilai CL} = & \text{STD} - \text{Bank (existing)} + \text{Current Portion} \\ & + \text{Trade payables} + \text{Accrued Expenses} \\ & + \text{Tax Payables} + \text{Deferred Income} \\ & + \text{Sundry CL} \end{aligned}$$

$$\text{Nilai CR} = \frac{\text{Nilai CA}}{\text{Nilai CL}} \times 100\%$$

$$\text{Nilai NPM} = \frac{\text{laba}}{\text{sales}} \times 100\%$$

$$\text{Debt service coverage} = \frac{\text{ebitda}}{\text{angsuran} + \text{bunga}} \times 100\%$$

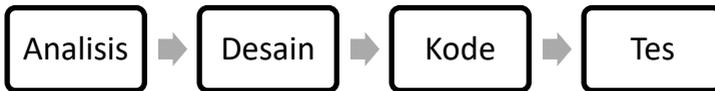
2.2.4. Sistem Informasi

2.2.4.1. Definisi Sistem Informasi

Sistem informasi adalah sekumpulan manusia dan mesin yang terpadu dengan tujuan untuk menyajikan informasi yang mendukung fungsi operasi, manajemen, dan pengambilan keputusan dalam sebuah organisasi. Informasi adalah data yang telah diolah menjadi bentuk yang berguna dan memiliki nilai bagi penerimanya [5].

2.2.4.2. Metodologi Waterfall

Model waterfall, dikenal juga dengan model sekuensial linear, mengusulkan pendekatan kepada perkembangan perangkat lunak sistematis dan sekuensial. Model waterfall dimulai pada tingkat kemajuan sistem pada analisis, desain, kode, pengujian, model sekuensial linier. [6] Tahapan pengembangan sistem pada model waterfall diilustrasikan pada Gambar 2.1.



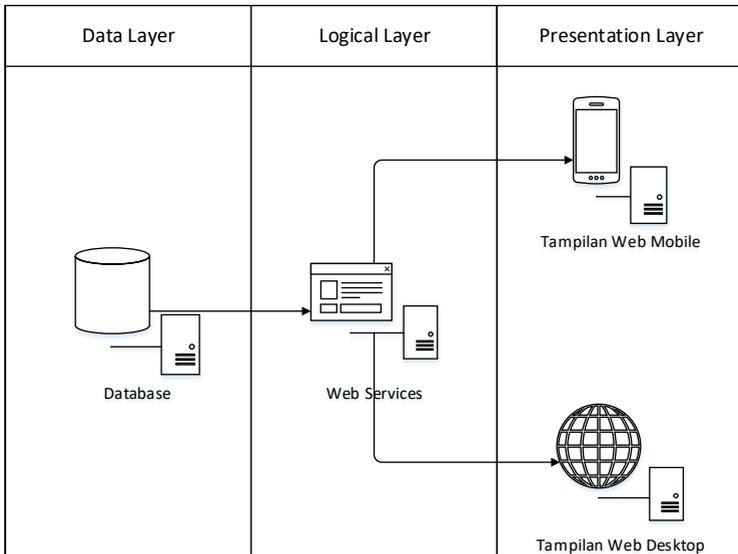
Gambar 2.1 Metodologi Waterfall

2.2.4.3. Three-tier architecture

Arsitektur Three Tier merupakan pengembangan dari arsitektur Client Server. Arsitektur Client Server adalah arsitektur jaringan yang memisahkan client (umumnya merupakan GUI) dengan server. Server dapat memberikan data atau informasi sesuai yang diminta oleh client. [4] Penulis memilih menggunakan *three-tier architecture* dikarenakan arsitektur yang saat ini ada dan digunakan oleh pihak bank dimana penulis mengerjakan tugas akhir adalah *three-tier architecture*.

Arsitektur three tier terdiri dari 3 bagian, yaitu:

1. Presentation layer
Layer pertama ini disebut user interface. Presentation layer merupakan layer yang berinteraksi secara langsung, memberikan informasi yang dapat dibaca dan dipahami dengan mudah oleh pengguna.
2. Logical layer
Logical layer merupakan penghubung dari 2 layer lainnya. Logical layer membuat keputusan logic, dan memproses data yang melaluinya.
3. Data layer
Merupakan layar di mana data tersimpan. Informasi yang diminta dari presentation layer akan diambil dari data layer.



Gambar 2.2 Arsitektur yang akan digunakan penulis

Gambar 2.2 merupakan penggambaran dari arsitektur three-tier yang akan digunakan oleh penulis. Adapun penjelasan terkait rancangan three-tier yang akan dibuat, antara lain:

- **Data layer**
 Dalam sistem informasi yang akan penulis buat, pada data layer akan terdapat database yang menyimpan seluruh data yang di masukkan. Pada layer ini penulis akan menggunakan MySQL sebagai database.
- **Logical layer**
 Dalam sistem informasi yang akan penulis buat, logical layer akan berperan sebagai penghubung antara presentation layer dan data layer. Presentation layer tidak boleh terhubung secara langsung ke data layer, harus melalui logical layer terlebih dahulu. Pada logical layer, penulis akan menggunakan JSON sebagai bahasa pemrograman *web-service* dan RESTful sebagai framework atau standar penulisan.

- Presentation layer
Dalam sistem yang akan penulis buat, presentation layer akan berisi tampilan yang dapat dilihat oleh pengguna sistem. Terdapat dua jenis tampilan, yaitu tampilan dalam versi *web desktop* dan *mobile desktop*.

2.2.4.4. CodeIgniter

CodeIgniter adalah framework PHP yang dibuat oleh Rick Ellis pada tahun 2006. CodeIgniter dibuat sebagai alat yang simpel dan elegan. CodeIgniter memungkinkan pengembangan baik di situs web maupun aplikasi web [8].

CodeIgniter memiliki beberapa kelebihan dibandingkan dengan framework lain, yaitu:

- Berukuran kecil
Dengan ukuran yang kecil akan mempermudah dan mempercepat proses mengunggah aplikasi website. Selain itu dengan ukuran yang kecil akan meminimalkan kebutuhan ruang penyimpanan (*hosting*).
- Memiliki dokumentasi yang lengkap
Dengan dokumentasi yang lengkap akan mempermudah untuk melakukan pembangunan website dengan memanfaatkan fitur – fitur yang terdapat pada CodeIgniter.
- Kompatibel dengan standar hosting
Dengan sistem yang sudah kompatibel dengan standar *hosting* yang ada, akan mempermudah untuk mencari *hosting* yang sesuai.
- Memiliki performa yang baik
Dengan performa yang baik akan mempermudah dan memberikan kepuasan bagi pengguna.

2.2.4.5. Perbedaan *Three-tier* dan MVC

Arsitektur *three-tier* memiliki kemiripan dengan MVC yaitu view yang tidak terhubung secara langsung dengan database, melainkan harus melalui Logical layer terlebih dahulu pada arsitektur *three-tier* dan melalui model terlebih dahulu

pada MVC. Baik arsitektur three-tier maupun MVC memiliki satu tujuan yaitu untuk meningkatkan keamanan website.

Perbedaan yang paling terlihat dari arsitektur three-tier dengan MVC adalah kemudahan dalam melakukan pengembangan lebih lanjut. Pengembangan lebih lanjut yang penulis maksud adalah terkait penggunaan logical layer pada three-tier dan model pada MVC. Pada arsitektur three-tier, presentation layer dapat menggunakan berbagai bahasa pemrograman asalkan dapat menerima format data yang diberikan oleh logical layer, sehingga dalam melakukan pengembangan dengan bahasa pemrograman yang berbeda, seperti android, hanya perlu memikirkan bagaimana menerima dan menampilkan data. Sedangkan pada MVC, dalam pengembangan perangkat lunak harus menggunakan framework yang sama, sehingga apabila ingin melakukan pengembangan dalam bahasa pemrograman yang berbeda perlu dibuatkan penghubung database yang baru juga.

Dalam studi kasus ini penulis memilih untuk menggunakan arsitektur three-tier sebagai arsitektur dalam pembuatan sistem informasi. Hal ini bertujuan untuk mempermudah pengembangan sistem informasi, misalnya untuk membuat versi android dan iOS. Tetapi penulis juga menggunakan view dan controller dari model MVC hal ini bertujuan untuk mempermudah dalam pengerjaan tugas akhir, dan menjaga kerapian dari sistem informasi yang akan penulis buat.

2.2.4.6. RESTful Web Services

Representational State Transfer atau lebih dikenal dengan Rest merupakan arsitektur yang dapat menentukan tiap keadaan dengan spesifik. RESTful web services dibangun untuk bekerja dengan baik di web. Dalam gaya penulisan REST, data dan fungsi dianggap sumber daya dan diakses menggunakan Uniform Resource Identifier (URI) [9].

Terdapat beberapa prinsip agar RESTful dapat berjalan dengan mudah, ringan dan cepat [9]:

- Identifikasi sumber daya melalui URI

- Antarmuka yang seragam
- Pesan yang mendeskripsikan diri sendiri
- Interaksi melalui *hyperlink*

2.2.4.7. JavaScript Object Notation (JSON)

JavaScript Object Notation atau yang lebih dikenal dengan JSON merupakan format pertukaran data yang ringan. JSON memiliki kemudahan untuk di tulis maupun dibaca oleh manusia, dan cukup mudah untuk di terjemahkan dan dibuat oleh mesin. JSON merupakan bahasa yang berbasiskan pada bahasa pemrograman JavaScript. JSON adalah sebuah format penulisan bahasa yang sepenuhnya berdiri sendiri, sehingga menjadikan JSON sebagai sebuah pertukaran data yang ideal [10].

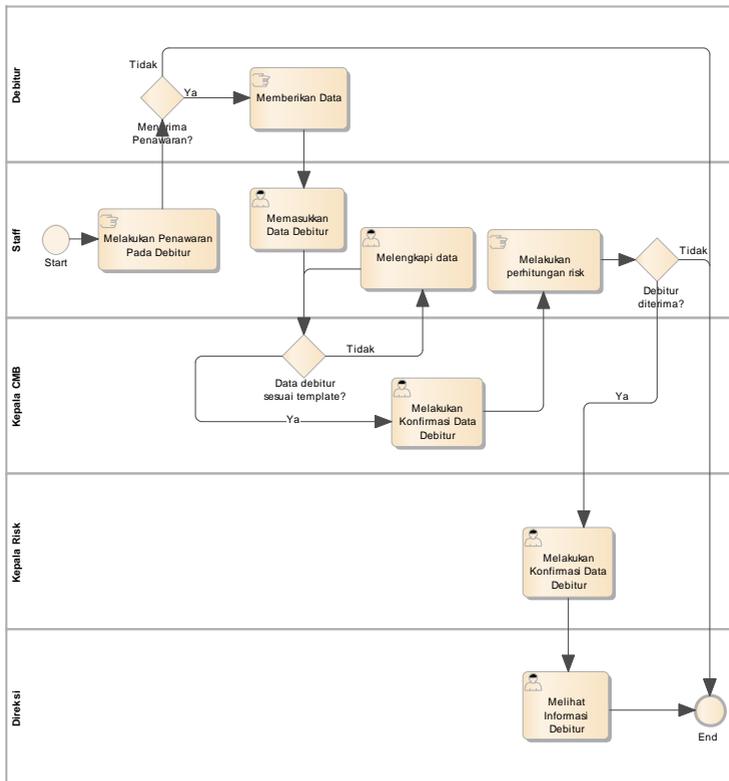
JSON terdiri dari dua struktur. Struktur pertama adalah pasangan nama/nilai, dalam bahasa pemrograman lain dapat dianggap sebagai objek. Struktur kedua adalah daftar nilai terurutkan, dalam bahasa pemrograman lain direalisasikan sebagai array, daftar, atau urutan [10].

2.2.4.8. MySQL

MySQL adalah database *open source* yang paling banyak digunakan saat tugas akhir ini dibuat, dengan nilai 1347,71 [11]. MySQL mampu untuk memberikan kinerja yang tinggi dalam skalabilitas yang besar dengan biaya yang efektif. MySQL Community Edition merupakan edisi yang bisa di dapatkan dengan gratis [12].

2.2.4.9. Gambaran Proses Bisnis

Dalam sistem yang akan dibuat terdapat beberapa pengguna, yaitu pihak staff, kepala CMB, kepala WCR, dan direksi. Terdapat bpmn (*business process modeling and annotation*) yang menggambarkan proses dari tiap pengguna. Pada Gambar 2.3 Proses kredit pada bank merupakan gambaran dari pemrosesan kredit.



Gambar 2.3 Proses kredit pada bank

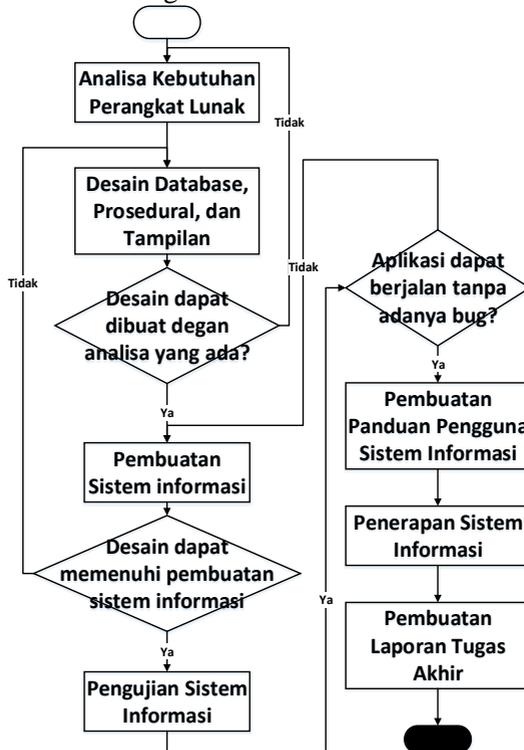
Halaman ini sengaja dikosongkan

BAB III METODOLOGI PENELITIAN

Metodologi penelitian ini merupakan sebuah panduan dalam mengerjakan tugas akhir agar dapat diselesaikan secara sistematis, dan terarah.

3.1. Tahapan Pelaksanaan Tugas Akhir

Dalam melaksanakan Tugas Akhir ini terdapat beberapa tahap yang akan dilakukan oleh penulis. Adapun tahapan dari tugas akhir ini akan diilustrasikan pada Gambar 3.1 Tahap Pelaksanaan Tugas Akhir.



Gambar 3.1 Tahap Pelaksanaan Tugas Akhir

3.1.1. Analisis Kebutuhan Perangkat Lunak

Proses analisa kebutuhan perangkat lunak. Untuk memahami perangkat lunak yang akan dibuat, penulis perlu mempelajari proses bisnis yang ada pada organisasi, beserta data dan informasi yang terdapat pada tiap proses bisnis.

Dalam melakukan proses analisis, penulis akan melakukan pembelajaran di RRM Bank Mandiri Regional VIII untuk mengetahui apa saja kebutuhan - kebutuhan RRM Bank Mandiri Regional VIII dalam pembuatan sistem.

3.1.2. Desain Database, Prosedural, dan Tampilan

Desain Database, Prosedural, dan Tampilan, sebenarnya merupakan proses yang terdiri dari beberapa proses. Proses yang terdapat dalam analisa perangkat lunak adalah desain struktur data atau pembuatan desain database, desain prosedural, dan desain tampilan perangkat lunak. Adapun penjelasan untuk tiap proses yang ada, antara lain:

3.1.3. Pembuatan Sistem informasi

Desain yang sebelumnya telah dibuat akan diterjemahkan ke dalam bentuk bahasa pemrograman yang dapat dimengerti oleh mesin. Pada proses ini sistem akan dibuat berdasarkan dengan desain – desain yang sebelumnya telah dibuat. Proses ini akan berjalan apabila proses desain telah selesai.

3.1.4. Pengujian Sistem Informasi

Pengujian sistem informasi terdiri dari beberapa proses. Proses yang terdapat pada pengujian perangkat lunak adalah pengujian keselarasan antara kode dengan desain yang telah dibuat, pengujian bug dan aplikasi diterima oleh pihak RRM Bank Mandiri Regional VIII.

Adapun penjelasan untuk tiap proses yang ada, antara lain:

- Pengujian keselarasan kode dengan desain

Penulis akan melakukan pengecekan antara desain yang sebelumnya telah di buat dengan realita pada sistem informasi yang penulis buat. Perbedaan yang penulis periksa contohnya data yang dikirim dan ditampilkan pada tiap proses, dan prosedur dari tiap proses.

- Pengujian bug
Penulis akan melakukan pengujian pada tiap fungsi yang terdapat dalam sistem informasi untuk memastikan tidak ada bug yang berpotensi menjadi celah keamanan ataupun ketidakakuratan pada informasi yang ada.
- Pengujian penerimaan oleh pihak RRM Bank Mandiri Regional VIII
Dalam melakukan tes sistem informasi, akan dilakukan pengecekan pada tiap fungsi yang ada dengan cara melakukan demonstrasi terhadap setiap fungsi yang ada, dan pihak RRM Bank Mandiri Regional VIII yang akan melakukan penilaian apakah sudah sesuai dengan apa yang dibutuhkan.

3.1.5. Pembuatan Panduan Pengguna Sistem Informasi

Pada tahap ini, penulis akan membuat buku panduan penggunaan sistem informasi agar pengguna dapat dengan mudah menggunakan setiap fungsi yang ada pada sistem informasi. Penulis berharap dengan ketersediaan buku panduan penggunaan sistem informasi, pengguna akan dapat dengan mudah menggunakan sistem informasi sehingga mempermudah dan mempercepat proses bisnis yang ada.

3.1.6. Penerapan Sistem Informasi

Penulis juga akan membantu pihak organisasi untuk mengimplementasikan sistem informasi yang akan penulis buat sehingga pihak RRM Bank Mandiri Regional VIII dapat segera memanfaatkan sistem informasi yang telah penulis buat.

3.1.7. Pembuatan Laporan Tugas Akhir

Pada tahap ini penulis akan melakukan pembuatan laporan tugas akhir sesuai dengan apa yang telah dikerjakan oleh penulis. Laporan akan berisi hasil analisa kebutuhan perangkat lunak, desain *database*, desain prosedur, desain GUI, kode dari Sistem Informasi, dan penilaian kesesuaian sistem informasi dari pihak RRM Bank Mandiri Regional VIII.

Halaman ini sengaja dikosongkan

BAB IV PERANCANGAN

Pada bab ini akan dijelaskan bagaimana rancangan bangun dari penelitian tugas akhir berdasarkan Iconix Agile Developmenn.

4.1. Kebutuhan Fungsional

Pada proses ini penulis akan melakukan Analisa terkait apa saja kemampuan yang dimiliki oleh system. Kemampuan yang dimaksud adalah layanan yang akan diberikan oleh sistem guna memenuhi kebutuhan pengguna. Kebutuhan fungsional sistem akan ditulis dalam suatu kode unik beserta deskripsi singkat terkait kebutuhan tersebut.

Berikut Kebutuhan Fungsional Sistem:

FR-101 Setiap pengguna harus memiliki akun yang diamankan dengan *username* dan *password*.

Keterangan :

Data di dalam sistem bersifat rahasia, tidak ada data yang ditampilkan untuk *guest*.

FR-102 Sistem hanya menampilkan data sesuai dengan level pengguna saat masuk ke dalam sistem

Keterangan :

Pada masing-masing pegawai terdapat wewenang yang berbeda-beda sehingga mempengaruhi data yang disajikan oleh sistem. Misalnya: untuk direksi memiliki kewenangan dalam melihat seluruh data debitur atau calon debitur diseluruh cabang. Sedangkan kepala, hanya berwenang dalam melihat data hanya dicabang didalam naungannya saja.

FR-201 Sistem dapat menampilkan rincian informasi mengenai debitur, yakni: data profil debitur, data OTS, data *covenant* data *financial ratio*

Keterangan :

Dalam proses pemutusan kredit, terdapat informasi-informasi yang sangat diperlukan. Informasi tersebut berasal dari beberapa kelompok data. Data profil debitur sebagai data induk yang berisi informasi garis besar terkait debitur atau calon debitur. Data OTS terkait hasil dari kunjungan ke debitur atau calon debitur. Sedangkan data *covenant* merupakan semua dokumen atau data yang diminta kepada debitur atau calon debitur. Data *financial ratio* berisi terkait data finansial dari debitur atau calon debitur.

FR-202 Sistem menyediakan fungsi untuk menambah dan mengubah informasi debitur, yakni: data debitur, data OTS, data *covenant* data *financial ratio*

Keterangan:

setiap bulan pasti terdapat penambahan debitur baru dan setiap kali kunjungan ke debitur atau calon debitur, pasti akan mengubah beberapa data untuk diperbarui (misalnya posisi kol, limit dan bade)

FR-203 Sistem menyediakan fungsi menyetujui dan/atau konfirmasi untuk data terkait debitur yang baru ditambahkan dan/atau diubah.

Keterangan :

Kepala WCR dan CMB memiliki kewenangan dan bertanggungjawab atas kebenaran (valid)

data debitur atau calon debitur sehingga perlu adanya konfirmasi data sebelum ditunjukkan kepada manajemen atau jajaran direksi.

FR-301 Sistem dapat melakukan kalkulasi/perhitungan secara tepat pada *financial ratio*

Keterangan :

Financial ratio menunjukkan posisi keuangan dari debitur atau calon debitur. Dalam proses penghitungan financial ratio terdapat beberapa data keuangan perusahaan yang digunakan sebagai bahan hitungan dan dengan rumus tertentu menghasilkan sebuah nilai yang dapat digunakan untuk memprediksi apakah debitur atau calon debitur tersebut dapat membayar kewajibannya atau tidak.

FR-401 Sistem dapat menampilkan daftar *covenant* dengan syarat-syarat sebagai berikut:

- *Covenant* belum terpenuhi baik yang sudah melewati batas penyerahan (jatuh tempo) maupun yang tidak memiliki batas penyerahan
- *Covenant* yang belum selesai/belum dikonfirmasi dan belum disetujui oleh kepala WCR
- *Covenant* yang sudah selesai/sudah dikonfirmasi tetapi belum disetujui oleh kepala WCR

Keterangan :

Covenant merupakan dokumen yang harus diserahkan kepada pihak bank pada periode atau waktu tertentu sehingga pihak bank harus tahu

daftar covenant yang harus diminta kepada debitur atau calon debitur

FR-501 Data di dalam sistem dapat dibuatkan *back-up* untuk mencegah kehilangan data

Keterangan :

Data yang terdapat di dalam sistem merupakan data yang digunakan dalam rentang waktu yang panjang. Adapun data yang digunakan dalam rentang waktu yang panjang tersebut antara lain data financial ratio, yang menunjukkan kondisi keuangan debitur dari waktu ke waktu, dan data covenant yang menunjukkan kewajiban – kewajiban debitur yang harus dipenuhi dalam waktu dekat maupun tanpa batas waktu.

4.2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang dapat mendukung dalam berjalannya sebuah sistem sehingga dapat digunakan oleh pengguna.

Berikut Kebutuhan non-fungsional system:

NFR-101 Sistem dapat diakses oleh 300 pengguna aktif secara bersamaan

Rinciang pengguna:

- Admin = 1 orang

- RM (basra= 60 orang, pemuda= 40 orang, jembatan merah= 60 orang)

- CRM (basra= 40 orang, pemuda= 35 orang, jembatan merah= 55 orang)

- Kepala CMB (1 kepala cmb= 1 orang, ada 3 cmb yakni basra, pemuda, jembatan merah sehingga = 3 orang)

- Kepala WCR (1 kepala wcr= 1 orang, ada 3 wcr yakni basra, pemuda, jembatan merah sehingga = 3 orang)

- RRH= 1 orang

- RCEO= 1 orang

- Direksi= 1 orang

NFR-102 Sistem dapat digunakan selama 7x24 jam dalam seminggu.

Keterangan:

sistem harus selalu available Karena pihak manajemen tidak selalu menunggu jam dan hari kerja untuk melakukan akses ke sistem. Beberapa kasus yang sering terjadi yang mengakibatkan manajemen mengakses sistem diluar jam dan hari kerja, antara lain :

- adanya permintaan dari dewan komisaris untuk mengirimkan data terbaru secara tiba-tiba

- adanya isu/permasalahan/kasus yang terjadi atau yang dapat menyangkut kepada debitur/calon debitur sehingga perlu mengubah beberapa data untuk mempengaruhi pengambilan keputusan kredit

- jatuh tempo penagihan pembayaran kepada debitur yang diluar hari kerja (baik Karena

permintaan debitur atau permintaan manajemen) sehingga manajemen perlu mengakses sistem terlebih dahulu untuk melihat data/record

- adanya permintaan manajemen untuk melakukan kunjungan ke debitur/calon debitur yang bermasalah secara mendadak dan diluar jam kerja sehingga perlu akses ke sistem untuk menambah data OTS nya

NFR-201 Data yang terdapat di dalam sistem harus dipastikan aman.

Keterangan :

Data yang disimpan merupakan data milik berbagai perusahaan sehingga apabila data yang ada harus dijaga dengan sangat ketat agar tidak bocor.

NFR-202 Sistem secara otomatis akan *sign out* ketika tidak digunakan selama 30 menit

Keterangan :

untuk menghindari akses sistem oleh orang yang tidak berwenang. Biasanya pegawai meninggalkan sebentar komputer mereka saat jam kerja dan dalam keadaan login sistem. Sehingga untuk menghindari tersebut, dibuat sign out otomatis

NFR-301 Sistem dapat diakses di mana pun dan kapan pun selama terhubung dengan server dimana sistem terinstall

Keterangan :

para manajemen (direksi, kepala) lebih suka membuka sistem menggunakan smartphone

mereka. Selain itu mereka sering berpergian dan sering mengakses sistem untuk memantau debitur atau calon debitur.

NFR-302 Sistem dapat diakses dengan menggunakan berbagai browser, baik itu pada perangkat *desktop* maupun *mobile*.

Keterangan :

Sistem digunakan bukan hanya oleh pengguna yang berada di area kantor, tetapi juga saat pengguna melakukan aktivitas lapangan seperti OTS. Saat di kantor pengguna menggunakan perangkat computer yang telah tersedia, tetapi saat OTS pengguna menggunakan perangkat *mobile*.

4.3. Domain Model

Domain Model merupakan representasi dari objek – objek nyata dan relasi antara objek dalam suatu permodelan. Domain model didapat dari kebutuhan kebutuhan yang sebelumnya sudah didefinisikan. Adapun domain model dalam tugas akhir ini terdapat dalam Lampiran A.

4.4. Use Case

Use case merupakan gambaran tentang aktivitas apa saja yang terdapat dalam sistem beserta siapa saja yang dapat menggunakan use case tersebut. Use case juga menggambarkan keterkaitan antara satu aktivitas dengan aktivitas yang lain. Selain itu juga terdapat rincian proses dari tiap use case beserta keadaan alternatif ketika keadaan umum use case tersebut tidak terpenuhi.

4.5. Use Case Diagram

Use case diagram merupakan suatu diagram yang menggambarkan Aktivitas apa saja yang terdapat dalam sistem dan relasi antara satu aktivitas dan aktivitas yang lain. Use case

diagram juga memberikan gambaran siapa saja yang dapat dan boleh melakukan suatu aktivitas. Adapun *Use Case Diagram* dalam tugas akhir ini terdapat dalam Lampiran C.

4.6. Use Case Description

Use Case Description merupakan tabel yang berisi rincian dari suatu use case. Rincian use case tersebut antara lain deskripsi singkat tujuan use case, prioritas use case, aktor yang menjalankan use case, keadaan yang harus dipenuhi sebelum menjalankan use case, skenario use case, dan skenario alternative use case. Adapun contoh table use case description terdapat dalam tabel.

Summary	: Pengguna masuk ke dalam sistem
Priority	: <Essential/Expected/Optional>
Use Frequency	: <Seberapa sering suatu use case akan dijalankan>
Direct Actors	: <Aktor yang menjalankan use case>
Prereq / Precondition	: <Keadaan yang harus dipenuhi sebelum use case dijalankan>
Main Success Scenario	: <Poin - poin yang menjelaskan bagaimana suatu proses berjalan dalam keadaan standar>
Alternative Scenario extension	: <Poin - poin yang menjelaskan bagaimana suatu proses berjalan ketika terdapat keadaan diluar <i>main success scenario</i> >

Adapun *Use Case Description* dalam tugas akhir ini terdapat dalam Lampiran B.

4.7. Robustness Analysis

Robustness Analysis merupakan penjelasan alur yang berdasar dari use case yang penulis rancang. Robustness dibuat untuk setiap use case yang ada. Robustness menggambarkan alur dari tiap use case dan hubungan dengan use case yang lain. Adapun *Robustness Analysis* dalam tugas akhir ini terdapat dalam Lampiran D.

4.8. Updated Domain Model

Merupakan pembaruan domain model berdasarkan hasil dari Robustness Analysis. Pembaruan yang dimaksud berupa penambahan *class*, *attribute* dan *operation* yang belum ada di domain model sebelumnya. Adapun *Updates Domain Model* dalam tugas akhir ini terdapat dalam Lampiran E.

4.9. Sequence Diagram

Sequence diagram merupakan diagram yang dibuat berdasarkan robustness diagram. Diagram ini dibuat untuk setiap use case. Adapun *Sequence Diagram* dalam tugas akhir ini terdapat dalam Lampiran F.

4.10. Class Diagram

Class diagram merupakan domain model yang di anggap sudah lengkap. Adapun pada *class diagram* terdapat atribut yang nantinya ada di *database* dan perintah – perintah apa saja yang berkaitan langsung dengan database. Adapun *Class Diagram* dalam tugas akhir ini terdapat dalam Lampiran G.

Halaman ini sengaja dikosongkan

BAB V IMPLEMENTASI

Pada bab ini akan menjelaskan tahap implementasi dari aplikasi yang dibuat oleh penulis. Penjelasan yang akan penulis paparkan dimulai dari lingkungan implementasi hingga tahapan implementasi atau penulisan kode program berdasarkan dai perencanaan yang telah di buat dari bab sebelumnya.

5.1. Lingkungan Pembuatan

Dalam pembuatan sistem ini penulis menggunakan perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut:

Tabel 5.1 Spesifikasi perangkat keras

Processor	Intel® core™ i3-4030u
RAM	8 GB
VGA	Intel® HD Graphics 4400
Resolusi Grafik	1920 x 1080

Tabel 5.2 Spesifikasi perangkat lunak

Sistem Operasi	Windows 10 Pro
Apache	2.4.23
PHP	7.0.13
mySQL	mysqlnd 5.0.12-dev
Json	1.4.0
cURL	7.51.0

5.2. Pembuatan Web Service

5.2.1. Fungsi Dalam Web Service

5.2.1.1. Cek Pengguna

Fungsi ini bertujuan untuk melakukan pengecekan apakah sesi pengguna masih aktif atau tidak. Apabila sesi pengguna sudah habis maka pengguna tidak dapat melakukan aktivitas apapun sebelum melakukan *login* ulang.

```
private function login_checker()
{
    $user_data = array(
        'id_user' => $this->post('uid'),
        'key' => $this->post('key'),
        'expire_time>=' => date("Y-m-d H:i:s")
    );
    $this->cek_user = $this->Aktivitas->rincian($user_data);

    if($this->cek_user)
    {
        $cek_konfigurasi = $this->Konfigurasi->daftar();
        $status_server = $cek_konfigurasi[0]->status_site;

        if($status_server==0 && $this->cek_user[0]->level_user!='1')
        {
            $this->response(['status' => FALSE, 'message' => 'Server Sedang
Dalam Perawatan', 'alert' => 'info'],
REST_Controller::HTTP_NOT_FOUND);
        }
    }
    else
    {
        $this->response(['status' => FALSE, 'message' => 'Silahkan Login',
'alert' => 'warning'], REST_Controller::HTTP_NOT_FOUND);
    }
}
```

5.2.1.2. Cek Debitur

Fungsi ini bertujuan untuk melakukan pengecekan apakah debitur yang pengguna pilih saat memasukkan OTS,

Financial Ratio dan *Covenant* benar – benar ada dan merupakan debitor yang ada pada CMB pengguna.

```
private function debitor_checker($id){
    $data = array(
        'id_debitur'=>$id,
        'debitur_id_cmb' => $this->cek_user[0]->user_id_cmb
    );

    $hasil = $this->Debitur->daftar($data);

    if ($hasil){
        return $hasil[0]->debitur_id_grup_debitur;
    }else{
        $this->response(['status' => FALSE, 'message' => 'Debitur Yang
        Anda Pilih Tidak Ditemukan Di CMB Anda', 'alert' => 'warning'],
        REST_Controller::HTTP_NOT_FOUND);
    }
}
```

5.2.1.3. Perhitungan Financial Ratio

Fungsi ini merupakan fungsi yang menerapkan dasar teori pada sub sub bab 2.2.3, tepatnya untuk perhitungan *financial ratio*. Dengan menggunakan fungsi ini sistem akan melakukan perhitungan *financial ratio* saat pengguna memasukkan data baru maupun mengubah data *financial ratio*.

```
private function fr_calculator(){
    $nilai_kecukupan = ( ( $this->post('persediaan') + $this-
    >post('piutang') + $this->post('kas') ) / $this->post('kredit') );

    $nilai_ca = $this->post('cash') + $this->post('trade_receivables') +
    $this->post('other_receivables') + $this->post('inventory') + $this-
    >post('prepaid_expenses') + $this->post('prepaid_taxes') + $this-
    >post('sundry_ca');

    $nilai_cl = $this->post('std_bank') + $this->post('current_portion') +
    $this->post('trade_payables') + $this->post('accrued_expenses') + $this-
    >post('tax_payables') + $this->post('deferred_income') + $this-
    >post('sundry_cl') ;

    $nilai_cr = ( $nilai_ca / $nilai_cl ) * 100;
```

```

$nilai_npm = ( $this->post('laba') / $this->post('sales') ) * 100;

$nilai_dsc = ( $this->post('ebitda') / ( $this->post('angsuran') + $this->post('bunga') ) * 100 );

$this->nilai_kecukupan = $nilai_kecukupan;
$this->nilai_ca = $nilai_ca;
$this->nilai_cl = $nilai_cl;
$this->nilai_cr = $nilai_cr;
$this->nilai_npm = $nilai_npm;
$this->nilai_dsc = $nilai_dsc;
}

```

5.2.2. Daftar API Web Service

Tabel 5.3 Daftar API Web Service

Alamat		Hak Akses
Log/in	Post	Guest
Log/out	Post	Semua Pengguna
Konfigurasi/rincian	Post	Administrator
Konfigurasi/ubah	Post	Administrator
User/daftar	Post	Administrator
User/daftar_per_cmb	Post	Administrator
User/rincian	Post	Semua Pengguna
User/rincian/\$id	Post	Administrator
User/tambah	Post	Administrator
User/ubah	Post	Semua Pengguna
User/ubah/\$id	Post	Administrator
User/ubah_password	Post	Semua Pengguna
User/ubah_password/\$id	Post	Administrator
CMB/daftar	Post	Administrator, Direksi
CMB/rincian/\$id	Post	Administrator,

		Direksi
CMB/tambah	Post	Administrator
CMB/ubah/\$id	Post	Administrator
Grup_debitur/daftar	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Grup_debitur/rincian/\$id	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Grup_debitur/tambah	Post	Kepala WCR, Kepala CMB, Staff
Grup_debitur/ubah/\$id	Post	Kepala WCR, Kepala CMB, Staff
Debitur/daftar	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Debitur/daftar_per_grup_debitur	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Debitur/daftar_per_cmb	Post	Direksi
Debitur/rincian/\$id	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Debitur/tambah	Post	Kepala WCR, Kepala CMB, Staff
Debitur/ubah/\$id	Post	Kepala WCR, Kepala CMB, Staff

Debitur/setujui/\$id	Post	Kepala WCR, Kepala CMB
OTS/daftar	Post	Direksi, Kepala WCR, Kepala CMB, Staff
OTS/rincian/\$id	Post	Direksi, Kepala WCR, Kepala CMB, Staff
OTS/tambah	Post	Kepala WCR, Kepala CMB, Staff
OTS/ubah/\$id	Post	Kepala WCR, Kepala CMB, Staff
OTS/Setujui/\$id	Post	Kepala WCR, Kepala CMB
Financial_ratio/daftar	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Financial_ratio/setujui/\$id	Post	Kepala WCR, Kepala CMB
Financial_ratio/rincian/\$id	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Financial_ratio/tambah	Post	Kepala WCR, Kepala CMB, Staff
Financial_ratio/ubah/\$id	Post	Kepala WCR, Kepala CMB, Staff
Financial_ratio/setujui/\$id	Post	Kepala WCR
Covenant/daftar	Post	Direksi, Kepala WCR,

		Kepala CMB, Staff
Covenant/rincian/\$id	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Covenant/tambah	Post	Kepala WCR, Kepala CMB, Staff
Covenant/ubah/\$id	Post	Kepala WCR, Kepala CMB, Staff
Covenant/setujui/\$id	Post	Kepala WCR
Covenant/konfirmasi/\$id	Post	Kepala WCR
Covenant/Reminder/	Post	Direksi, Kepala WCR, Kepala CMB, Staff
Covenant/Reminder/\$id	Post	Direksi

5.3. Pengamanan Sistem Monitoring Debitur

Sistem yang penulis buat akan menyimpan data milik calon debitur maupun debitur yang merupakan perusahaan. Hal ini menyebabkan perlu pengamanan yang cukup ketat dalam mengakses sistem.

5.3.1. Pembatasan browser

Untuk mencegah penggunaan perangkat lunak yang berguna untuk meretas, sistem dilengkapi dengan pembatasan browser. Hanya perambah tertentu saja yang di perbolehkan untuk mengakses sistem. Selain Perambah yang diizinkan oleh penulis maka pengguna tidak dapat mengakses halaman login maupun mengakses web service secara langsung.

5.3.2. Pembatasan sesi pada web service

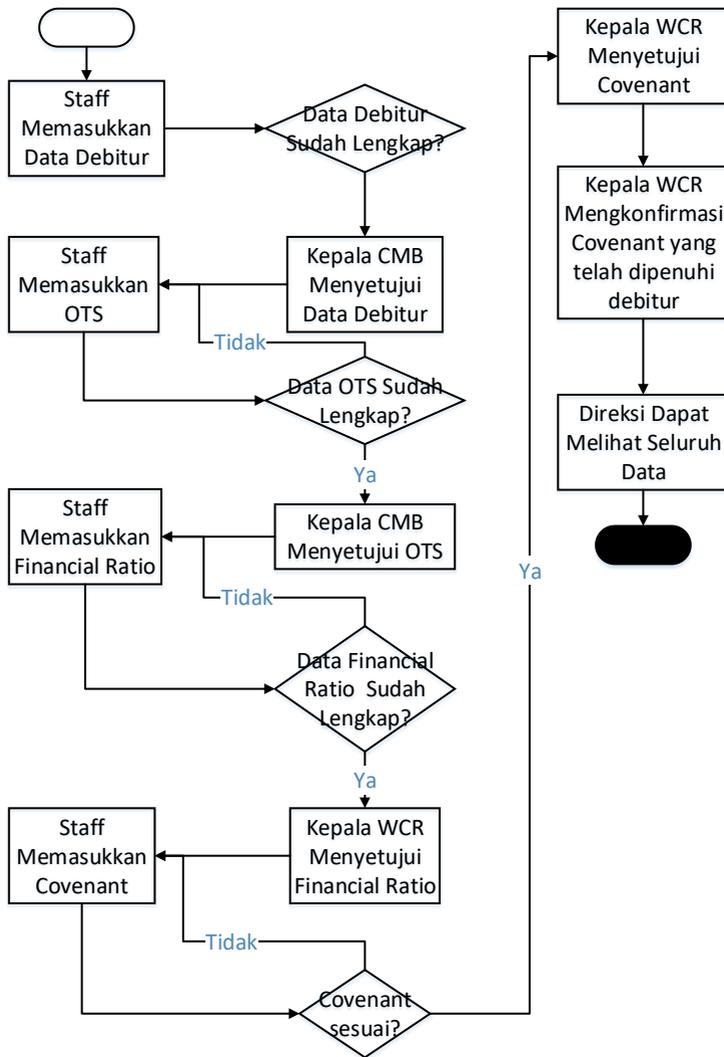
Salah satu teknik dalam peretasan adalah melakukan pencurian sesi pengguna. Untuk mengatasi pencurian sesi pengguna, maka penulis melakukan pengamanan tambahan. Selain sesi pada *presentation layer* pengguna, sistem juga memiliki sesi pada *logical layer*.

Sesi pada *logical layer* akan dibuat habis masa berlakunya ketika pengguna keluar dari sistem. Sehingga, sesi pengguna yang telah dicuri tersebut tidak dapat digunakan dengan semestinya. Orang yang mencuri sesi tersebut masih dapat login pada *presentation layer* tetapi tidak dapat menambah, mengubah maupun melihat data dalam sistem. Hal ini disebabkan *logical layer* menganggap sesi pengguna tersebut sudah tidak berlaku.

Pengamanan dengan sesi pada web service tersebut dengan cara membuat key. Key yang dibuat memiliki masa berlaku selama 24 jam. Sehingga, apabila pengguna melakukan aktivitas terus menerus melebihi 24 jam maka pengguna akan diminta untuk melakukan login ulang meskipun sesi pengguna di *presentation layer* belum habis.

5.4. Alur Proses Bisnis Pada Sistem

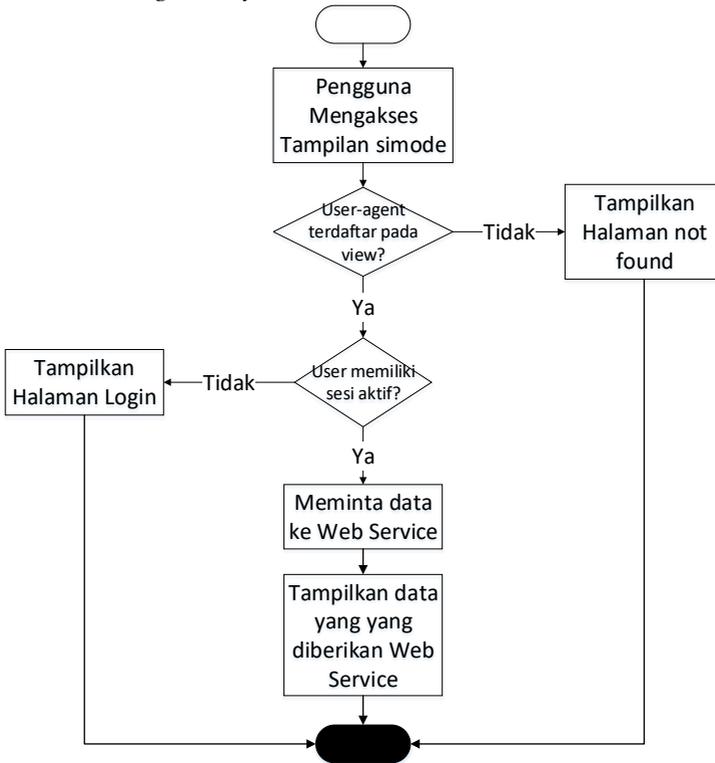
Pada bab ini penulis akan menjelaskan bagaimana alur proses bisnis dari awal staff memasukkan data hingga direksi melihat data.



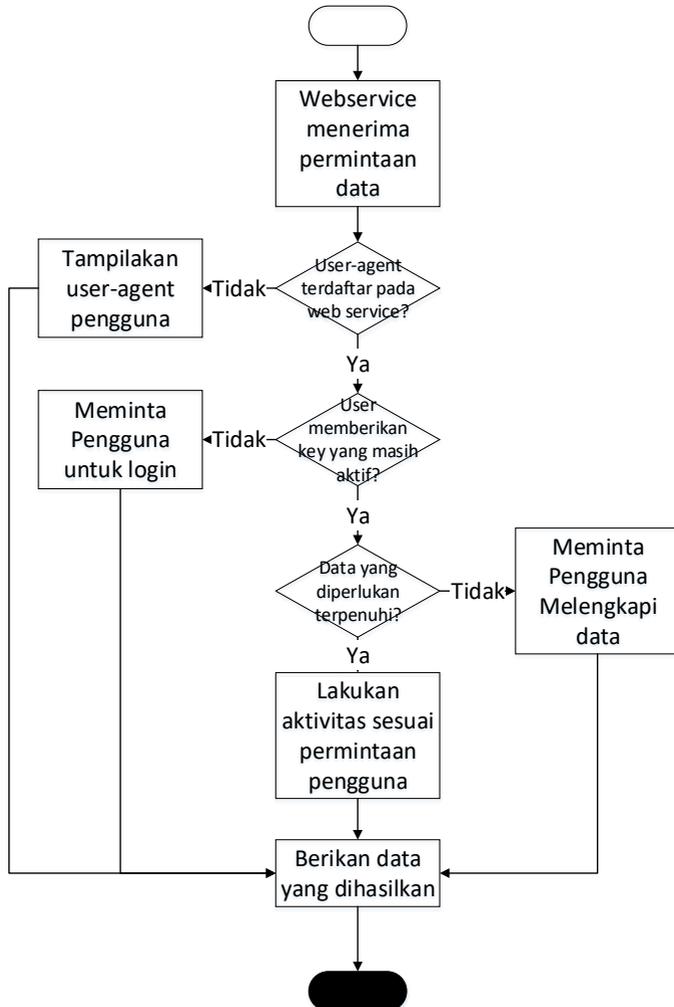
Gambar 5.1 Alur proses bisnis pada sistem

5.5. Alur Sistem

Pada sub-bab ini penulis akan menjelaskan bagaimana alur sistem secara umum. Pada Gambar 5.2 merupakan alur *presentation layer* yaitu bagaimana pengguna berinteraksi dengan sistem. Sedangkan, Gambar 5.3 merupakan alur *logical layer*, yaitu bagaimana proses dari permintaan data yang dilakukan oleh *presentation layer* hingga pengembalian hasil proses oleh *logical layer*.



Gambar 5.2 Alur pada *Presentation Layer*



Gambar 5.3 Alur pada *Logical Layer*

Halaman ini sengaja dikosongkan

BAB VI HASIL DAN PEMBAHASAN

Pada bab ini penulis akan menjelaskan tentang hasil dari pengujian aplikasi yang penulis buat.

6.1. Lingkungan Uji Coba

Dalam melakukan uji coba sistem ini, penulis menggunakan perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut:

Tabel 6.1 Spesifikasi perangkat keras

Processor	Intel® core™ i3-4030u
RAM	8 GB
VGA	Intel® HD Graphics 4400
Resolusi Grafik	1920 x 1080

Tabel 6.2 Spesifikasi perangkat lunak

Sistem Operasi	Windows 10 Pro
Apache	2.4.23
PHP	7.0.13
mySQL	mysqlnd 5.0.12-dev
Json	1.4.0
cURL	7.51.0

6.2. Requirements Traceability Matrix

Pada bab ini penulis melakukan pengujian untuk membuktikan bahwa setiap kebutuhan sudah terpenuhi. Penulis melakukan kan pengujian dengan *requirements traceability matrix*, Pada dapat dilihat bahwa setiap kebutuhan sudah memiliki *design* dan *coding*, dan tidak ada *design* dan *coding* yang tidak memiliki kebutuhan,

REQUIREMENT				DESIGN				Coding	
ID	Business Requirement	Functional Requirement	Priority	Use Case ID	Use Case Name	Use Case Description	Sequence Diagram	Web Service	View
1	Setiap data yang terdapat di dalam sistem hanya dapat di akses oleh orang yang berkepentingan	FR-101	High	UC-01	Login	UD-01	SD-01	Log	Log
		FR-101	High	UC-02	Logout	UD-02	SD-02	Log	Log
		FR-101	High	UC-05	Mengganti Password	UD-05	SD-05	User	Profil
		FR-102	High	UC-10	Melihat Daftar CMB	UD-10	SD-10	CMB	CMB
		FR-102	High	UC-29	Melihat Daftar Covenant	UD-29	SD-29	Covenant	Covenant

		FR-102	High	UC-19	Melihat Daftar Debitur	UD-19	SD-19	Debitur	Debitur
		FR-102	High	UC-35	Melihat Daftar Financial Ratio	UD-35	SD-35	Financial_ratio	Financial_ratio
		FR-102	High	UC-15	Melihat Daftar Grup Debitur	UD-15	SD-15	Grup_debitur	Grup_debitur
		FR-102	High	UC-24	Melihat Daftar OTS	UD-24	SD-24	OTS	OTS
		FR-102	High	UC-06	Melihat Daftar Pengguna	UD-06	SD-06	User	Pengguna
		FR-201	High	UC-11	Melihat Rincian CMB	UD-11	SD-11	CMB	CMB
		FR-201	High	UC-30	Melihat Rincian Covenant	UD-30	SD-30	Covenant	Covenant
		FR-201	High	UC-20	Melihat Rincian Debitur	UD-20	SD-20	Debitur	Debitur
		FR-201	High	UC-36	Melihat Rincian Financial Ratio	UD-36	SD-36	Financial_ratio	Financial_ratio

		FR-201	High	UC-16	Melihat Rincian Grup Debitur	UD-16	SD-16	Grup_debitur	Grup_debitur
		FR-201	High	UC-25	Melihat Rincian OTS	UD-25	SD-25	OTS	OTS
		FR-201	High	UC-07	Melihat Rincian Pengguna	UD-07	SD-07	User	Pengguna
		FR-201	High	UC-03	Melihat Rincian Profil	UD-03	SD-03	User	Profil
2	Data dan informasi mengenai debitur di simpan dan dapat diubah apabila ada data yang dinilai kurang sesuai	FR-202	High	UC-12	Menambah CMB	UD-12	SD-12	CMB	CMB
		FR-202	High	UC-31	Menambah Covenant	UD-31	SD-31	Covenant	Covenant
		FR-202	High	UC-21	Menambah Debitur	UD-21	SD-21	Debitur	Debitur
		FR-202	High	UC-37	Menambah Financial Ratio	UD-37	SD-37	Financial_ratio	Financial_ratio
		FR-202	High	UC-17	Menambah Grup Debitur	UD-17	SD-17	Grup_debitur	Grup_debitur
		FR-202	High	UC-26	Menambah OTS	UD-26	SD-26	OTS	OTS

		FR-202	High	UC-08	Menambah Pengguna	UD-08	SD-08	User	Pengguna
		FR-202	High	UC-13	Mengubah Rincian CMB	UD-13	SD-13	CMB	CMB
		FR-202	High	UC-32	Mengubah Rincian Covenant	UD-32	SD-32	Covenant	Covenant
		FR-202	High	UC-22	Mengubah Rincian Debitur	UD-22	SD-22	Debitur	Debitur
		FR-202	High	UC-38	Mengubah Rincian Financial Ratio	UD-38	SD-38	Financial_ratio	Financial_ratio
		FR-202	High	UC-18	Mengubah Rincian Grup Debitur	UD-18	SD-18	Grup_debitur	Grup_debitur
		FR-202	High	UC-27	Mengubah Rincian OTS	UD-27	SD-27	OTS	OTS
		FR-202	High	UC-09	Mengubah Rincian Pengguna	UD-09	SD-09	User	Pengguna

		FR-202	High	UC-04	Mengubah Rincian Profil	UD-04	SD-04	User	Profil
3	Setiap data terkait calon debitur dan debitur akan diverifikasi oleh Kepala CMB dan Kepala WCR	FR-203	High	UC-33	Mengkonfirmasi Covenant	UD-33	SD-33	Covenant	Covenant
		FR-203	High	UC-34	Menyetujui Covenant	UD-34	SD-34	Covenant	Covenant
		FR-203	High	UC-23	Menyetujui Debitur	UD-23	SD-23	Debitur	Debitur
		FR-203	High	UC-39	Menyetujui Financial Ratio	UD-39	SD-39	Financial_ratio	Financial_ratio
		FR-203	High	UC-28	Menyetujui OTS	UD-28	SD-28	OTS	OTS
4	Memastikan tidak terjadi kehilangan data	FR-501	High	UC-14	Manajemen Konfigurasi Sistem	UD-14	SD-14	Konfigurasi	Pengaturan

6.3. Pengujian Fungsional

Pengujian fungsional didapat dengan menjalankan aplikasi dan melakukan aktivitas sesuai dengan skenario yang telah ditentukan. Hasil pengujian fungsional dapat di lihat pada tabel.

Tabel 6.3 Pengujian fungsional

No	Skenario	Hasil yang di harap	Hasil yang didapat
1.	Login dengan username dan password yang benar	Pengguna masuk ke halaman sesuai dengan level yang di miliki	Sesuai
2.	Pengguna gagal login sebanyak 3 kali	Sistem meminta masukan berupa captcha	Sesuai
3.	Pengguna mengakses halaman yang bukan haknya	Sistem memindahkan pengguna ke halaman utama	Sesuai
4.	Administrator menambah pengguna baru	Terdapat pengguna baru di daftar pengguna	Sesuai
5.	Administrator menambah pengguna dengan username yang sudah ada	Muncul peringatan gagal menambahkan pengguna	Sesuai
6.	Administrator mengubah password pengguna	Muncu peringatan berhasil mengganti password	Sesuai
7.	Administrator mengubah password pengguna, namun password baru dan	Muncul peringatan gagal mengganti password	Sesuai

	konfirmasi password tidak sama		
8.	Administrator menambah CMB baru	Muncul peringatan berhasil menambahkan CMB baru	Sesuai
9.	Administrator menambah CMB baru dengan nama yang sama dengan yang sudah ada	Muncul peringatan gagal menambahkan CMB baru	Sesuai
10.	Administrator mengubah nama CMB	Muncul peringatan berhasil memperbarui CMB	Sesuai
11.	Administrator mengubah nama CMB dengan nama yang sudah ada	Muncul peringatan gagal memperbarui CMB	Sesuai
12.	Staff menambah Grup debitur baru	Muncul peringatan berhasil menambahkan grup debitur	Sesuai
13.	Staff menambah grup debitur baru dengan nama yang sudah ada	Muncul peringatan gagal mengubah grup debitur	Sesuai
14.	Staff menambah debitur baru	Muncul peringatan berhasil menambahkan debitur baru	Sesuai
15.	Staff menambah debitur baru dengan nama yang sudah ada	Muncul peringatan gagal	Sesuai

		menambahkan debitur	
16.	Staff mengubah rincian debitur	Muncul peringatan berhasil mengubah rincian debitur	Sesuai
17.	Staff mengubah nama debitur dengan nama yang sudah ada	Muncul peringatan gagal mengubah rincian ddebitur	Sesuai
18.	Kepala CMB menyetujui debitur	Muncul peringatan berhasil menyetujui debitur	Sesuai
19.	Staff menambah OTS baru	Muncul peringatan berhasil menyetujui OTS baru	Sesuai
20.	Staff menambah OTS baru dengan tanggal yang sudah ada	Muncul peringatan gagal menambah OTS baru	Sesuai
21.	Staff mengubah OTS dengan tanggal OTS yang sudah ada	Muncul peringatan gagal mengubah rincian OTS	Sesuai
22.	Kepala CMB menyetujui OTS	Muncul peringatan berhasil menyetujui OTS	Sesuai
23.	Staff menambah financial ratio baru	Muncul peringatan berhasil membuat	Sesuai

		financial ratio baru	
24.	Kepala WCR menyetujui financial ratio	Muncul peringatan berhasil menyetujui financial ratio	Sesuai
25.	Staff menambah covenant	Muncul peringatan berhasil menambah covenant	Sesuai
26.	Kepala WCR Menyetujui covenant	Muncul peringatan berhasil menyetujui covenant	Sesuai
27.	Kepala WCR mengkonfirmasi covenant	Muncul peringatan berhasil mengkonfirmasi covenant	Sesuai
28.	Pengguna memperbarui rincian profil	Muncul peringatan berhasil memperbarui profil	Sesuai
29.	Pengguna mengganti password	Muncul peringatan berhasil mengganti password	Sesuai
30.	Pengguna mengganti password namun password lama salah	Muncul peringatan gagal mengganti password	Sesuai

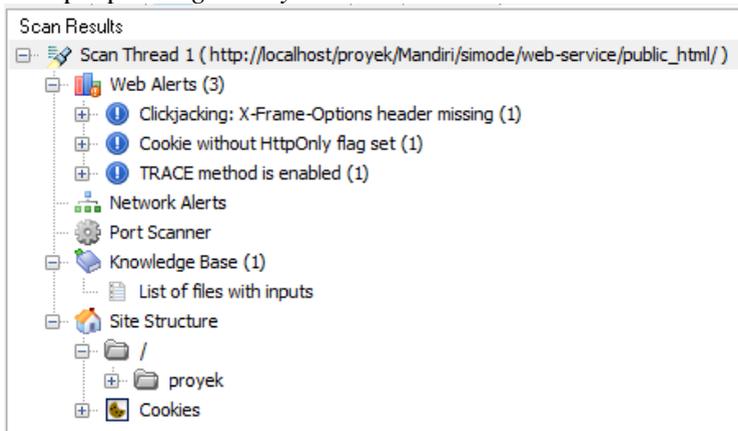
31.	Pengguna mengganti password namun password baru dan konfirmasi password tidak sesuai	Muncul peringatan gagal memperbarui password	Sesuai
-----	--	--	--------

6.4. Pengujian Keamanan

Pengujian keamanan dilakukan dalam rangka memenuhi kebutuhan NFR-201, yaitu terkait keamanan data yang terdapat di dalam sistem. Penulis melakukan pengujian untuk memastikan tidak ada celah berbahaya yang dapat dimanfaatkan oleh pihak-pihak tidak bertanggung jawab. Pengujian dilakukan pada logical layer dan presentation layer.

6.4.1. Pengujian pada Logical Layer

Penulis melakukan melakukan pengujian keamanan pada *logical layer* dengan cara melakukan *web vulnerability scanning*. Adapun hasil pengujian dapat dilihat pada Gambar 6.1. Terdapat tiga celah keamanan dengan tingkat ancaman rendah sehingga tidak membahayakan *web service* yang terdapat pada *logical layer*.

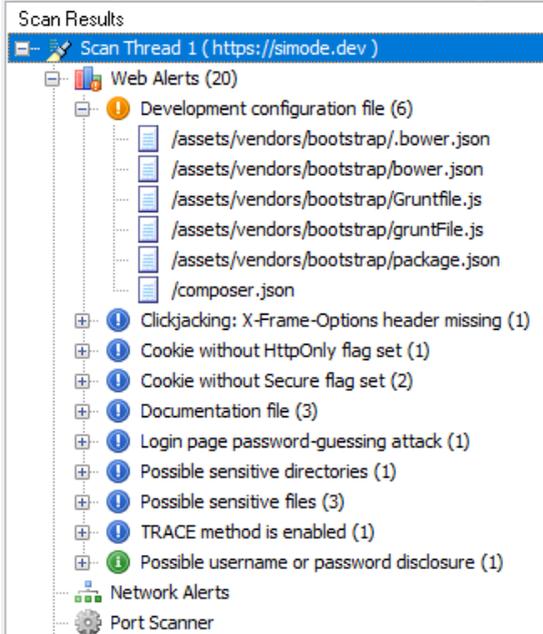


Gambar 6.1 Hasil pengujian keamanan *logical layer*

6.4.2. Pengujian pada Presentation Layer

Penulis melakukan melakukan pengujian keamanan pada *logical layer* dengan cara melakukan *web vulnerability scanning*. Adapun hasil pengujian dapat dilihat pada Gambar 6.2. Terdapat satu celah keamanan yang bersifat informatif, delapan celah dengan tingkat ancaman rendah, dan satu celah dengan tingkat ancaman sedang.

Celah keamanan yang bersifat informatif berasal dari salah satu font yang penulis gunakan, dan tidak membahayakan sistem. Sedangkan pada celah dengan tingkat ancaman rendah berasal dari *framework* tampilan yang penulis gunakan dan membahayakan sistem. Sehingga dapat disimpulkan, celah keaman pada tampilan sistem masuk dalam kategori rendah.



Gambar 6.2 Hasil pengujian pada *presentation layer*

6.5. Pengujian compatibility Browser

Pada sub-bab ini penulis akan menguji *compatibility* antara aplikasi yang telah penulis buat dengan beberapa alat

perambah yang ada. Pengujian dilakukan pada perangkat *desktop* dan *mobile* dengan tujuan untuk memenuhi NFR-302 sekaligus melakukan pengecekan pembatasan browser yang telah penulis rancang di bab sebelumnya.

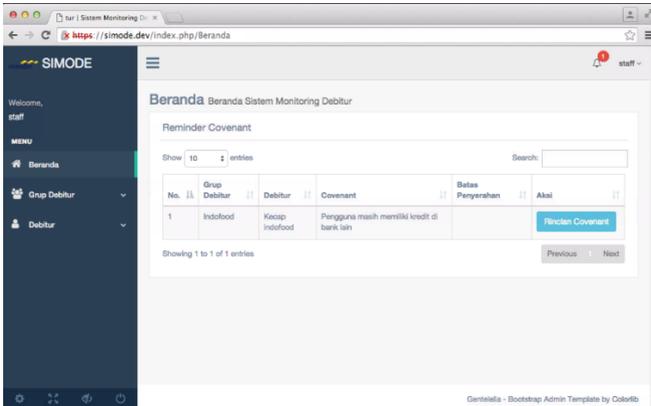
Tabel 6.4 Hasil pengujian *compatibility browser*

Sistem Operasi	Perambah	Hasil yang diharapkan	Hasil yang didapat
Mac OS X 10.8	Google Chrome 48	Dapat mengakses	Sesuai
Mac OS X 10.8	Safari 6.2	Dapat mengakses	Sesuai
Mac OS X 10.8	Firefox	Tidak Dapat mengakses	Sesuai
Galaxy Note 3 / 4.4	Android Browser 4.4	Dapat mengakses	Sesuai
Galaxy Note 3 / 4.4	Chrome Mobile 58	Dapat mengakses	Sesuai
Galaxy Note 3 / 4.4	Firefox Mobile 53	Tidak Dapat mengakses	Sesuai
IPad 3 / 6.1	Safari 6.0	Dapat mengakses	Sesuai

6.5.1. Perangkat *Desktop*

6.5.1.1. Mac OSX 10.8 – Google Chrome 48

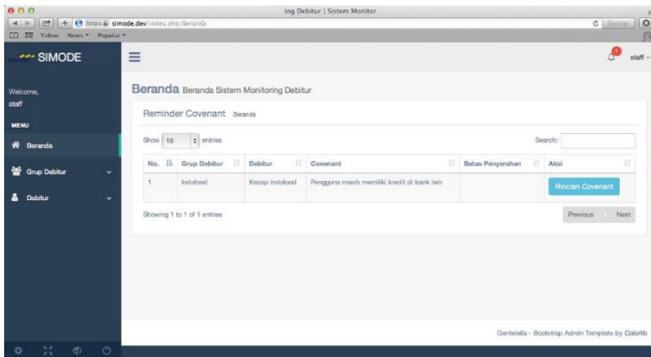
Perambah sudah terdaftar sehingga dapat mengakses halaman login maupun yang lain. Tampilan yang dihasilkan juga sesuai dengan yang dibuat oleh penulis.



Gambar 6.3 Hasil Pengujian Pada Google Chrome 48

6.5.1.2. Mac OSX 10.8 – Safari 6.2

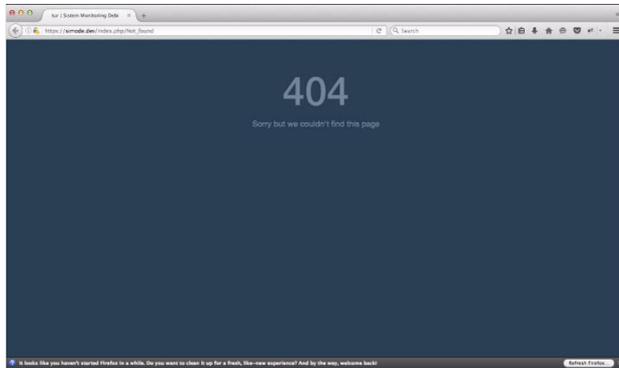
Perambah sudah terdaftar sehingga dapat mengakses halaman login maupun yang lain. Namun terdapat *bug* pada bagian footer.



Gambar 6.4 Hasil pengujian pada Safari 6.2

6.5.1.3. Mac OSX 10.8 – Firefox 48

Merupakan contoh dari browser desktop dengan *user-agent* yang tidak termasuk dalam daftar. Sehingga, browser tidak dapat mengakses halaman login maupun halaman lain. Hanya halaman not_found yang dapat di akses.

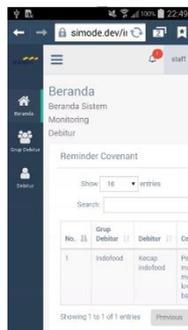


Gambar 6.5 Hasil pengujian pada Firefox 48

6.5.2. Perangkat *Mobile*

6.5.2.1. Galaxy Note 3 – Android Browser 4.4

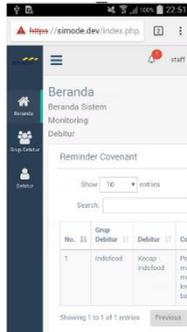
Perambah sudah terdaftar sehingga dapat mengakses halaman login maupun yang lain. Tampilan yang dihasilkan juga sesuai dengan yang dibuat oleh penulis.



Gambar 6.6 Hasil pengujian pada Android Browser 4.4

6.5.2.2. Galaxy Note 3 – Chrome Mobile 58

Perambah sudah terdaftar sehingga dapat mengakses halaman login maupun yang lain. Tampilan yang dihasilkan juga sesuai dengan yang dibuat oleh penulis.



Gambar 6.7 Pengujian Pada Chrome Mobile 58

6.5.2.3. Galaxy Note 3 – Firefox Mobile 53

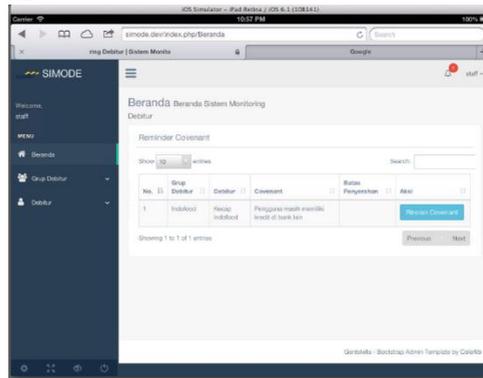
Merupakan contoh dari perangkat browser dengan *user-agent* yang tidak termasuk dalam daftar. Sehingga, browser tidak dapat mengakses halaman login maupun halaman lain. Hanya halaman `not_found` yang dapat di akses.



Gambar 6.8 Hasil pengujian pada Firefox Mobile 53

6.5.2.4. iPad 3 / 6.1 – Mobile Safari 6.0

Perambah sudah terdaftar sehingga dapat mengakses halaman login maupun yang lain. Namun terdapat *bug* pada bagian footer.



Gambar 6.9 Hasil pengujian pada Mobile Safari 6.0

6.6. Instalasi Sistem Monitoring Debitur

Pada sub-bab ini penulis melakukan pengujian instalasi dengan menggunakan perangkat yang sudah di desain dengan menggunakan arsitektur *three-tier*. Pengujian ini bertujuan untuk memastikan bahwa aplikasi benar-benar siap untuk dijalankan pada arsitektur *three-tier*.

6.6.1. Data Layer

1. Buat database baru



Gambar 6.10 Buat database baru

2. Import database



Importing into the database "simode"

File to import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
 A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Browse your computer: No file chosen (Max: 2,048KiB)

You may also drag and drop a file on any page.

Gambar 6.11 Import Database Simode

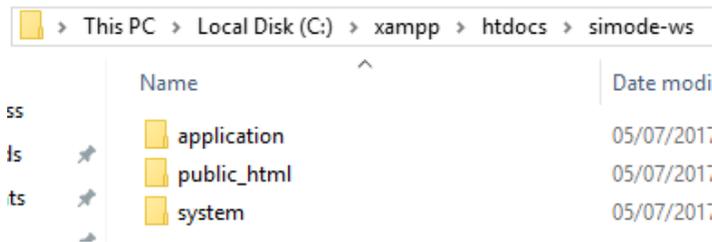
3. Bila berhasil struktur database akan menjadi seperti gambar di bawah ini



Gambar 6.12 Struktur database

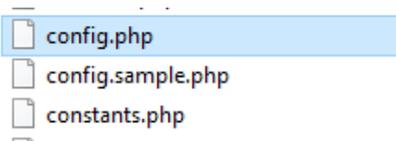
6.6.2. Logical Layer

1. Upload file – file simode-ws ke server logical atau yang digunakan sebagai API



Gambar 6.13 file – file simode-ws

2. Edit file config.php yang berada di ../application/config/config.php



Gambar 6.14 ubah config

3. Ubah base_url menyesuaikan server

```
$config['base_url'] = 'http://localhost/
```

Gambar 6.15 file config simode-ws

4. Edit file config.php yang berada di ../application/config/database.php

PC > Local Disk (C:) > xampp > htdocs > simode-ws > application > config

Name	Date modified	Type
autoload.php	09/01/2017 09.46	PHP File
config.php	05/07/2017 08.56	PHP File
config.sample.php	09/01/2017 09.46	PHP File
constants.php	09/01/2017 09.46	PHP File
database.php	05/07/2017 09.44	PHP File
database.sample.php	09/01/2017 09.46	PHP File
doctypes.php	09/01/2017 09.46	PHP File
foreign_chars.php	09/01/2017 09.46	PHP File
hooks.php	09/01/2017 09.46	PHP File

Gambar 6.16 ubah database

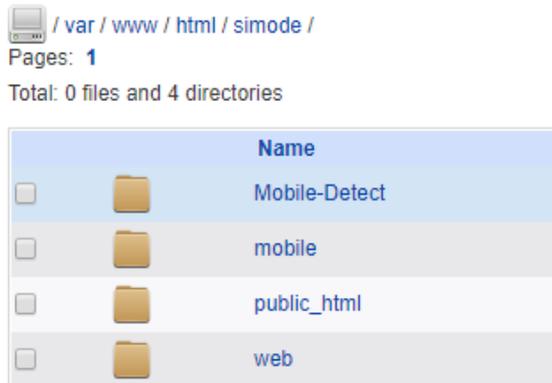
5. Ubah setting database.php sesuai dengan server database

```
$db['default'] = array(
    'dsn' => '',
    'hostname' => '192.168.169.150',
    'username' => 'remote',
    'password' => 'remote',
    'database' => 'simode',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);
```

Gambar 6.17 pengaturan database

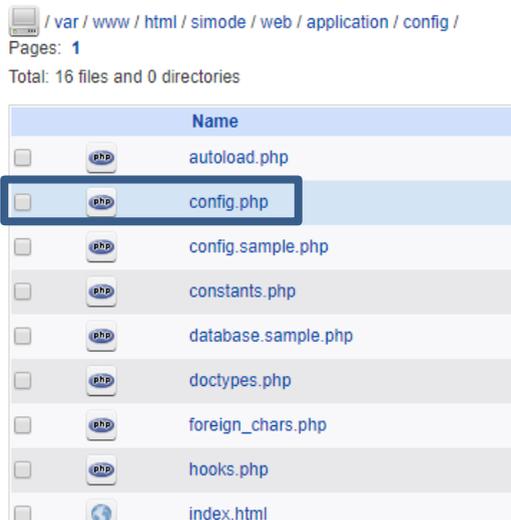
6.6.3. Presentation Layer

1. Upload file – file simode ke server yang digunakan sebagai presentation layer



Gambar 6.18 Upload file simode

2. Ubah file config.php



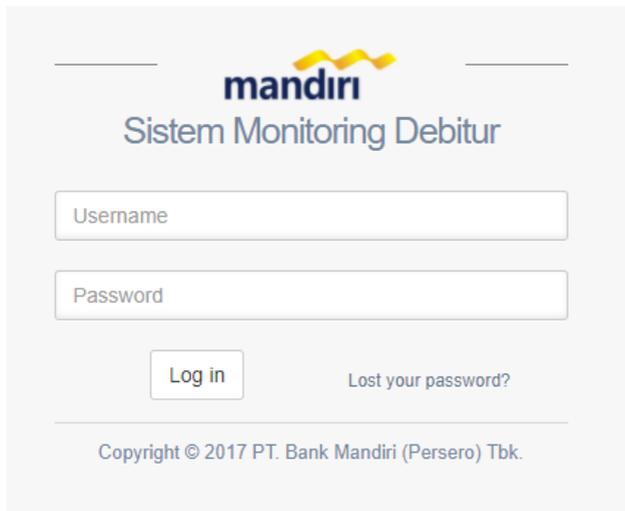
Gambar 6.19 Ubah file config

- Ubah `base_url` dan `file_url` sesuai dengan server presentation layer, sedangkan `webservice_url` sesuai dengan alamat web service

```
24 */
25 $config['base_url'] = ($_SERVER[
26 $config['file_url'] = ($_SERVER[
27 $config['webservice_url'] = 'http
28
```

Gambar 6.20 config pada presentation layer

- Lakukan uji coba dengan membuka alamat simode apabila muncul tampilan seperti dibawah ini berarti sudah benar.



mandiri

Sistem Monitoring Debitur

Username

Password

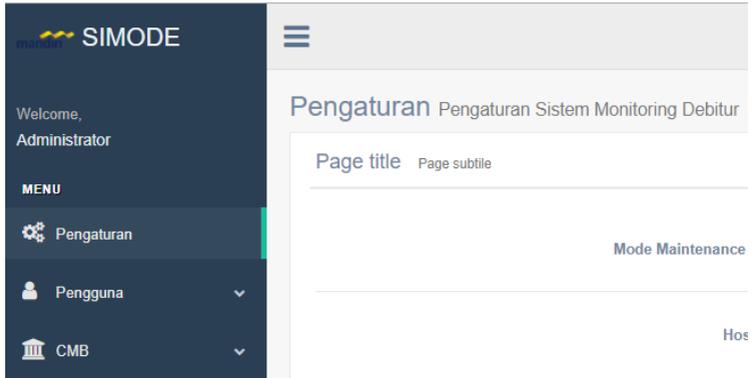
Log in

[Lost your password?](#)

Copyright © 2017 PT. Bank Mandiri (Persero) Tbk.

Gambar 6.21 Halaman login simode

5. Coba login dengan username 'admin' dan password 'admin' apabila berhasil login berarti antara data layer, logical layer dan presentation layer sudah terhubung.



Gambar 6.22 Tampilan bila berhasil login

Halaman ini sengaja dikosongkan

BAB VII

KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan yang diambil berdasarkan keseluruhan pengerjaan tugas akhir yang telah dilakukan. Pada bab ini juga berisikan beberapa saran beserta permasalahan yang dihadapi selama pengerjaan tugas akhir guna menyempurnakan dan perbaikan untuk penelitian yang akan datang terkait permasalahan klasifikasi persediaan menggunakan metode *ABC-Fuzzy Classification*.

7.1. Kesimpulan

Kesimpulan yang diperoleh dari proses pengerjaan tugas akhir yang telah dilakukan adalah:

1. Sistem monitoring debitur memiliki fungsi untuk menyetujui rincian data dari debitur, rincian data yang telah di setujui menandakan data tersebut telah di periksa dan telah sesuai.
2. Sistem monitoring debitur memiliki fungsi konfirmasi untuk menandai covenant yang sudah dipenuhi oleh debitur
3. Sistem telah melalui pengujian fungsi, kemanan, *compatibility*, dan dibuat berdasarkan jaringan yang saat ini ada sehingga sistem dapat dan layak untuk diterapkan.

7.2. Saran

Dalam pengerjaan tugas akhir ini, terdapat beberapa hal yang masih kurang dari sempurna dan masih banyak yang harus diperbaiki lagi. Berikut beberapa saran dari penulis yang dapat dijadikan referensi untuk pengembangan yang lebih baik lagi:

1. Perlu pengembangan lebih lanjut untuk dibuatkan aplikasi berbasis android atau perangkat smartpone lain agar pengguna dapat lebih mudah mengakses sistem.

2. Kebutuhan pengguna untuk tingkat pengambil keputusan paling akhir yaitu direksi perlu digali lebih dalam agar sistem dapat berfungsi dengan lebih maksimal.
3. Sistem harus dikembangkan lebih dalam lagi agar dapat menggantikan aktivitas – aktivitas yang saat ini masih dilakukan secara manual, bukan hanya sebagai pendamping aktivitas yang ada.

DAFTAR PUSTAKA

- [1] I. B. Indonesia, *Mengelola Kredit Secara Sehat*, Jakarta: PT Gramedia Pustaka Utama, 2014.
- [2] S. R. D. SETIAWAN, "Kompas," Kompas, 06 01 2017. [Online]. Available: <http://bisniskeuangan.kompas.com/read/2017/01/06/184755826/bi.pertumbuhan.kredit.di.2016.capai.9.persen>. [Accessed 21 06 2017].
- [3] M. Syafrizal, "ISO 17799: Standar Sistem Manajemen Keamanan Informasi," 2007, 24 11 2007.
- [4] "Perancangan dan Pembuatan Sistem Informasi Manajemen Keuangan Sub Bagian Perbendaharaan, Studi Kasus Pemerintahan Kabupaten Malang".
- [5] D. B. P. Yuli Fauziah, "Pengembangan Sistem Informasi Perparkiran Menggunakan Arsitektur Three-tier," Prodi Teknik Informatika, Fakultas Teknologi Industri, UPN "Veteran" Yogyakarta, Yogyakarta, 2015.
- [6] T. W. U. P. Eka Nur Ahmad Romadhoni, "Implementasi Model Waterfall pada Pengembangan Sistem Informasi Alumni SMKN 1 Jenangan Ponorogo," *Seminar Nasional Informatika*, 2015.
- [7] B. A. Pratama, "ANALISIS FAKTOR - FAKTOR YANG MEMPENGARUHI KEBIJAKAN PENYALURAN KREDIT PERBANKAN," UNIVERSITAS DIPONEGORO, SEMARANG, 2010.
- [8] "CodeIgniter," Ellislab, [Online]. Available: <https://ellislab.com/codeigniter>. [Accessed 28 Februari 2016].

- [9] Oracle, "What Are RESTful Web Services?," Oracle, [Online]. Available: <http://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html>. [Accessed 12 Maret 2016].
- [10] "Introducing JSON," [Online]. Available: <http://www.json.org/>. [Accessed 12 Maret 2016].
- [11] S. IT, "DB-Engines Ranking," SOLID IT, [Online]. Available: <http://db-engines.com/en/ranking>. [Accessed 12 Maret 2016].
- [12] O. Corporation, "MySQL Editions," Oracle Corporation, [Online]. Available: <https://www.mysql.com/products/>. [Accessed 12 Maret 2016].

BIODATA PENULIS

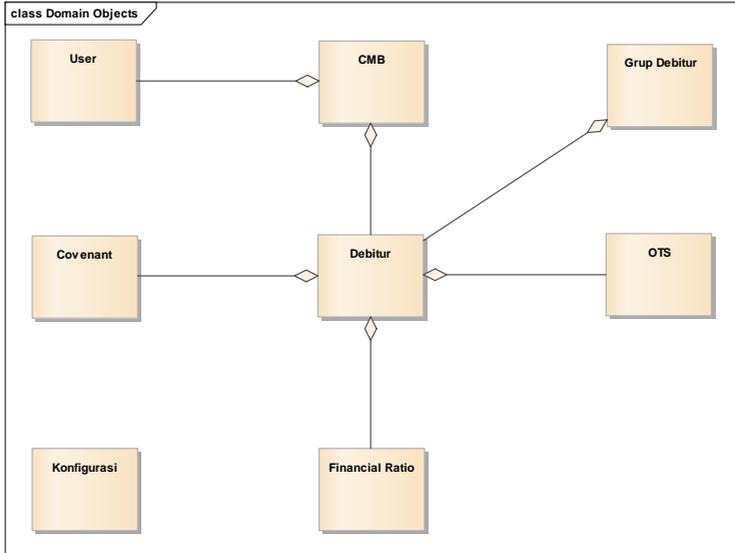


Penulis lahir di Surabaya pada tanggal 30 April tahun 1994. Merupakan anak pertama dari 3 bersaudara. Penulis telah menempuh beberapa pendidikan formal yaitu; SDN Simomulyo IV Surabaya, SMPN Negeri 2 Surabaya dan SMAN 2 Surabaya. Pada tahun 2012 pasca kelulusan SMA penulis melanjutkan pendidikan di departemen Sistem Informasi FTIf – Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa dengan NRP 5212100161.

Ketertarikannya pada desain manajemen jaringan dan keamanan asset informasi membuat penulis mengambil bidang minat Laboratorium Infrastruktur dan Keamanan Teknologi Informasi (Lab. IKTI) pada tahun keempat dan menjadikan pembuatan sistem yang aman dengan menggunakan arsitektur *three-tier* sebagai topik dalam pengerjaan tugas akhir. Penulis dapat dihubungi melalui *e-mail* igedeadityawiratama@gmail.com.

Halaman ini sengaja dikosongkan

LAMPIRAN A DOMAIN MODEL

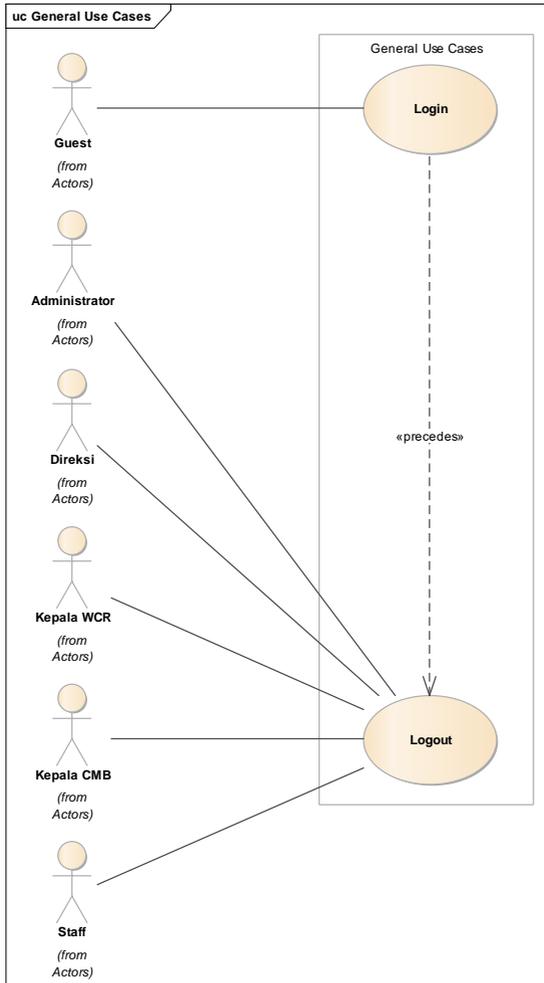


A-2

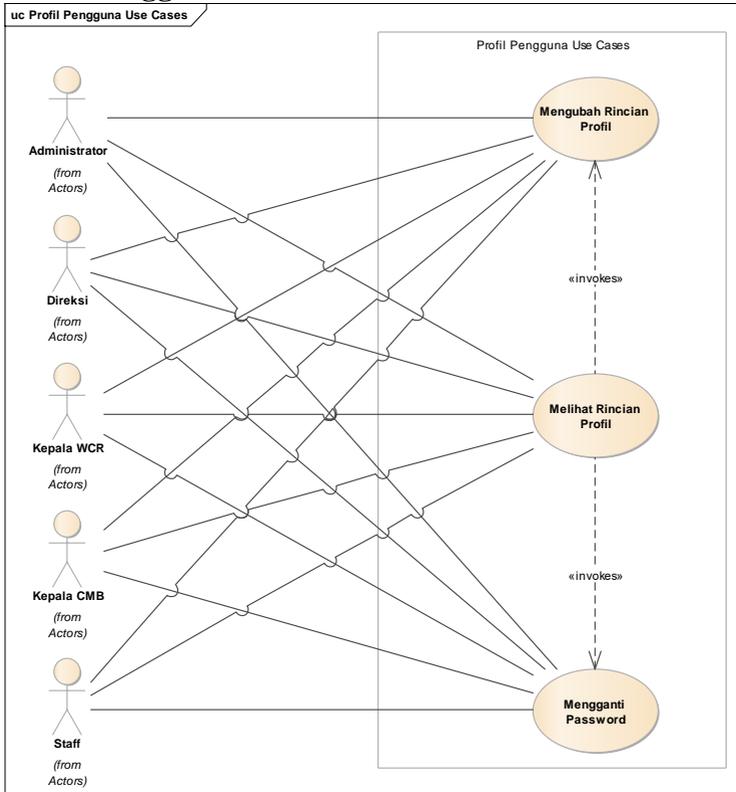
Halaman ini sengaja dikosongkan

LAMPIRAN B USE CASE DIAGRAM

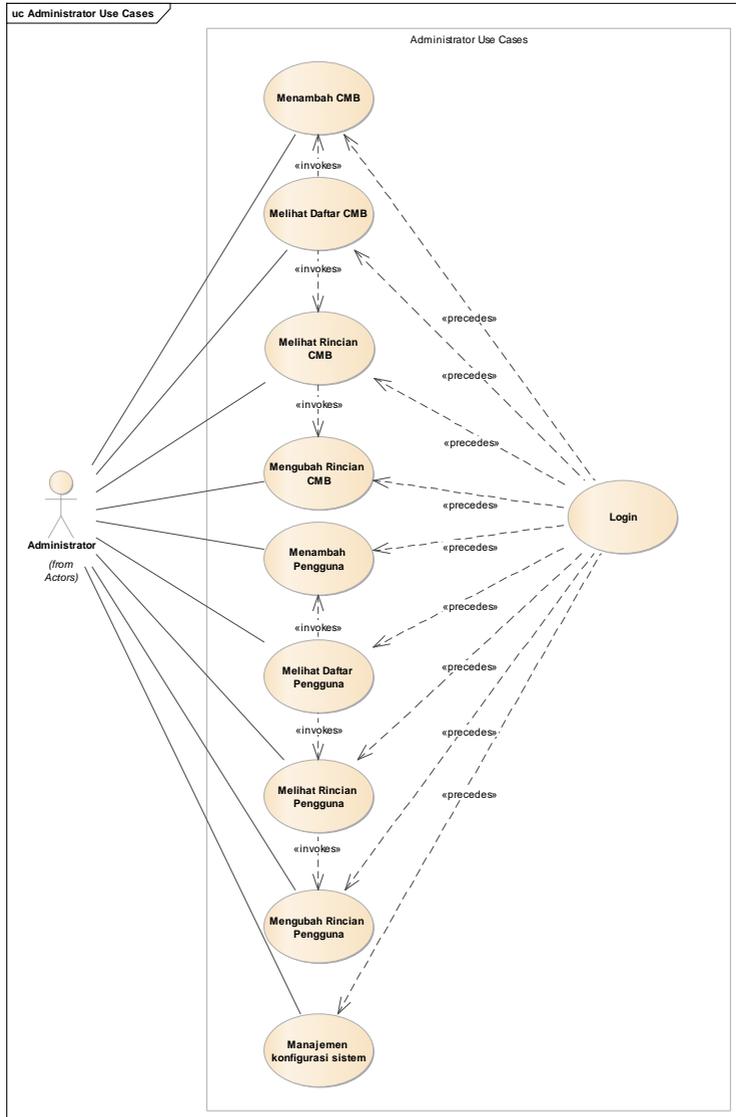
General Use Cases



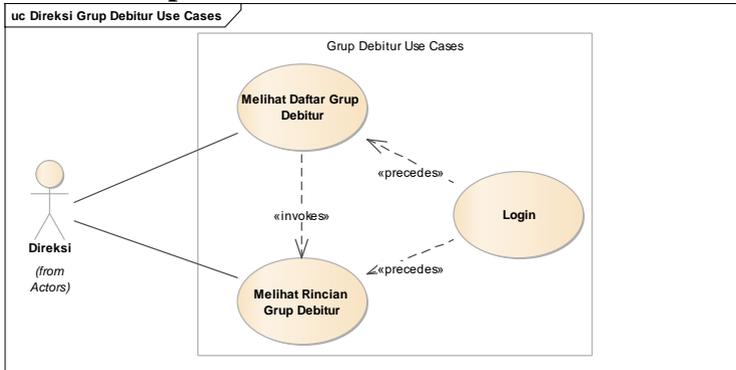
Profil Pengguna Use Cases



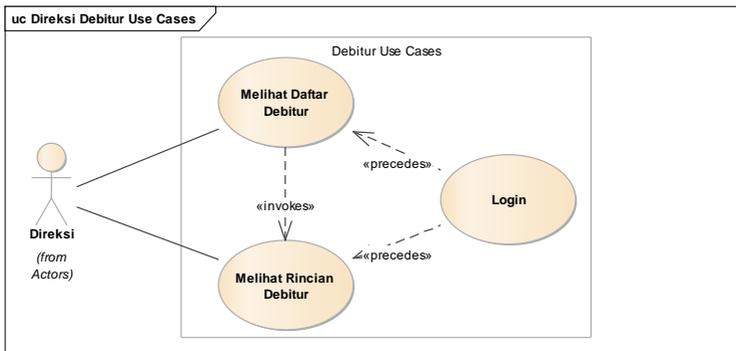
Administrator Use Cases



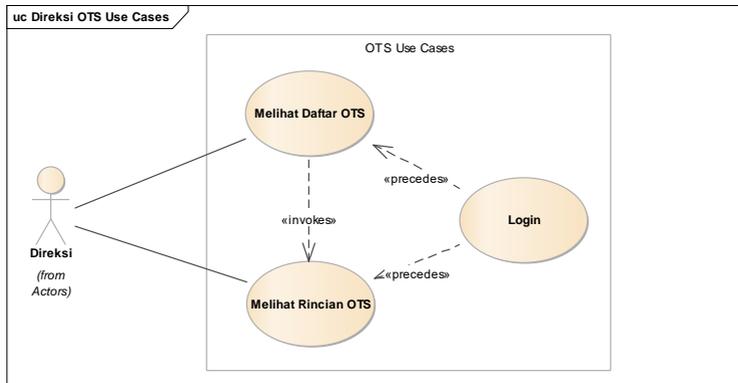
Direksi Grup Debitur Use Cases



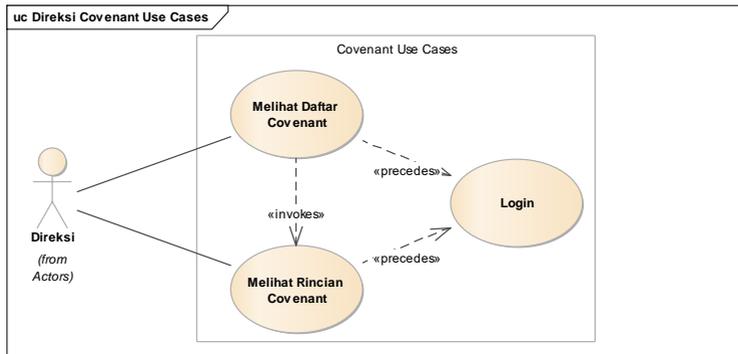
Direksi Debitur Use Cases



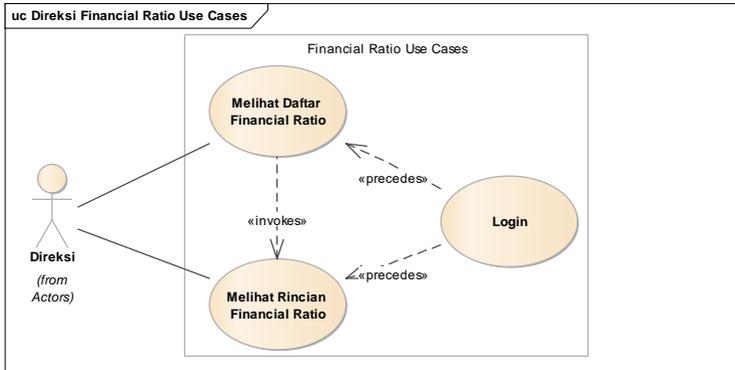
Direksi OTS Use Cases



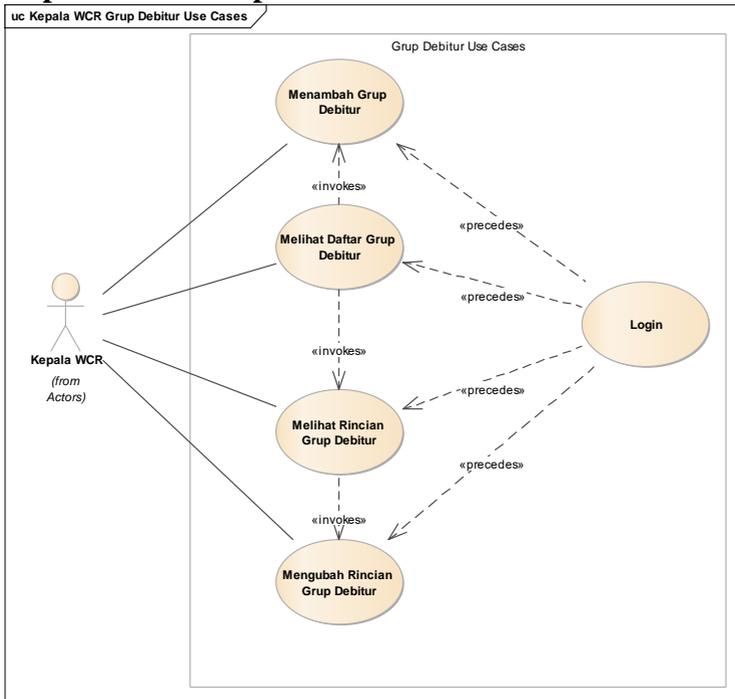
Direksi Covenant Use Cases



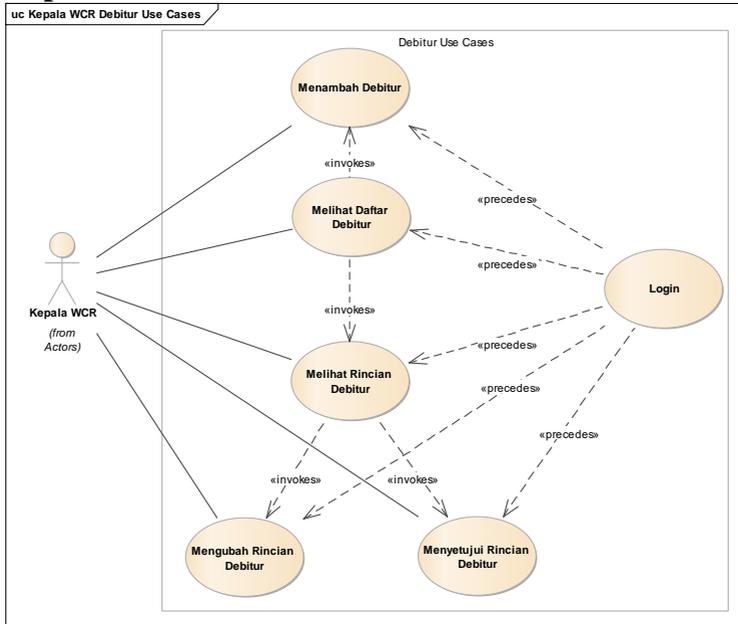
Direksi Financial Ratio Use Cases



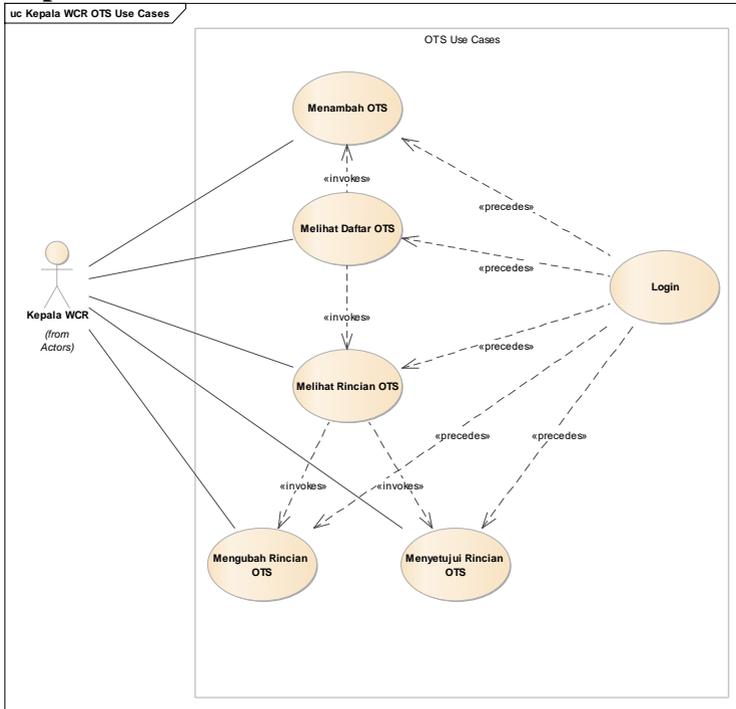
Kepala WCR Grup Debitur Use Cases



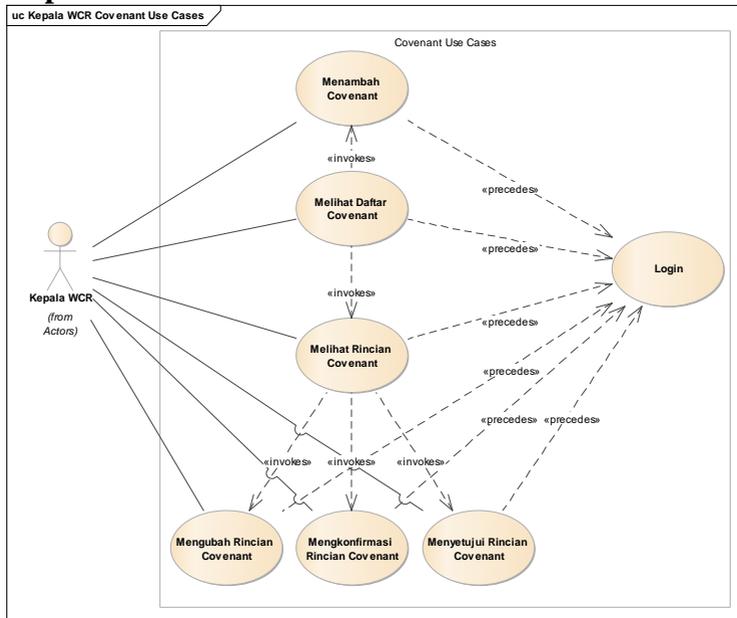
Kepala WCR Debitur Use Cases



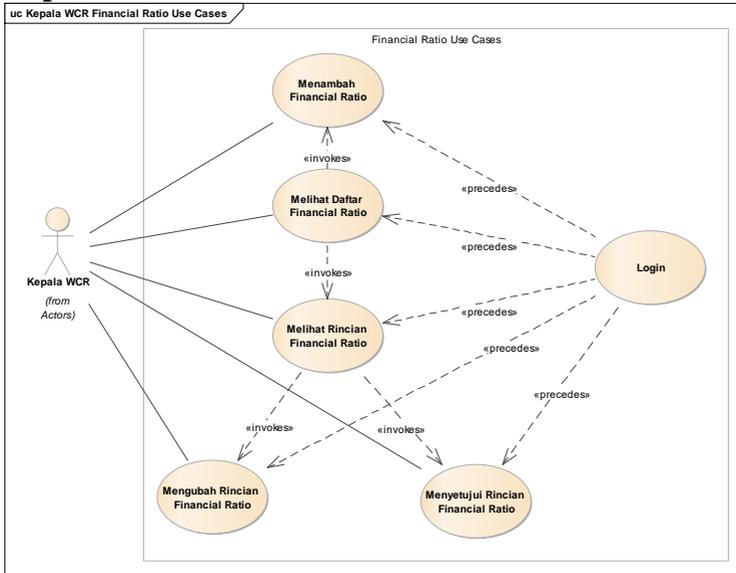
Kepala WCR OTS Use Cases



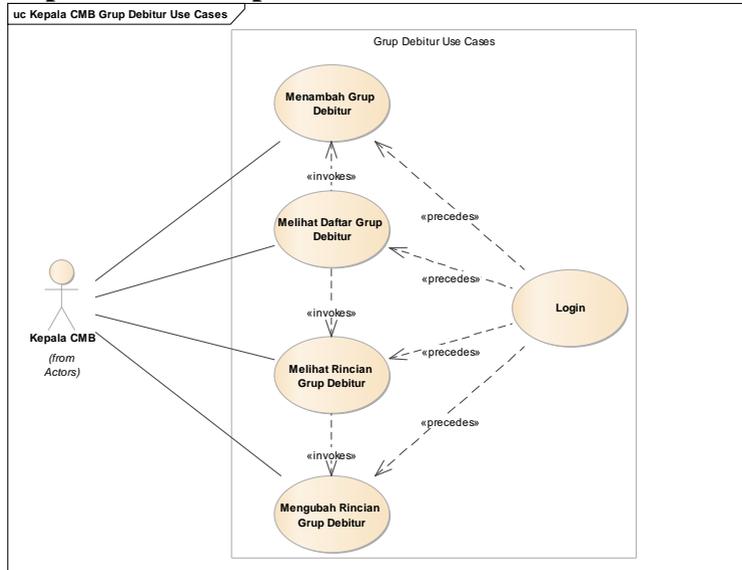
Kepala WCR Covenant Use Cases



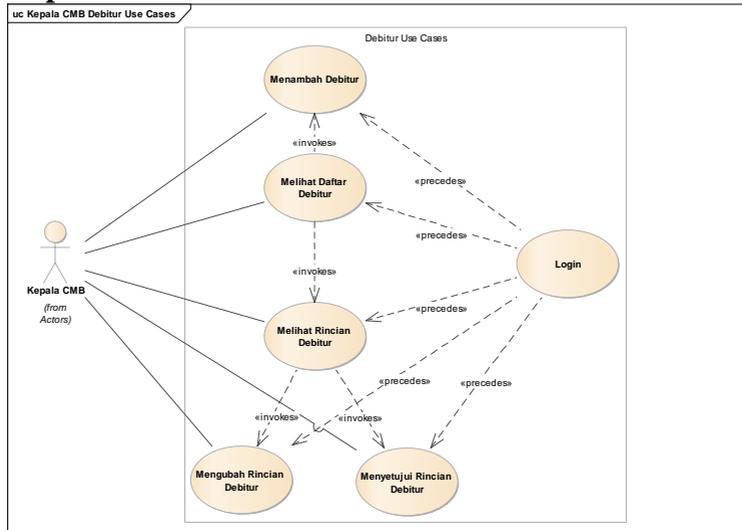
Kepala WCR Financial Ratio



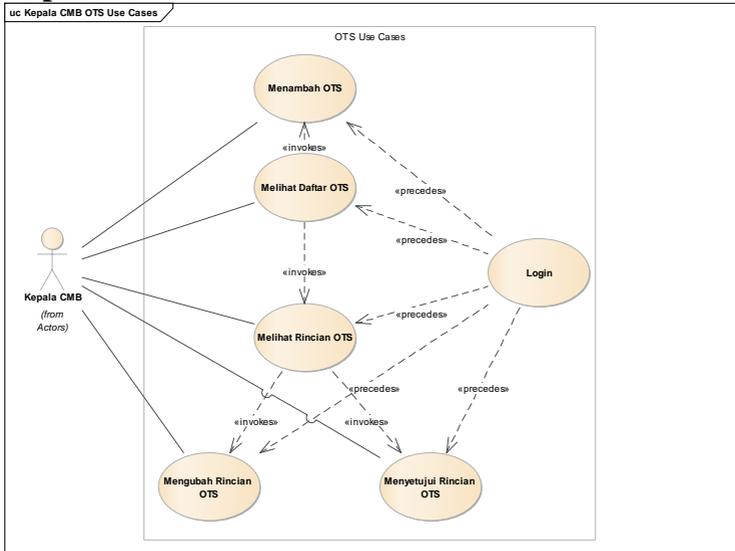
Kepala CMB Grup Debitur Use Cases



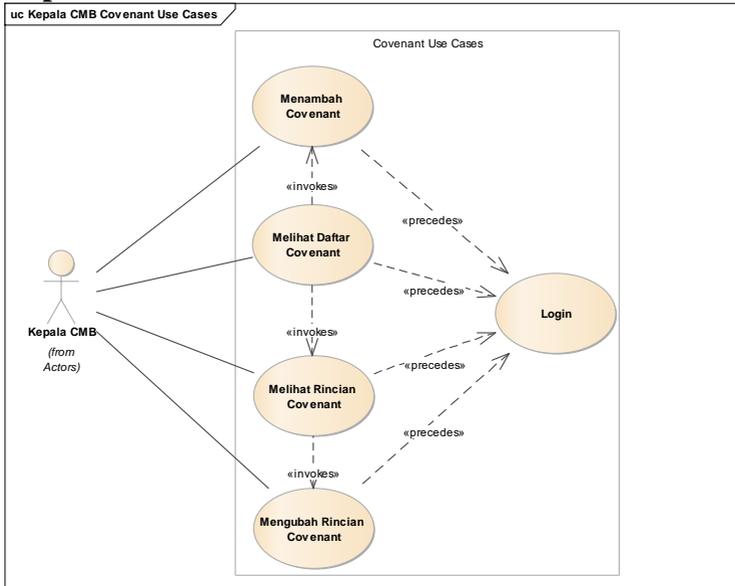
Kepala CMB Debitur Use Cases



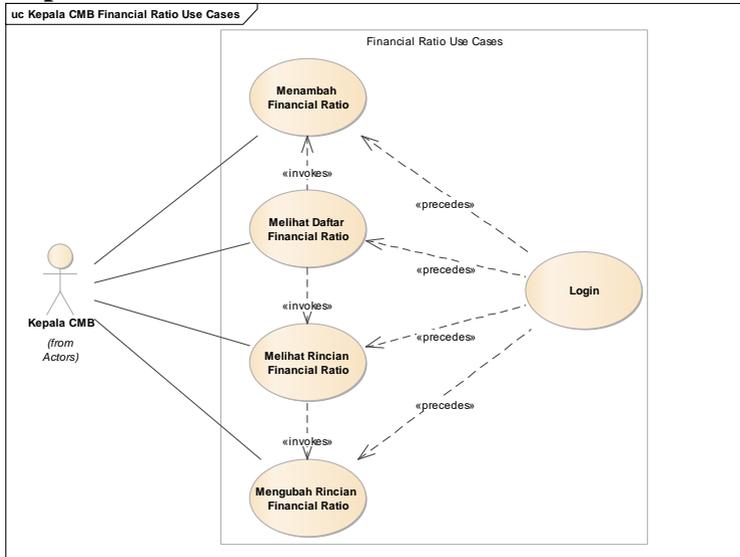
Kepala CMB OTS Use Cases



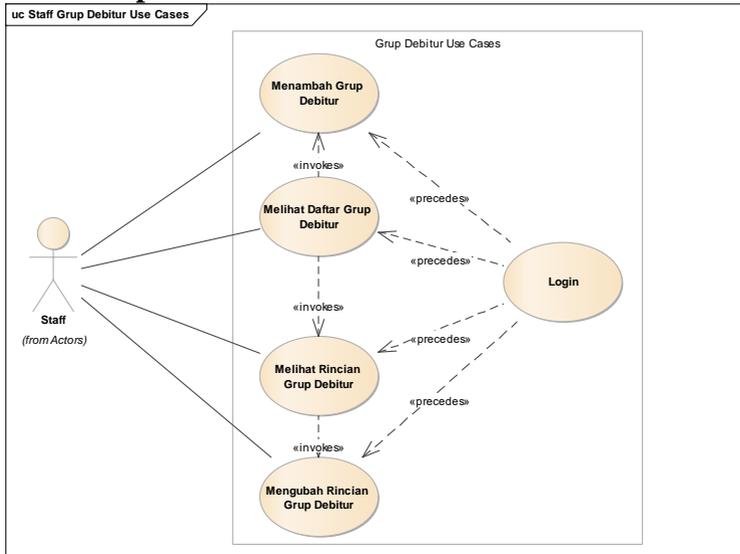
Kepala CMB Covenant Use Cases



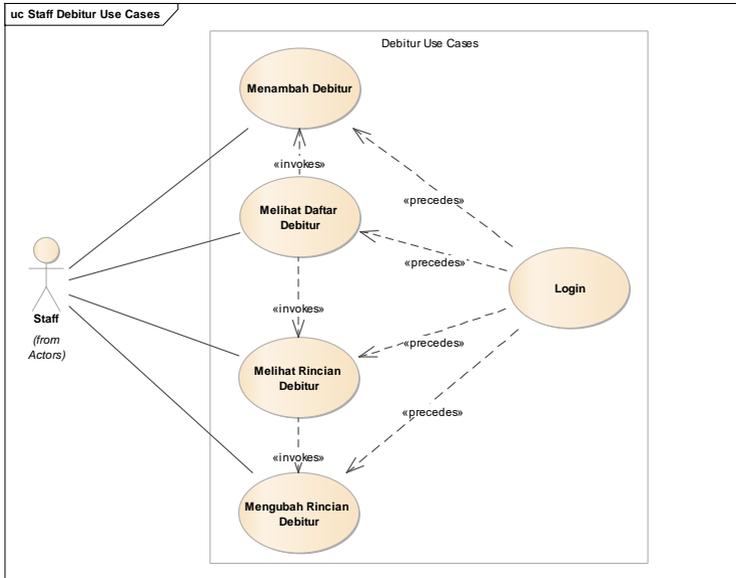
Kepala CMB Financial Ratio Use Cases



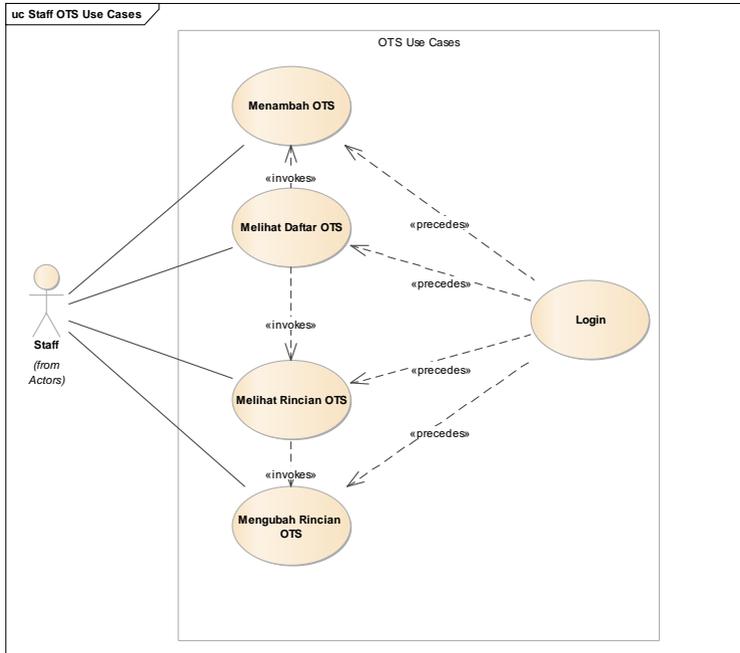
Staff Grup Debitur Use Cases



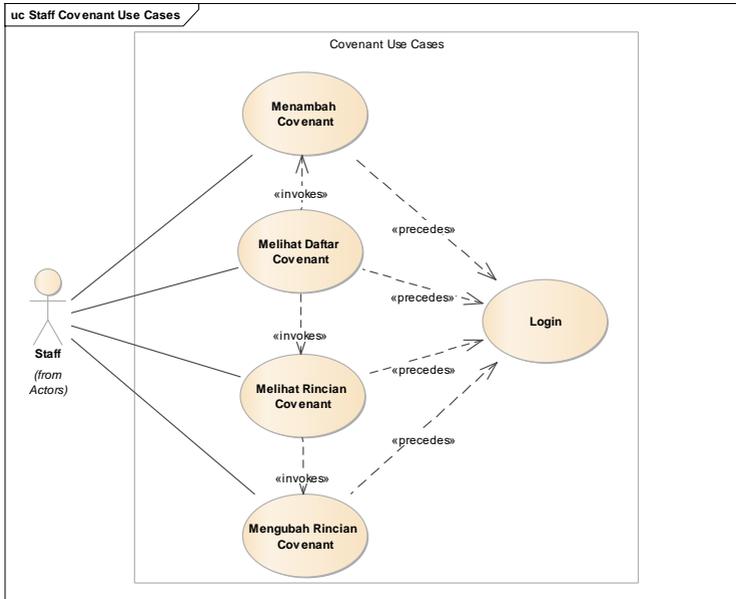
Staff Debitur Use Cases



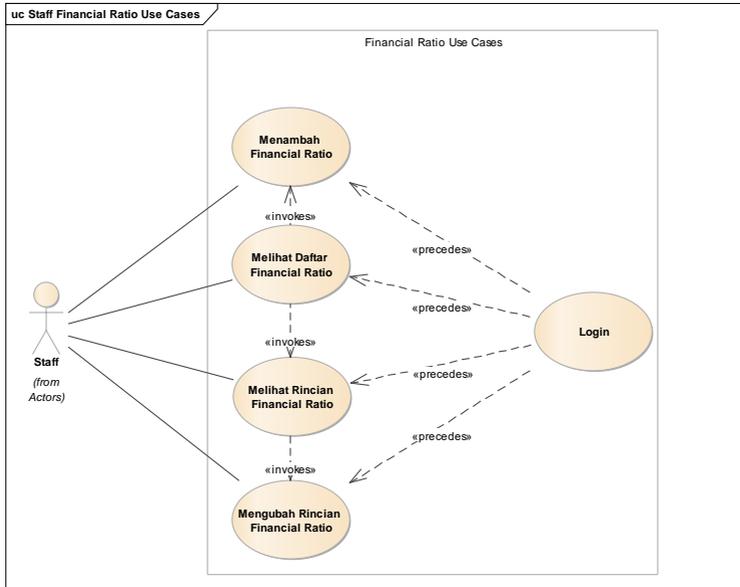
Staff OTS Use Cases



Staff Covenant Use Cases



Staff Financial Ratio Use Cases



B-18

Halaman ini sengaja dikosongkan

LAMPIRAN C USE CASE DESCRIPTION

UC-01 Login

Summary	: Pengguna masuk ke dalam sistem
Priority	: High
Use Frequency	: Always
Direct Actors	: “Admin”, “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. Pengguna memasukkan alamat web 2. Sistem menampilkan laman login
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna membuka halaman login 2. Sistem menampilkan halaman login 3. Pengguna memasukkan username dan password 4. Sistem melakukan pengecekan dan memindahkan pengguna kehalaman beranda.
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Sistem tidak dapat menemukan username dan password yang cocok sehingga menampilkan pesan "username dan/atau password salah" 2. Pengguna melakukan kesalahan login lebih dari tiga kali, sehingga sistem meminta masukan tambahan berupa kode captcha.

UC-02 Logout

Summary	: Pengguna keluar dari dalam sistem
Priority	: High
Use Frequency	: Often
Direct Actors	: “Admin”, “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: 1. Pengguna melakukan <i>Login</i> ke sistem
Main Success Scenario	: 1. Pengguna menekan tombol logout. 2. Sistem menghapus sesi pengguna dan memindahkan pengguna ke halaman login
Alternative Scenario extension	: 1. Sesi pengguna sudah habis, sehingga sistem memindahkan pengguna ke halaman login tanpa menghapus sesi pengguna

UC-03 Melihat Rincian Profil

Summary	: Pengguna ingin melihat rincian profil pribadi
Priority	: High
Use Frequency	: Often
Direct Actors	: “Admin”, “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: 1. Pengguna melakukan <i>Login</i> ke sistem
Main Success Scenario	: 1. Pengguna mengakses halaman rincian profil. 2. Sistem menampilkan rincian profil.
Alternative Scenario extension	: 1. Tidak terdapat rincian profil, sehingga sistem hanya menampilkan halaman kosong.

UC-04 Mengubah Rincian Profil

Summary	: Pengguna ingin mengubah rincian profil pribadi
Priority	: High
Use Frequency	: Often
Direct Actors	: “Admin”, “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: 1. Pengguna melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman profil
Main Success Scenario	: 1. Pengguna mengubah rincian profil, dan menekan "Ubah Rincian Profil" 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Profil"
Alternative Scenario extension	: 1. Tidak terdapat perubahan pada rincian profil sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Profil"

UC-05 Mengganti Password

Summary	: Pengguna ingin mengubah password pribadi
Priority	: High
Use Frequency	: Often
Direct Actors	: “Admin”, “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: 1. Pengguna melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman profil
Main Success Scenario	: 1. Pengguna memasukkan password lama, password baru dan konfirmasi password lalu menekan "Ganti Password". 2. Sistem memastikan password baru dan konfirmasi password baru sama serta melakukan pengecekan password lama pengguna lalu menampilkan pesan "Berhasil memperbarui password"
Alternative Scenario extension	: 1. Password baru dan konfirmasi password baru tidak sama, sehingga sistem menampilkan "Password baru dan konfirmasi password baru harus sama" 2. Password lama salah, sehingga sistem menampilkan "Password lama yang Anda masukkan salah"

UC-06 Melihat Daftar Pengguna

Summary	:	“Administrator” ingin melihat daftar Pengguna.
Priority	:	High
Use Frequency	:	Often
Direct Actors	:	“Administrator” dan “Direksi”
Prereq / Precondition	:	1. “Administrator” sudah <i>Login</i> ke dalam sistem
Main Success Scenario	:	1. Administrator mengakses halaman daftar Pengguna. 2. Sistem menampilkan daftar Pengguna.
Alternative Scenario extension	:	1. Tidak terdapat Pengguna, sehingga sistem hanya menampilkan tabel kosong.

UC-07 Melihat Rincian Pengguna

Summary	: “Administrator” ingin melihat rincian Pengguna.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Administrator” sudah <i>Login</i> ke dalam sistem 2. Pengguna berada di halaman daftar pengguna 3. Terdapat pengguna dalam daftar pengguna
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator mengakses halaman rincian pengguna. 2. Sistem menampilkan rincian pengguna.
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat rincian pengguna, sehingga sistem hanya menampilkan halaman kosong.

UC-08 Menambah Pengguna

Summary	: “Administrator” ingin menambah Pengguna.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Administrator” sudah <i>Login</i> ke dalam sistem 2. Pengguna berada di halaman daftar pengguna
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator mengakses halaman tambah Pengguna. 2. Sistem menampilkan halaman tambah pengguna. 3. Administrator memasukkan rincian pengguna dan menekan "Tambah" 4. Sistem menyimpan Rincian Pengguna dan menampilkan pesan "Berhasil menambahkan Pengguna".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. <i>Username</i> dan/atau <i>e-mail</i> yang ditambahkan sudah ada, sistem menampilkan pesan "Gagal menambahkan Pengguna".

UC-09 Mengubah Rincian Pengguna

Summary	: “Administrator” ingin mengubah rincian Pengguna.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Administrator” sudah <i>Login</i> ke dalam sistem 2. Pengguna berada di halaman rincian pengguna
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator mengubah rincian pengguna, dan menekan "Ubah Rincian Pengguna". 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Pengguna"
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian pengguna sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Pengguna"

UC-10 Melihat Daftar CMB

Summary	:	“Administrator” dan “Direksi” ingin melihat daftar CMB.
Priority	:	High
Use Frequency	:	Often
Direct Actors	:	“Administrator” dan “Direksi”
Prereq / Precondition	:	1. “Administrator” sudah <i>Login</i> ke dalam sistem
Main Success Scenario	:	1. Administrator membuka halaman daftar CMB 2. Sistem menampilkan daftar CMB.
Alternative Scenario extension	:	1. Belum terdapat CMB, sehingga sistem hanya menampilkan tabel kosong.

UC-11 Melihat Rincian CMB

Summary	: “Administrator” ingin melihat rincian CMB.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Administrator” sudah <i>Login</i> ke dalam sistem 2. Pengguna berada di halaman daftar rincian CMB 3. Terdapat CMB dalam daftar CMB
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator mengakses halaman rincian CMB. 2. Sistem menampilkan rincian CMB.
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat rincian CMB, sehingga sistem hanya menampilkan halaman kosong.

UC-12 Menambah CMB

Summary	: “Administrator” ingin menambah CMB.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Administrator” sudah <i>Login</i> ke dalam sistem 2. Pengguna berada di halaman daftar CMB
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator mengakses halaman tambah CMB. 2. Sistem menampilkan halaman tambah CMB. 3. Administrator memasukkan rincian pengguna dan menekan "Tambah". 4. Sistem menyimpan Rincian CMB dan menampilkan pesan "Berhasil menambahkan CMB".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. CMB yang ditambahkan sudah ada, sistem menampilkan pesan "Gagal menambahkan CMB".

UC-13 Mengubah Rincian CMB

Summary	: “Administrator” ingin mengubah rincian CMB.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Administrator” sudah <i>Login</i> ke dalam sistem 2. Pengguna berada di halaman rincian CMB
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator mengubah rincian CMB, dan menekan "Ubah Rincian CMB". 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian CMB"
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian CMB sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian CMB"

UC-14 Manajemen konfigurasi sistem

Summary	: “Administrator” ingin mengubah konfigurasi sistem.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Administrator”
Prereq / Precondition	: 1. “Administrator” sudah <i>Login</i> ke dalam sistem
Main Success Scenario	: <ol style="list-style-type: none"> 1. Administrator membuka halaman konfigurasi 2. Sistem menampilkan halaman konfigurasi. 3. Administrator memasukkan konfigurasi dan menekan simpan 4. Sistem melakukan penyimpanan konfigurasi dan menampilkan pesan "Berhasil menyimpan konfigurasi sistem"
Alternative Scenario extension	: 1. Tidak terdapat perubahan pada konfigurasi, sistem menampilkan "Tidak terdapat perubahan pada konfigurasi"

UC-15 Melihat Daftar Grup Debitur

Summary	: “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat daftar grup debitur.
Priority	: High
Use	: Often
Frequency	
Direct Actors	: “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem.
Main Success Scenario	: 1. Penggu Pengguna mengakses halaman daftar grup debitur. 2. Sistem menampilkan daftar grup debitur.
Alternative Scenario extension	: 1. Tidak terdapat grup debitur, sehingga sistem hanya menampilkan tabel kosong.

UC-16 Melihat Rincian Grup Debitur

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat rincian grup debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar grup debitur 3. Terdapat grup debitur dalam daftar grup debitur
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengakses halaman rincian grup debitur. 2. Sistem menampilkan rincian grup debitur.
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat rincian grup debitur, sehingga sistem hanya menampilkan halaman kosong.

UC-17 Menambah Grup Debitur

Summary	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin menambah grup debitur.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar grup debitur.
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengakses halaman tambah grup debitur 2. Sistem menampilkan halaman tambah grup debitur. 3. Pengguna memasukkan rincian grup debitur dan menekan "Tambah" 4. Sistem menyimpan rincian grup debitur dan menampilkan pesan "Berhasil menambahkan Grup Debitur".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Nama grup debitur yang dimasukkan sudah ada, sehingga sistem menampilkan pesan "Gagal menambahkan Grup Debitur".

UC-18 Mengubah Rincian Grup Debitur

Summary	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin mengubah rincian debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian grup debitur
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengubah rincian grup debitur, dan menekan "Ubah Rincian Grup Debitur". 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Grup Debitur"
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian grup debitur sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Grup Debitur"

UC-19 Melihat Daftar Debitur

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat daftar debitur.
Priority	:	High
Use Frequency	:	Often
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem
Main Success Scenario	:	1. Pengguna mengakses halaman daftar debitur. 2. Sistem menampilkan daftar debitur.
Alternative Scenario extension	:	1. Tidak terdapat debitur, sehingga sistem hanya menampilkan tabel kosong.

UC-20 Melihat Rincian Debitur

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat rincian debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar debitur 3. Terdapat debitur dalam daftar debitur
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengakses halaman rincian debitur. 2. Sistem menampilkan rincian debitur.
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat rincian debitur, sehingga sistem hanya menampilkan halaman kosong.

UC-21 Menambah Debitur

Summary	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin menambah debitur.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar debitur
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengakses halaman tambah debitur 2. Sistem menampilkan halaman tambah debitur. 3. Pengguna memasukkan rincian debitur dan menekan "Tambah" 4. Sistem menyimpan rincian debitur dan menampilkan pesan "Berhasil menambahkan Debitur".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Debitur yang dimasukkan sudah ada, sehingga sistem menampilkan pesan "Gagal menambahkan debitur".

UC-22 Mengubah Rincian Debitur

Summary	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin mengubah rincian debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian debitur
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengubah rincian debitur, dan menekan "Ubah Rincian Debitur" 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Debitur"
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian debitur sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Debitur"

UC-23 Menyetujui Debitur

Summary	: “Kepala WCR”, “Kepala CMB” melakukan persetujuan terkait data debitur yang baru saja ditambahkan dan atau diubah bahwa data telah valid atau benar.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”, “Kepala CMB”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian debitur 3. Debitur belum disetujui
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna menekan "Setujui Debitur". 2. Sistem mengubah status debitur menjadi di setujui dan menampilkan "Berhasil Menyetujui Debitur".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Debitur sudah disetujui sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Debitur"

UC-24 Melihat Daftar OTS

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat data OTS dari debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian debitur
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengakses halaman daftar OTS. 2. Sistem menampilkan daftar OTS.
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat OTS, sehingga sistem hanya menampilkan tabel kosong.

UC-25 Melihat Rincian OTS

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat data OTS dari debitur.
Priority	:	High
Use Frequency	:	Often
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar OTS 3. Terdapat OTS dalam daftar OTS
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengakses halaman rincian OTS. 2. Sistem menampilkan rincian OTS.
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat rincian OTS, sehingga sistem hanya menampilkan halaman kosong.

UC-26 Menambah OTS

Summary	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin menambah data OTS dari debitur.
Priority	: High
Use	: Often
Frequency	
Direct Actors	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar OTS
Main Success Scenario	: 1. Pengguna mengakses halaman tambah OTS 2. Sistem menampilkan halaman tambah OTS. 3. Pengguna memasukkan rincian OTS dan menekan "Tambah". 4. Sistem menyimpan rincian OTS dan menampilkan pesan "Berhasil menambahkan OTS".
Alternative Scenario extension	: 1. Tanggal OTS yang dimasukkan sudah ada, sehingga sistem menampilkan pesan "Gagal menambahkan OTS".

UC-27 Mengubah Rincian OTS

Summary	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin mengubah data OTS dari debitur.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian OTS
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengubah rincian OTS, dan menekan "Ubah Rincian OTS". 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian OTS"
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian OTS sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian OTS"

UC-28 Menyetujui OTS

Summary	: “Kepala WCR”, “Kepala CMB” melakukan persetujuan terkait data OTS yang baru saja ditambahkan dan atau diubah bahwa data telah valid (benar)
Priority	: High
Use	: Often
Frequency	
Direct Actors	: “Kepala WCR”, “Kepala CMB”
Prereq / Precondition	: 1. “Kepala WCR”, “Kepala CMB” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian OTS
Main Success Scenario	: 1. Pengguna menekan "Setujui OTS". 2. Sistem mengubah status OTS menjadi di setujui dan menampilkan "Berhasil Menyetujui OTS".
Alternative Scenario extension	: 1. OTS sudah disetujui sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian OTS"

UC-29 Melihat Daftar Covenant

Summary	: “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat data <i>covenant</i> dari debitur.
Priority	: High
Use	: Often
Frequency	
Direct Actors	: “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian debitur
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengakses halaman daftar covenant. 2. Sistem menampilkan daftar covenant.
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat covenant, sehingga sistem hanya menampilkan tabel kosong.

UC-30 Melihat Rincian Covenant

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat data <i>covenant</i> dari debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar covenant 3. Terdapat covenant dalam daftar covenant
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengakses halaman rincian covenant. 2. Sistem menampilkan rincian covenant.
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat rincian covenant, sehingga sistem hanya menampilkan halaman kosong.

UC-31 Menambah Covenant

Summary	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin menambah data <i>covenant</i> dari debitur.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar covenant
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengakses halaman tambah covenant, 2. Sistem menampilkan halaman tambah covenant. 3. Pengguna memasukkan rincian covenant dan menekan "Tambah" 4. Sistem menyimpan rincian covenant dan menampilkan pesan "Berhasil menambahkan Covenant".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Covenant yang dimasukkan sudah ada, sehingga sistem menampilkan pesan "Gagal menambahkan Covenant".

UC-32 Mengubah Rincian Covenant

Summary	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin mengubah data <i>covenant</i> dari debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian covenant
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengubah rincian covenant, dan menekan "Ubah Rincian Covenant". 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Covenant"
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian covenant sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Covenant"

UC-33 Mengkonfirmasi Rincian Covenant

Summary	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM” mengkonfirmasi data <i>covenant</i> yang menandakan bahwa <i>covenant</i> telah diterima dan bisa dianggap selesai
Priority	:	High
Use Frequency	:	Often
Direct Actors	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian <i>covenant</i> 3. <i>Covenant</i> belum dikonfirmasi
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna menekan "Konfirmasi <i>Covenant</i>". 2. Sistem mengubah status <i>covenant</i> menjadi di setujui dan menampilkan "Berhasil Konfirmasi <i>Covenant</i>".
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. <i>Covenant</i> sudah disetujui sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian <i>Covenant</i>"

UC-34 Menyetujui Covenant

Summary	:	“Kepala WCR” menyetujui data <i>covenant</i> yang menandakan bahwa <i>covenant</i> telah benar-benar diterima dan telah sesuai dengan kesepakatan dengan pihak Mandiri
Priority	:	High
Use Frequency	:	Often
Direct Actors	:	“Kepala WCR”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian <i>covenant</i> 3. <i>Covenant</i> belum disetujui
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna menekan "Setujui <i>Covenant</i>". 2. Sistem mengubah status <i>covenant</i> menjadi di setujui dan menampilkan "Berhasil Menyetujui <i>Covenant</i>".
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. <i>Covenant</i> sudah disetujui sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian <i>Covenant</i>"

UC-35 Melihat Daftar Financial Ratio

Summary	: “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat data <i>financial ratio</i> dari debitur.
Priority	: High
Use	: Often
Frequency	
Direct Actors	: “Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian debitur
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengakses halaman daftar financial ratio. 2. Sistem menampilkan daftar financial ratio.
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Tidak terdapat financial ratio, sehingga sistem hanya menampilkan tabel kosong.

UC-36 Melihat Rincian Financial Ratio

Summary	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin melihat data <i>financial ratio</i> dari debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Direksi”, “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar <i>financial ratio</i> 3. Terdapat <i>financial ratio</i> dalam daftar <i>financial ratio</i>
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengakses halaman rincian <i>financial ratio</i>. 2. Sistem menampilkan rincian <i>financial ratio</i>.
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat rincian <i>financial ratio</i>, sehingga sistem hanya menampilkan halaman kosong.

UC-37 Menambah Financial Ratio

Summary	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin menambah data <i>financial ratio</i> dari debitur.
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman daftar <i>financial ratio</i>
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna mengakses halaman tambah financial rasio 2. Sistem menampilkan halaman tambah financial ratio. 3. Pengguna memasukkan rincian financial ratio dan menekan "Tambah" 4. Sistem menyimpan rincian financial ratio dan menampilkan pesan "Berhasil menambahkan Financial Ratio".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Bulan dan tahun yang dimasukkan sudah ada, sehingga sistem menampilkan pesan "Gagal menambahkan Financial Ratio".

UC-38 Mengubah Rincian Financial Ratio

Summary	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM” ingin mengubah data <i>financial ratio</i> dari debitur.
Priority	:	High
Use	:	Often
Frequency		
Direct Actors	:	“Kepala WCR”, “Kepala CMB” dan “RM/CRM”
Prereq / Precondition	:	<ol style="list-style-type: none"> 1. “Kepala WCR”, “Kepala CMB” dan “RM/CRM” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman rincian <i>financial ratio</i>
Main Success Scenario	:	<ol style="list-style-type: none"> 1. Pengguna mengubah rincian <i>financial ratio</i>, dan menekan "Ubah Rincian Financial Ratio". 2. Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Financial Ratio"
Alternative Scenario extension	:	<ol style="list-style-type: none"> 1. Tidak terdapat perubahan pada rincian <i>financial ratio</i> sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Financial Ratio"

UC-39 Menyetujui Financial Ratio

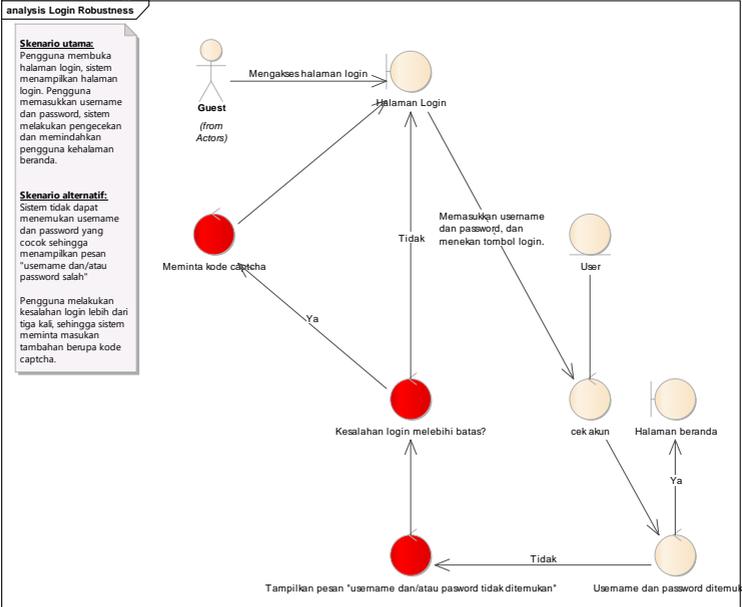
Summary	: “Kepala WCR” melakukan persetujuan terkait data <i>financial ratio</i> yang baru saja ditambahkan dan atau diubah bahwa data telah valid (benar)
Priority	: High
Use Frequency	: Often
Direct Actors	: “Kepala WCR”
Prereq / Precondition	: <ol style="list-style-type: none"> 1. “Kepala WCR” melakukan <i>Login</i> ke sistem 2. Pengguna berada di halaman <i>financial rasio</i> 3. <i>Financial ratio</i> belum disetujui
Main Success Scenario	: <ol style="list-style-type: none"> 1. Pengguna menekan "Setujui Financial Ratio". 2. Sistem mengubah status OTS menjadi di setujui dan menampilkan "Berhasil Menyetujui Financial Ratio".
Alternative Scenario extension	: <ol style="list-style-type: none"> 1. Financial ratio sudah disetujui sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Financial Ratio"

C-40

Halaman ini sengaja dikosongkan

LAMPIRAN D ROBUSTNESS DIAGRAM

UC-01 Login

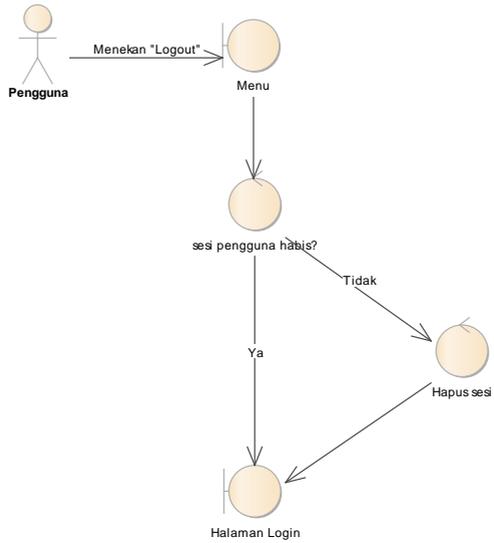


UC-02 Logout

analysis Logout Robustness

Skenario utama:
Pengguna menekan tombol logout. Sistem menghapus sesi pengguna dan memindahkan pengguna ke halaman login

Skenario alternatif:
Sesi pengguna sudah habis
Sistem memindahkan pengguna ke halaman login tanpa menghapus sesi pengguna

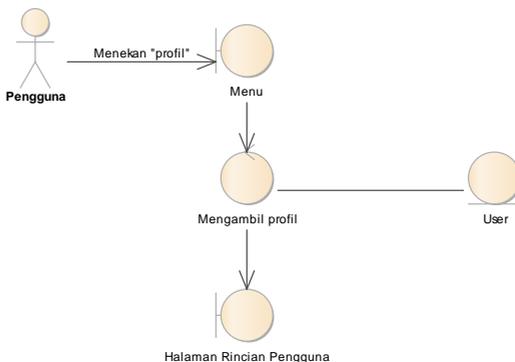


UC-03 Melihat Rincian Profil

analysis Melihat rincian profil Robustness

Skenario utama:
Pengguna mengakses halaman rincian profil. Sistem menampilkan rincian profil.

Skenario alternatif:
Tidak terdapat rincian profil, sehingga sistem hanya menampilkan halaman kosong.

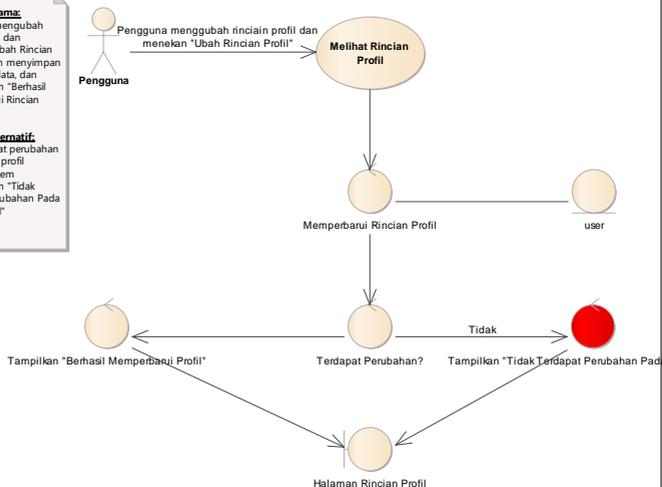


UC-04 Mengubah Rincian Profil

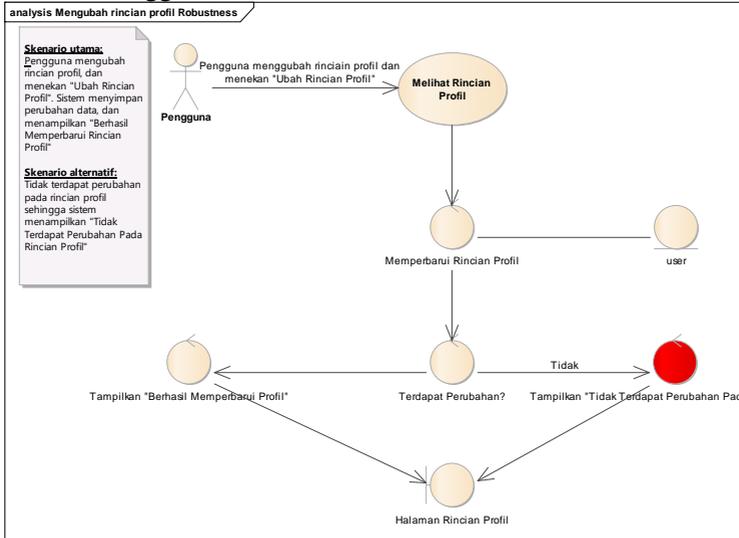
analysis Mengubah rincian profil Robustness

Skenario utama:
Pengguna mengubah rincian profil, dan menekan "Ubah Rincian Profil". Sistem menyimpan perubahan data, dan menampilkan "Berhasil Memperbarui Rincian Profil".

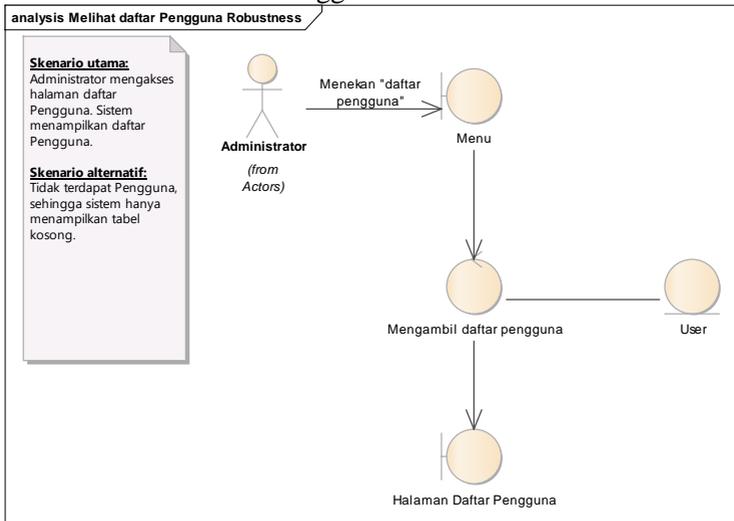
Skenario alternatif:
Tidak terdapat perubahan pada rincian profil sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Profil".



UC-05 Mengganti Password



UC-06 Melihat Daftar Pengguna

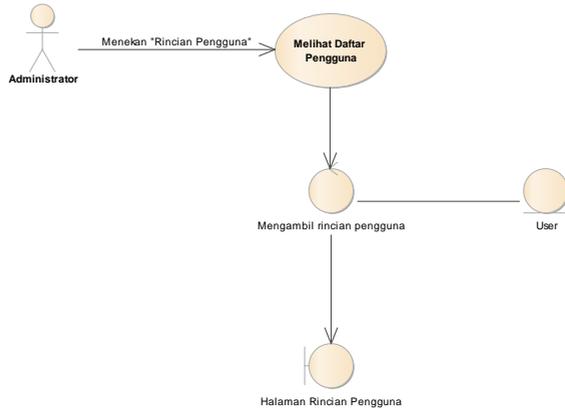


UC-07 Melihat Rincian Pengguna

analisis Melihat rincian pengguna Robustness

Skenario utama:
Administrator mengakses halaman rincian pengguna. Sistem menampilkan rincian pengguna.

Skenario alternatif:
Tidak terdapat rincian pengguna, sehingga sistem hanya menampilkan halaman kosong.

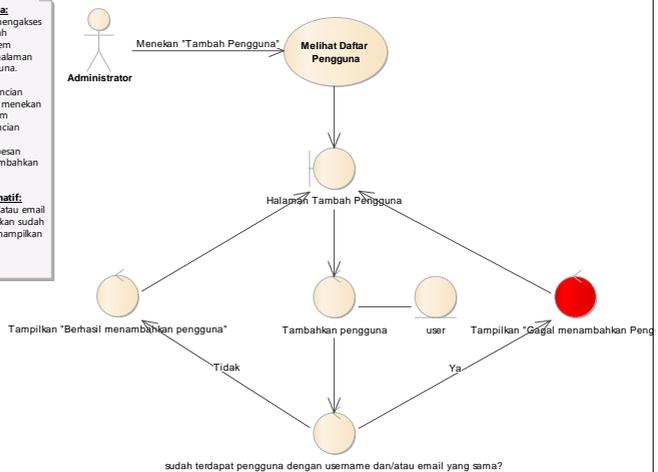


UC-08 Menambah Pengguna

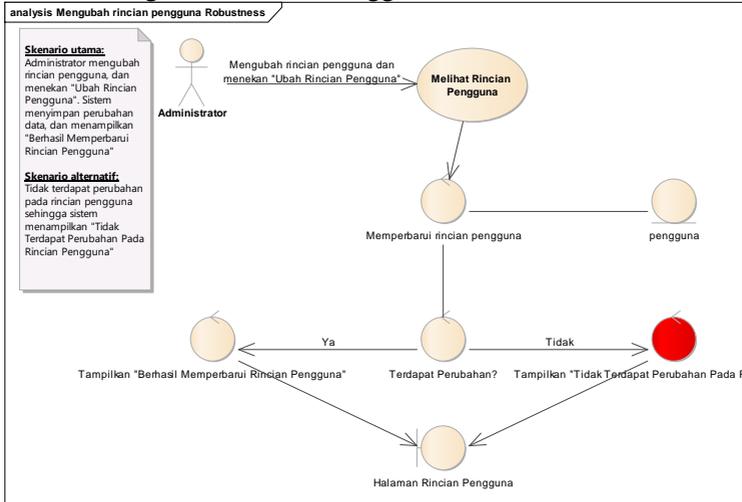
analisis Menambah pengguna Robustness

Skenario utama:
Administrator mengakses halaman tambah Pengguna. Sistem menampilkan halaman tambah pengguna. Administrator memasukkan rincian pengguna dan menekan "Tambah", sistem menyimpan Rincian Pengguna dan menampilkan pesan "Berhasil menambahkan Pengguna".

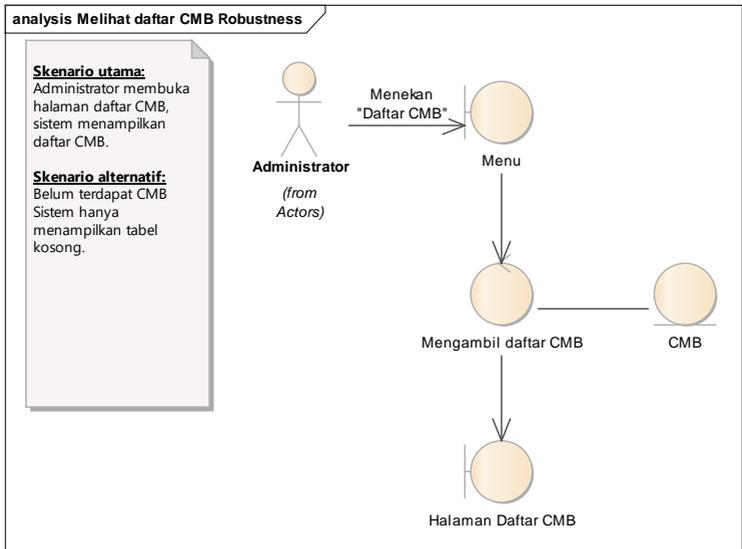
Skenario alternatif:
Username dan/atau email yang ditambahkan sudah ada, sistem menampilkan pesan "Gagal menambahkan Pengguna".



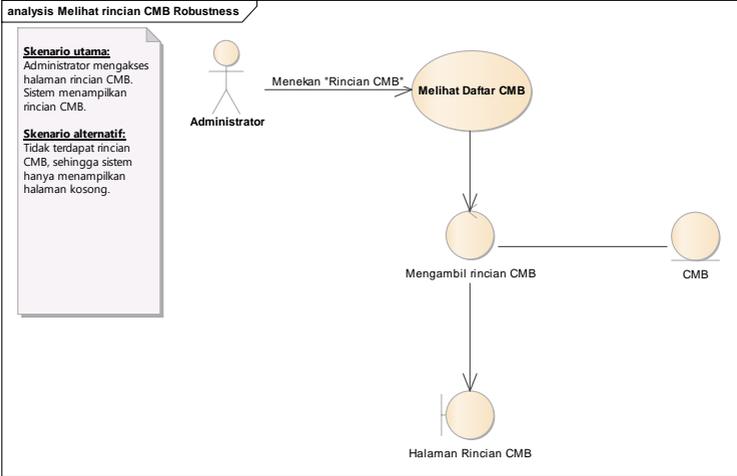
UC-09 Mengubah Rincian Pengguna



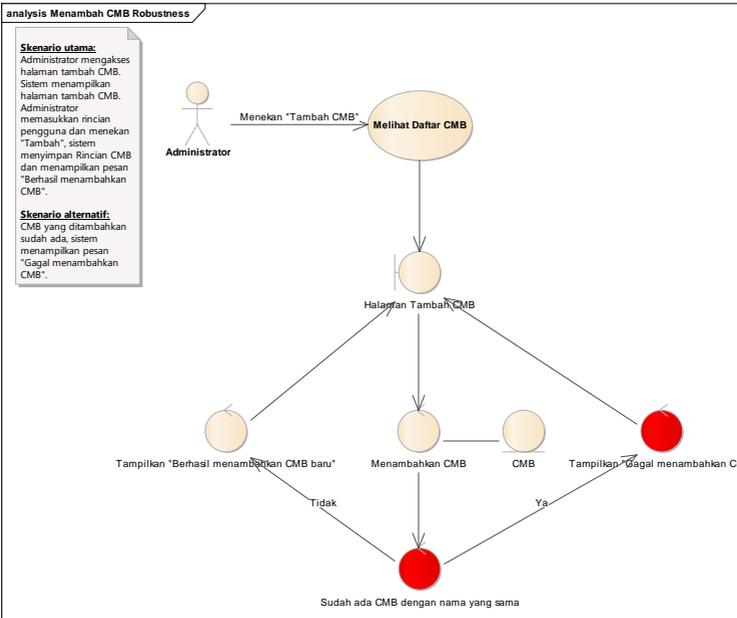
UC-10 Melihat Daftar CMB



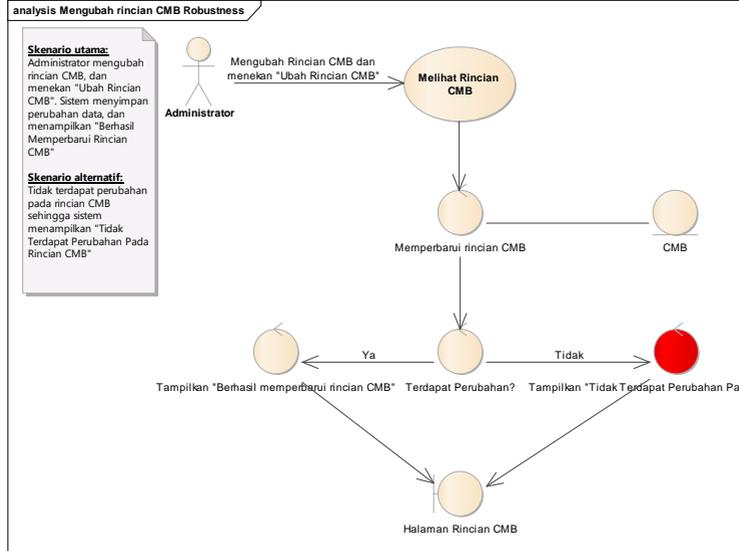
UC-11 Melihat Rincian CMB



UC-12 Menambah CMB



UC-13 Mengubah Rincian CMB



UC-14 Manajemen konfigurasi sistem

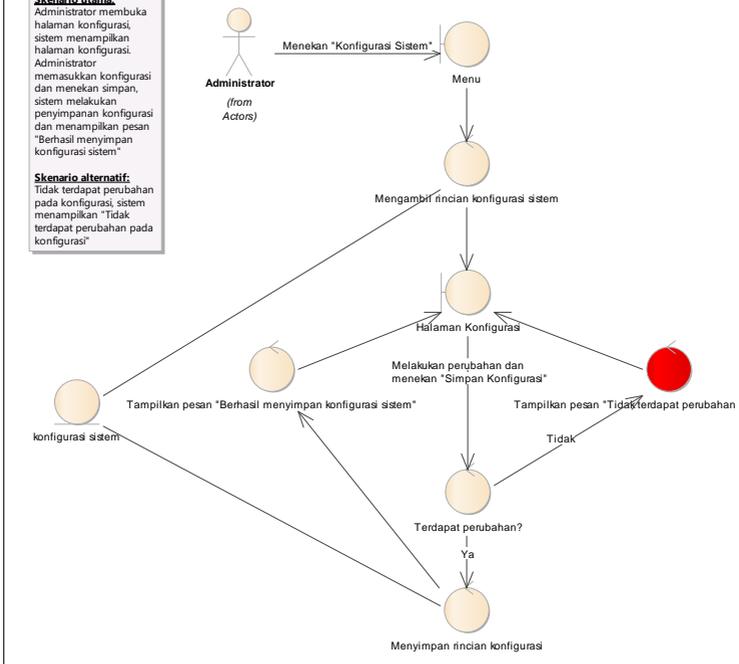
analisis Manajemen konfigurasi sistem Robustness

Skenario utama:

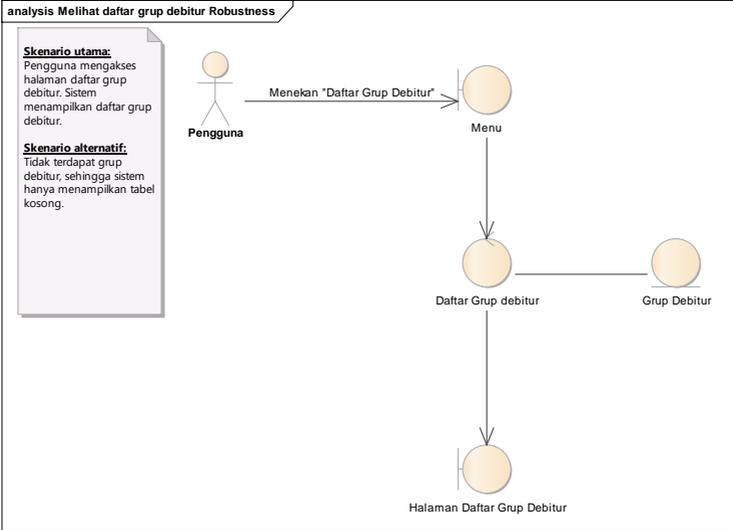
Administrator membuka halaman konfigurasi, sistem menampilkan halaman konfigurasi. Administrator memasukkan konfigurasi dan menekan simpan, sistem melakukan penyimpanan konfigurasi dan menampilkan pesan "Berhasil menyimpan konfigurasi sistem"

Skenario alternatif:

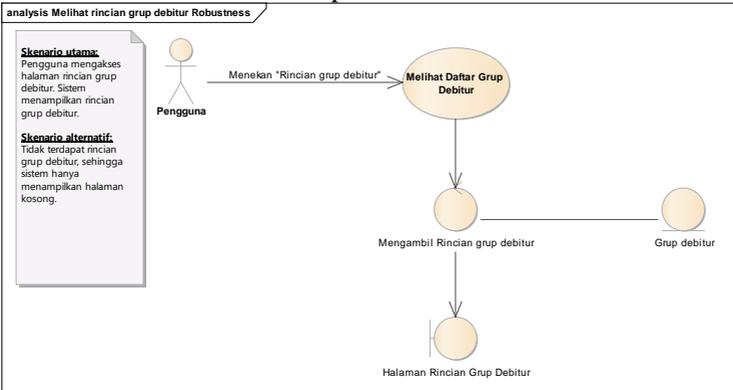
Tidak terdapat perubahan pada konfigurasi, sistem menampilkan "Tidak terdapat perubahan pada konfigurasi"



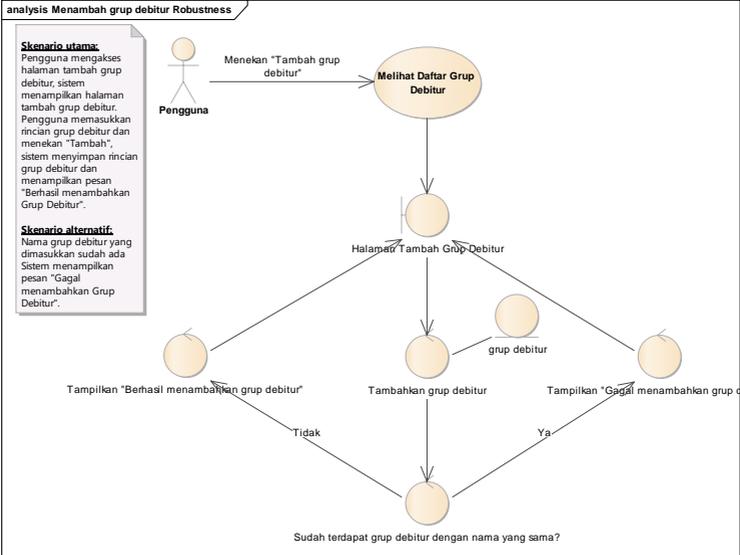
UC-15 Melihat Daftar Grup Debitur



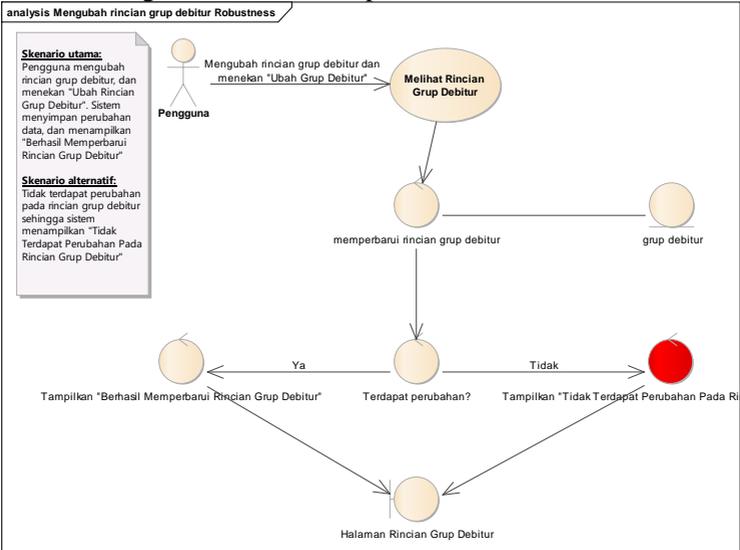
UC-16 Melihat Rincian Grup Debitur



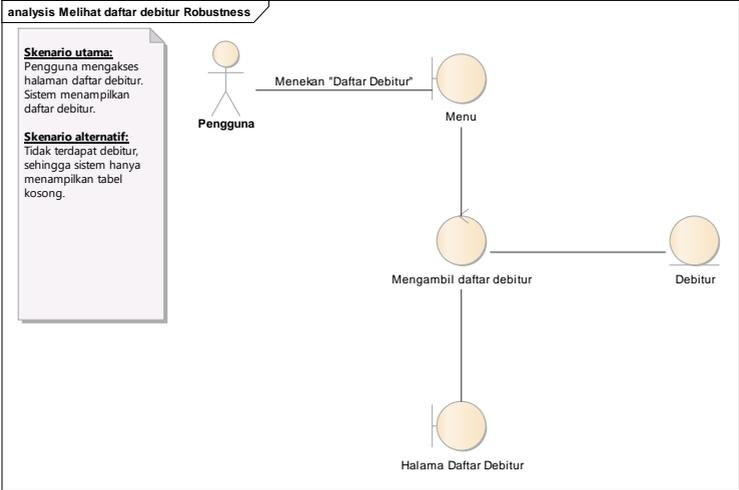
UC-17 Menambah Grup Debitur



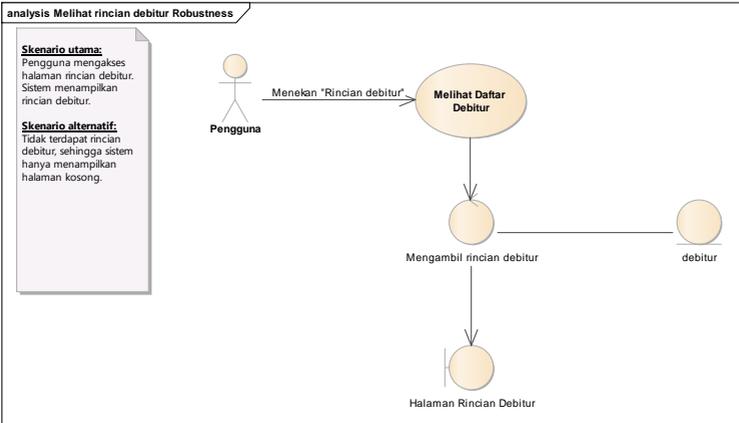
UC-18 Mengubah Rincian Grup Debitur



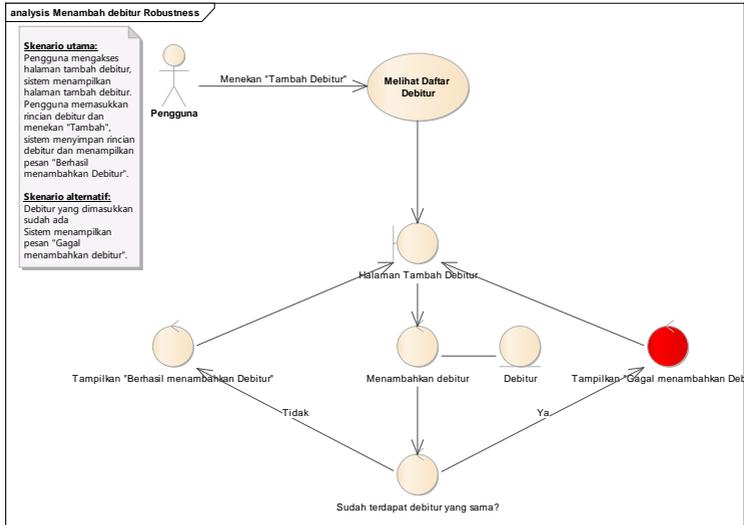
UC-19 Melihat Daftar Debitur



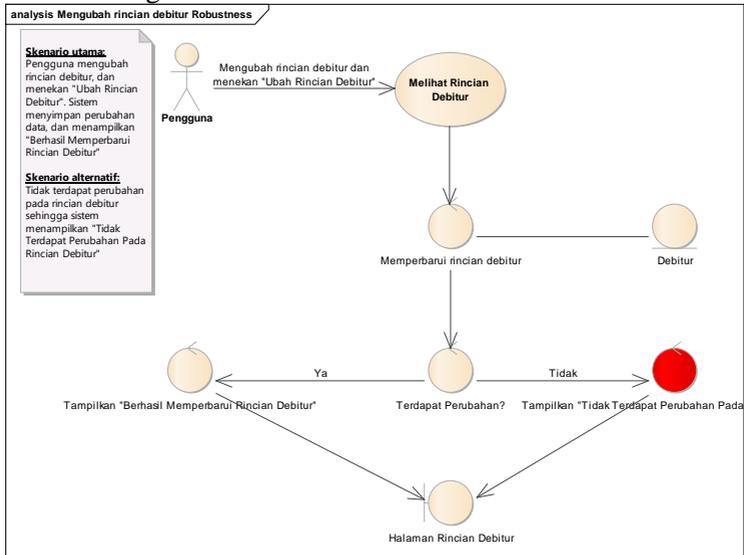
UC-20 Melihat Rincian Debitur



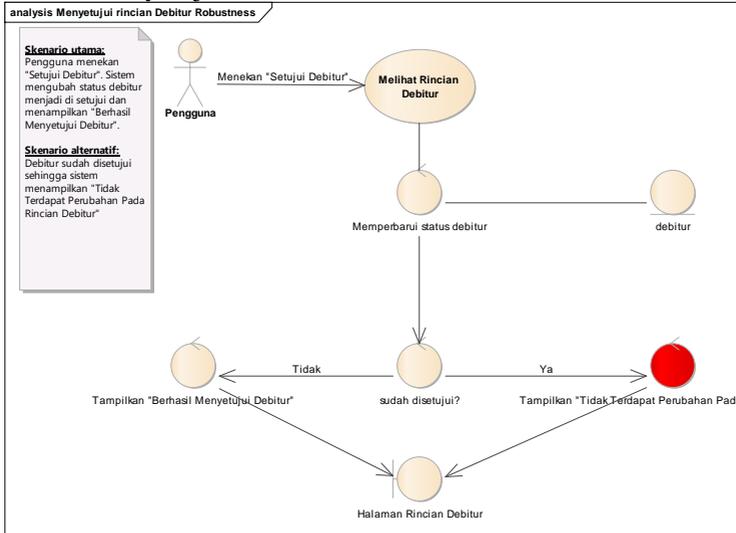
UC-21 Menambah Debitur



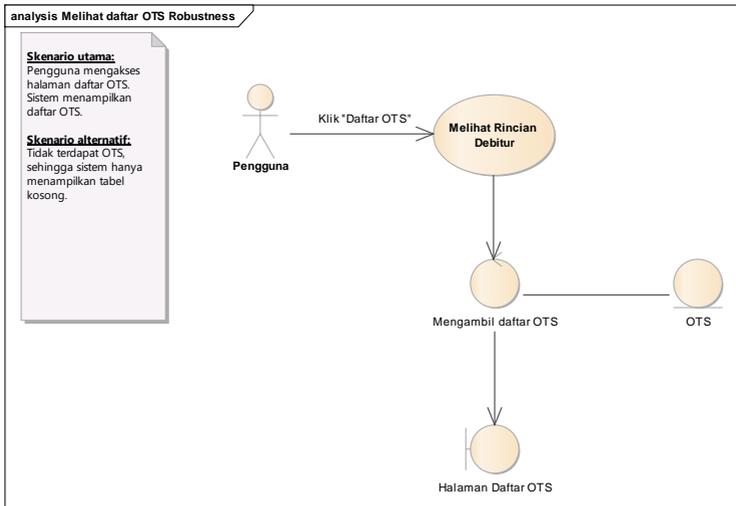
UC-22 Mengubah Rincian Debitur



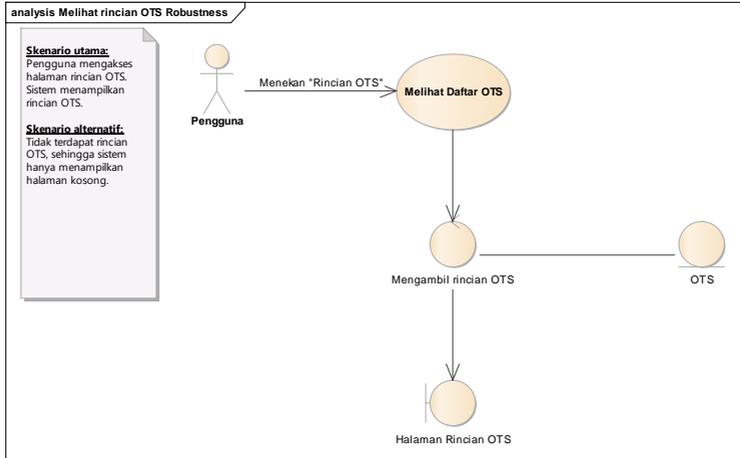
UC-23 Menyetujui Debitur



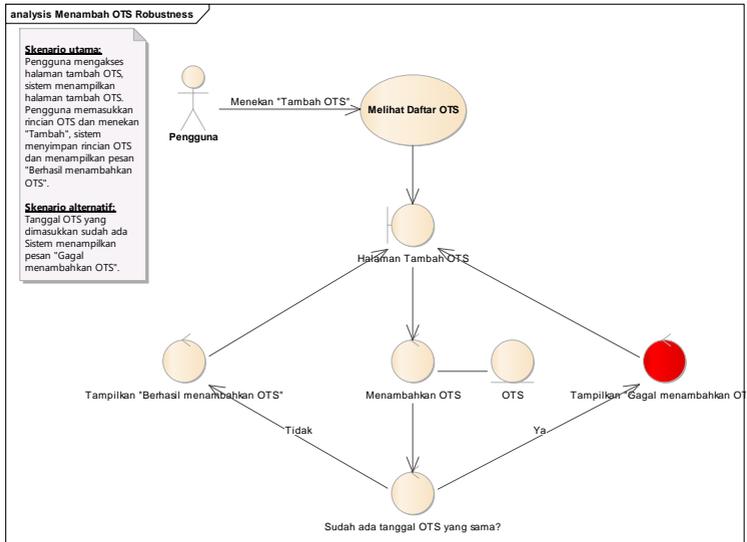
UC-24 Melihat Daftar OTS



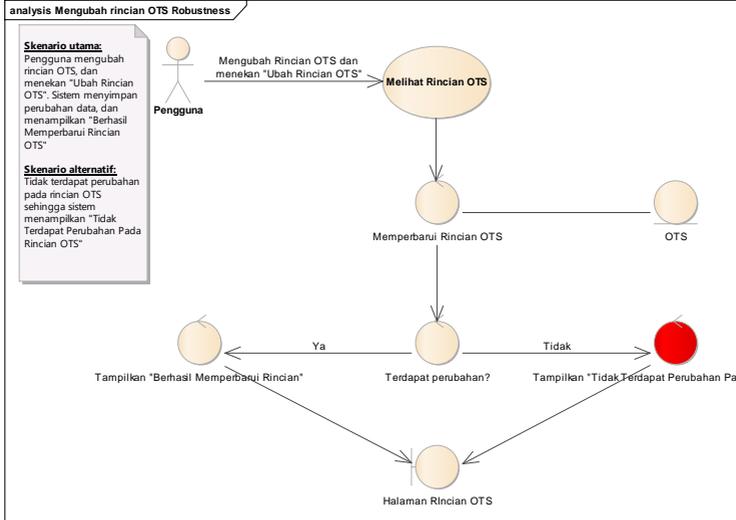
UC-25 Melihat Rincian OTS



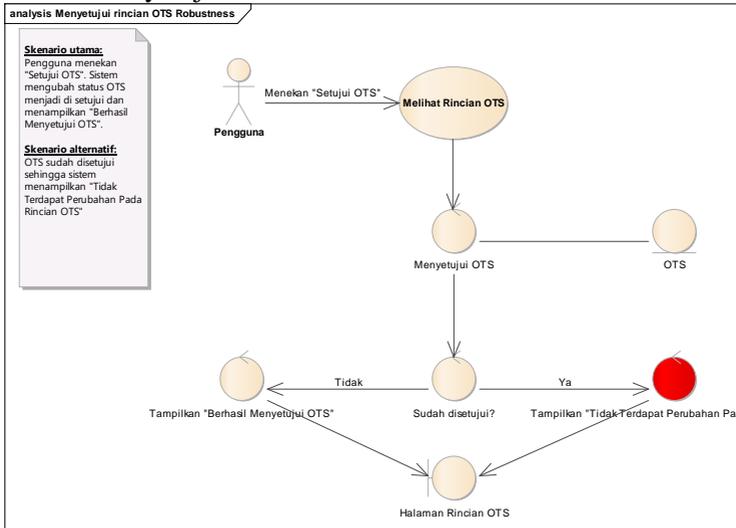
UC-26 Menambah OTS



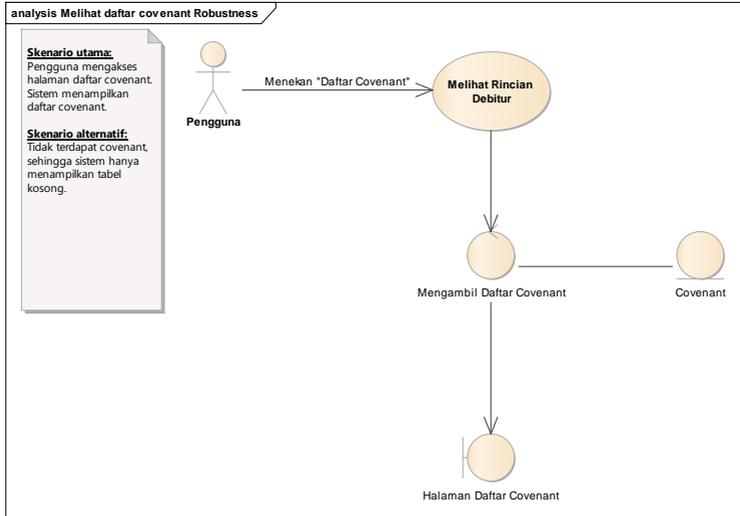
UC-27 Mengubah Rincian OTS



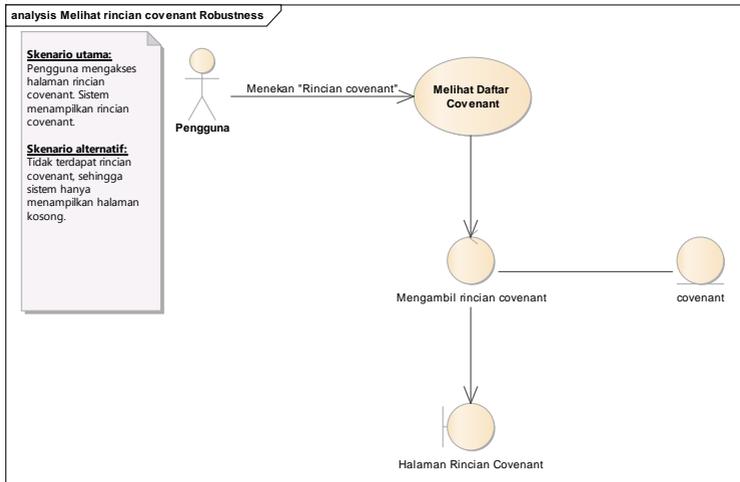
UC-28 Menyetujui OTS



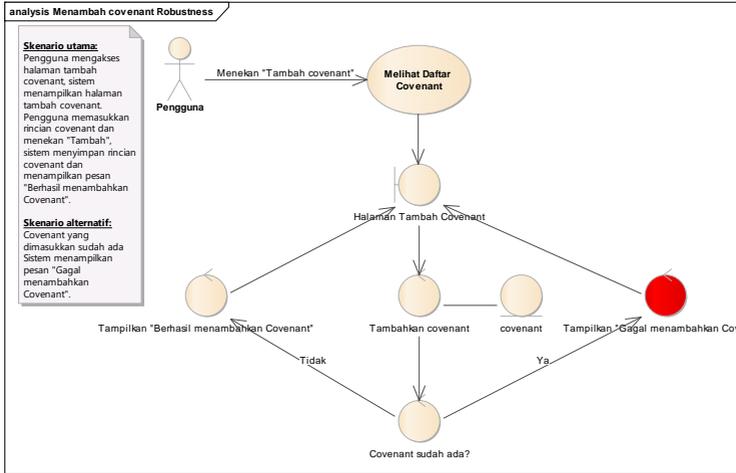
UC-29 Melihat Daftar Covenant



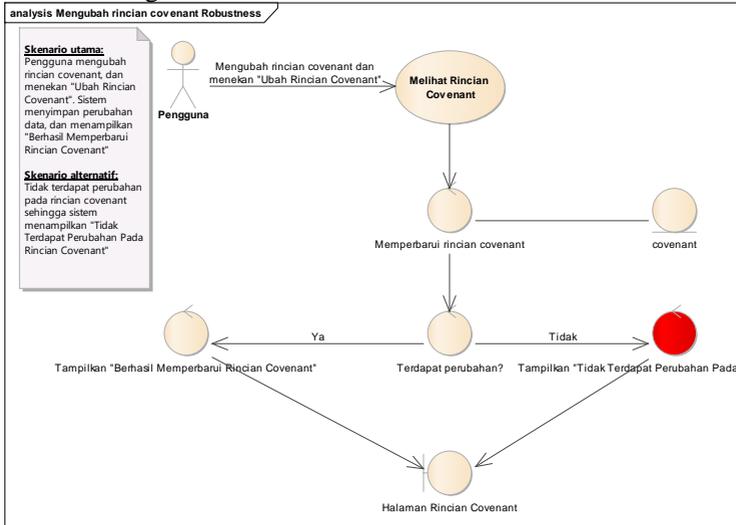
UC-30 Melihat Rincian Covenant



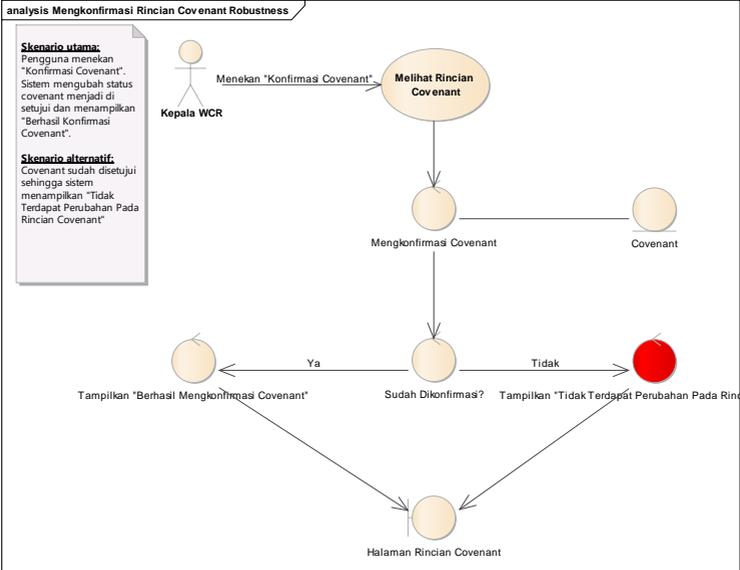
UC-31 Menambah Covenant



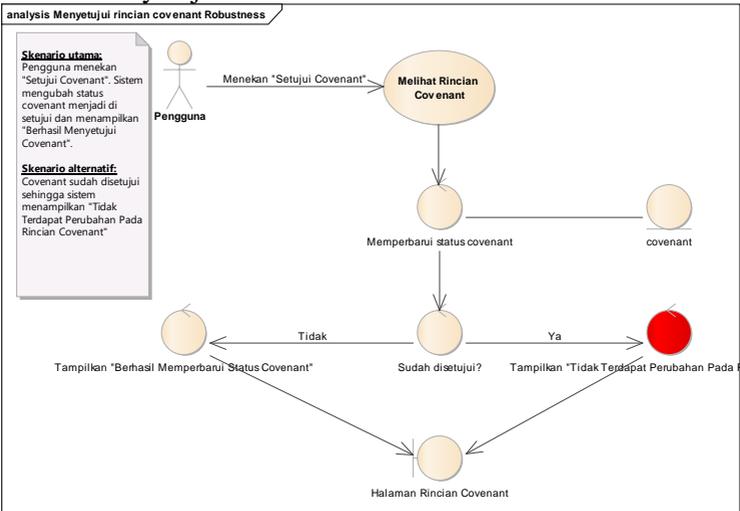
UC-32 Mengubah Rincian Covenant



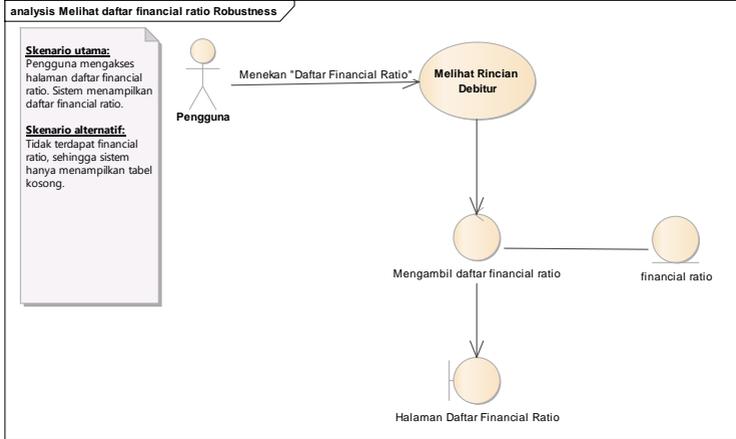
UC-33 Mengkonfirmasi Rincian Covenant



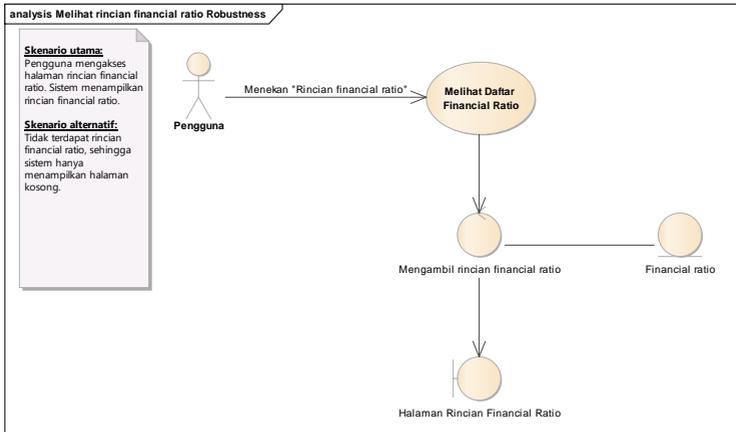
UC-34 Menyetujui Covenant



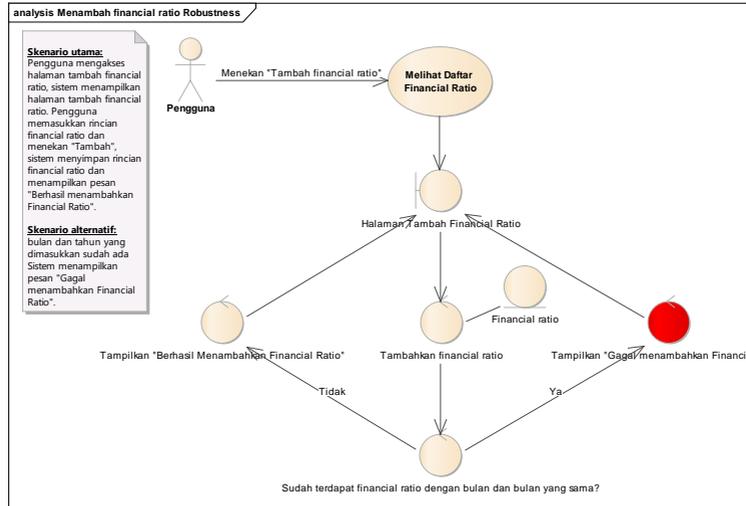
UC-35 Melihat Daftar Financial Ratio



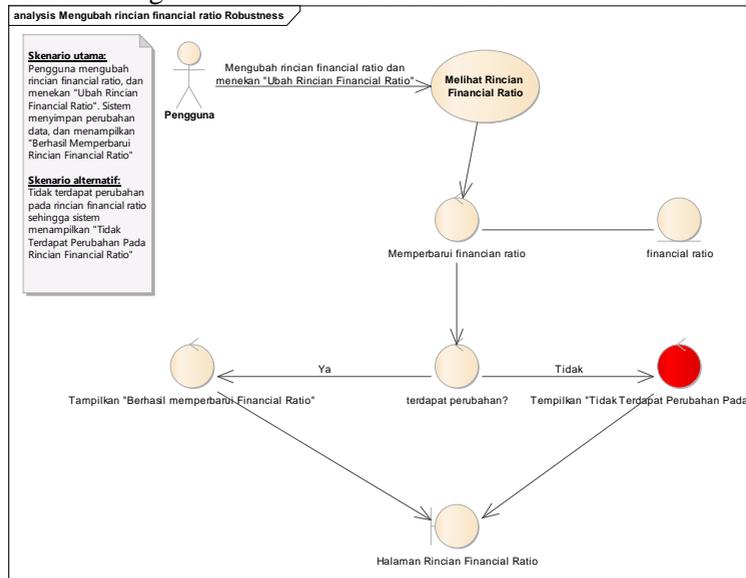
UC-36 Melihat Rincian Financial Ratio



UC-37 Menambah Financial Ratio



UC-38 Mengubah Rincian Financial Ratio

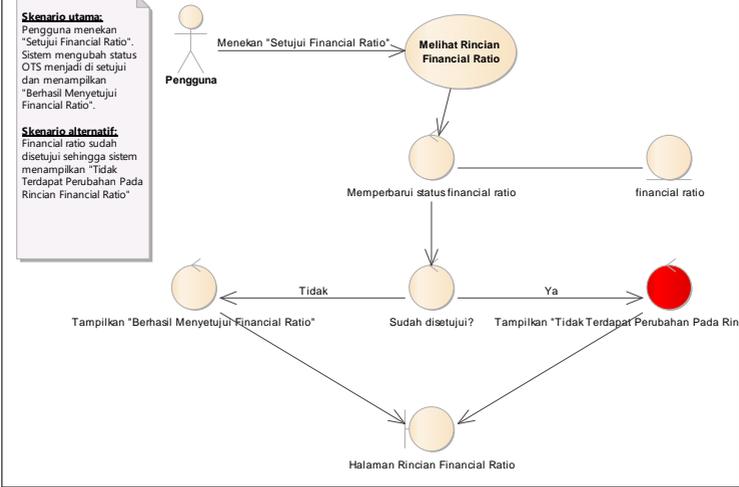


UC-39 Menyetujui Financial Ratio

analysis Menyetujui rincian financial ratio Robustness

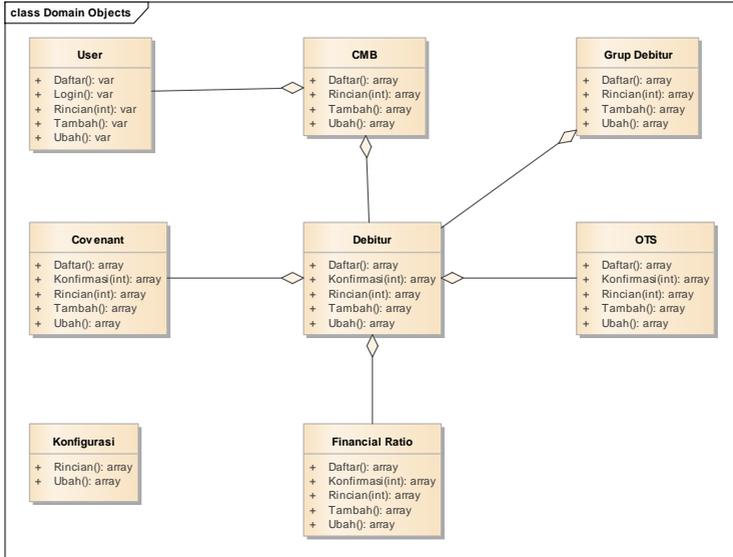
Skenario utama:
 Pengguna menekan "Setujui Financial Ratio". Sistem mengubah status OTS menjadi di setujui dan menampilkan "Berhasil Menyetujui Financial Ratio".

Skenario alternatif:
 Financial ratio sudah disetujui sehingga sistem menampilkan "Tidak Terdapat Perubahan Pada Rincian Financial Ratio"



LAMPIRAN E

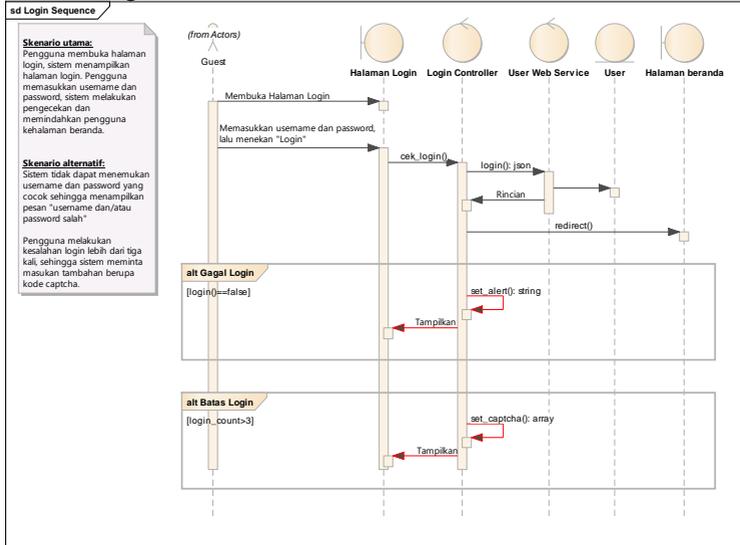
UPDATED DOMAIN MODEL



Halaman ini sengaja dikosongkan

LAMPIRAN F SEQUENCE DIAGRAM

UC-01 Login

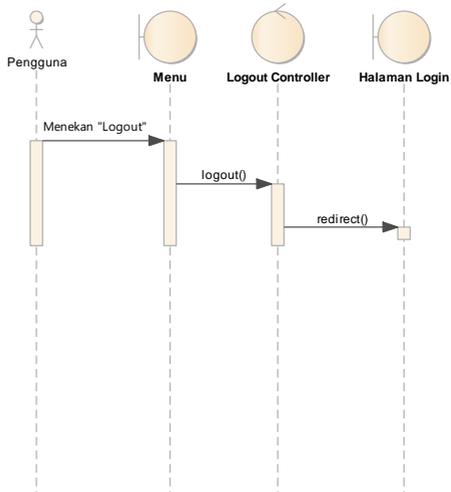


UC-02 Logout

sd Logout Sequence

Skenario utama:
 Pengguna menekan tombol logout. Sistem menghapus sesi pengguna dan memindahkan pengguna ke halaman login

Skenario alternatif:
 Sesi pengguna sudah habis. Sistem memindahkan pengguna ke halaman login tanpa menghapus sesi pengguna

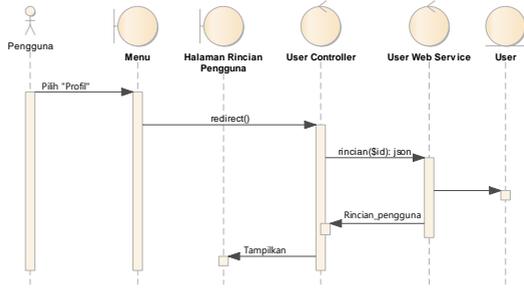


UC-03 Melihat Rincian Profil

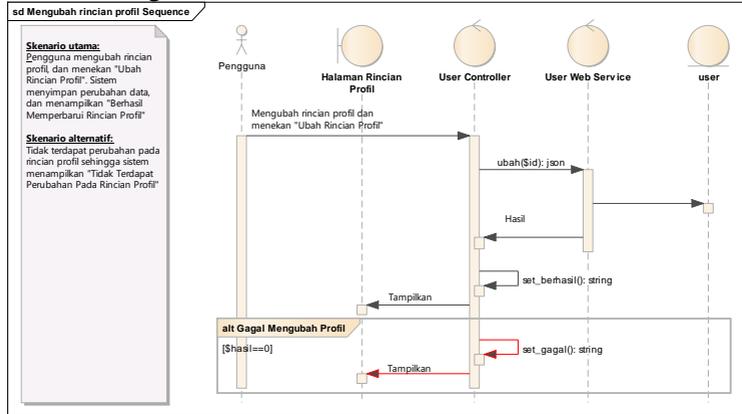
sd Melihat Rincian Profil Sequence

Skenario utama:
 Pengguna mengakses halaman rincian profil. Sistem menampilkan rincian profil.

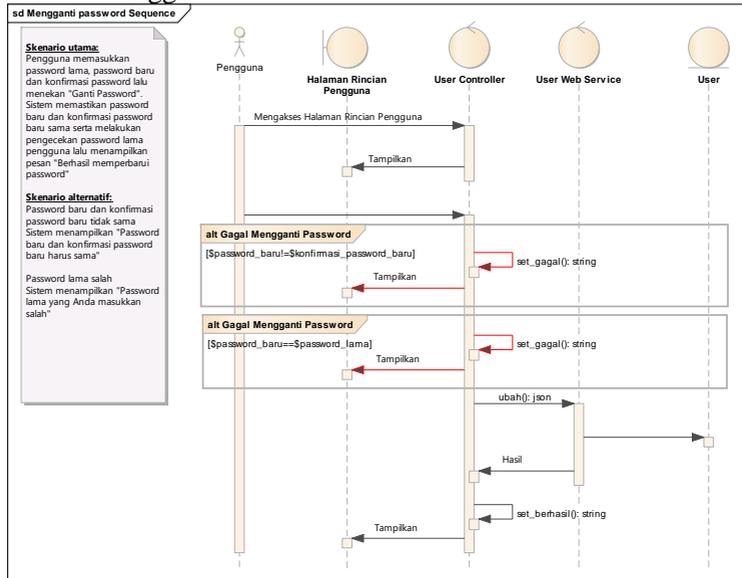
Skenario alternatif:
 Tidak terdapat rincian profil, sehingga sistem hanya menampilkan halaman kosong.



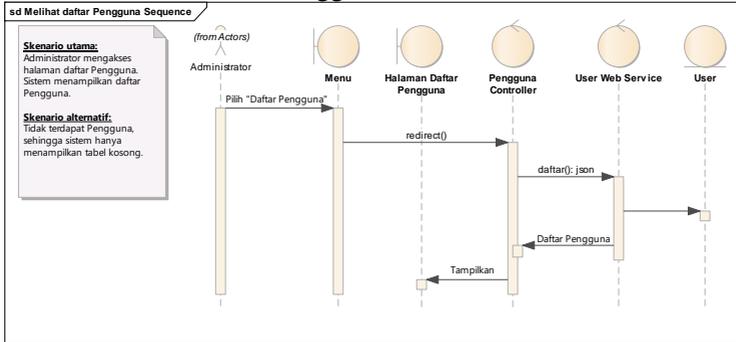
UC-04 Mengubah Rincian Profil



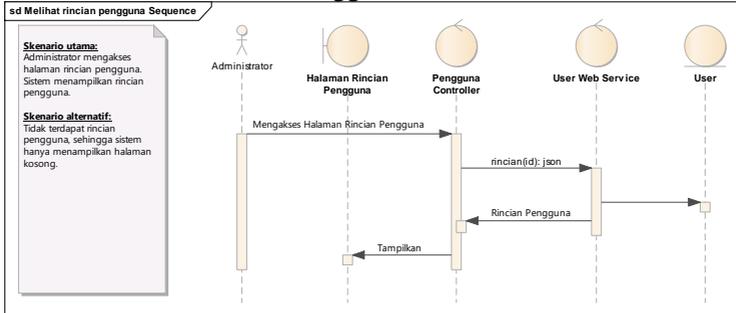
UC-05 Mengganti Password



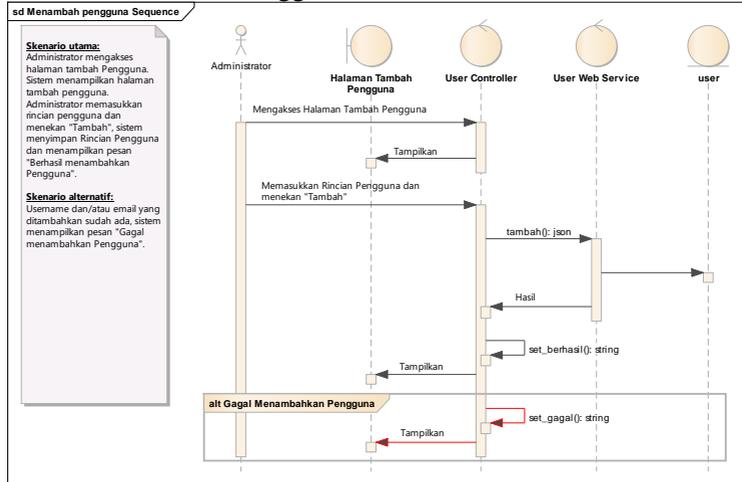
UC-06 Melihat Daftar Pengguna



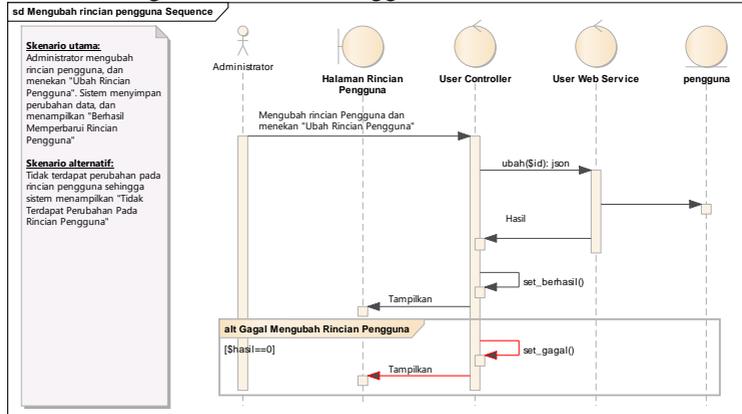
UC-07 Melihat Rincian Pengguna



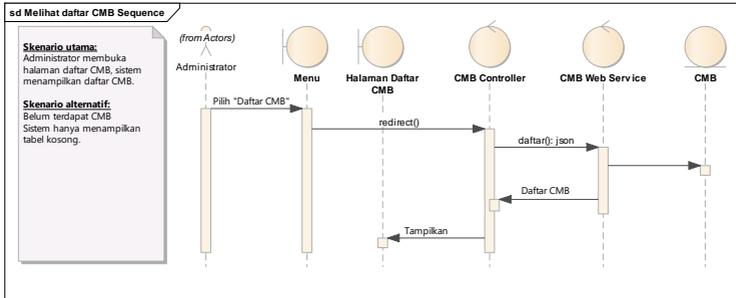
UC-08 Menambah Pengguna



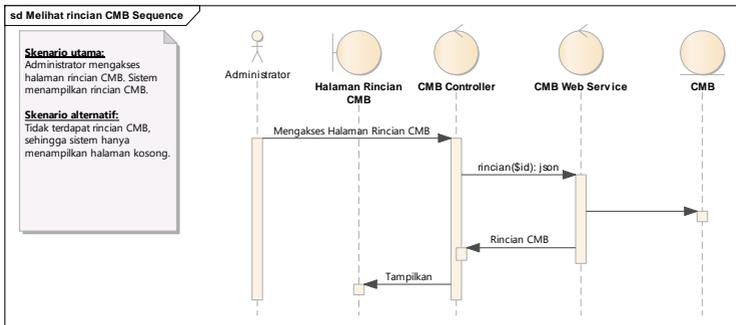
UC-09 Mengubah Rincian Pengguna



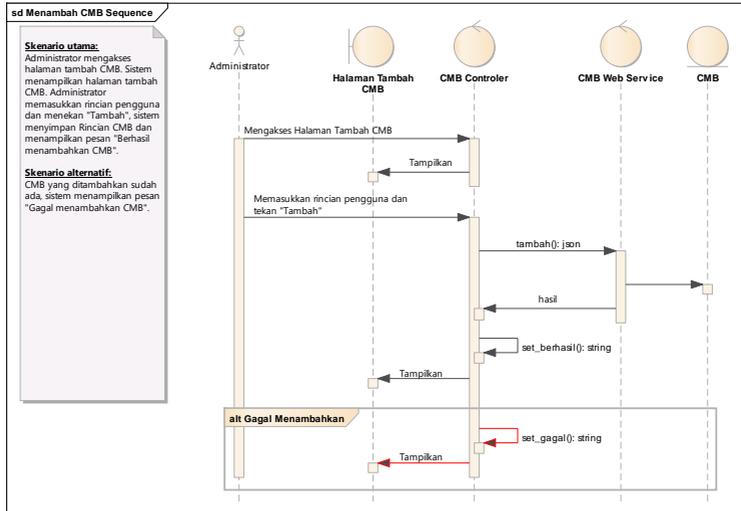
UC-10 Melihat Daftar CMB



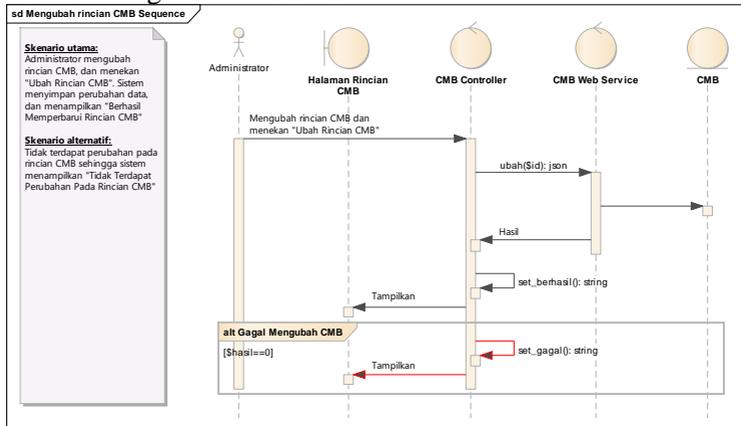
UC-11 Melihat Rincian CMB



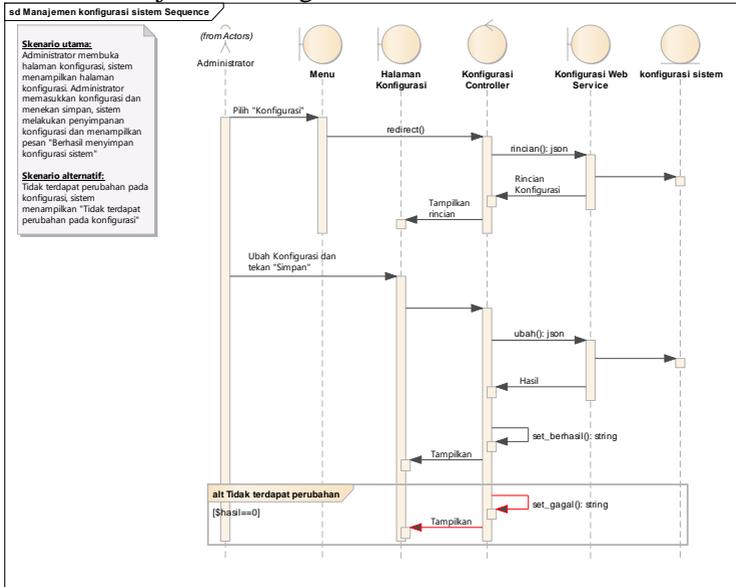
UC-12 Menambah CMB



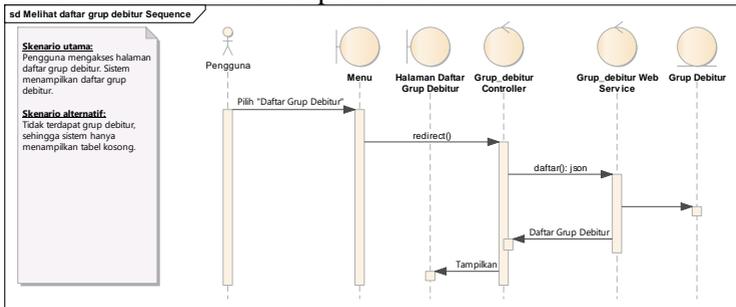
UC-13 Mengubah Rincian CMB



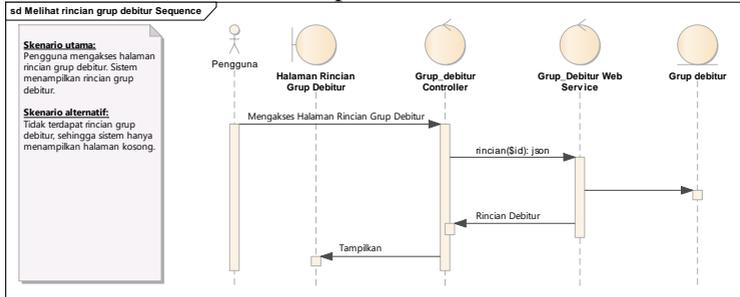
UC-14 Manajemen konfigurasi sistem



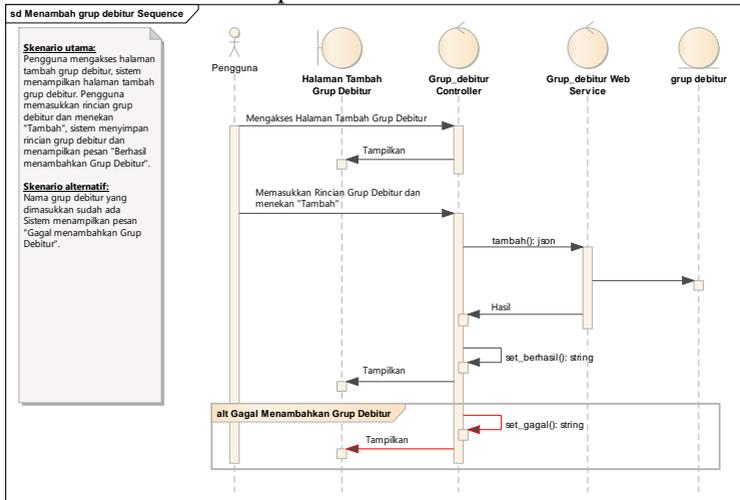
UC-15 Melihat Daftar Grup Debitur



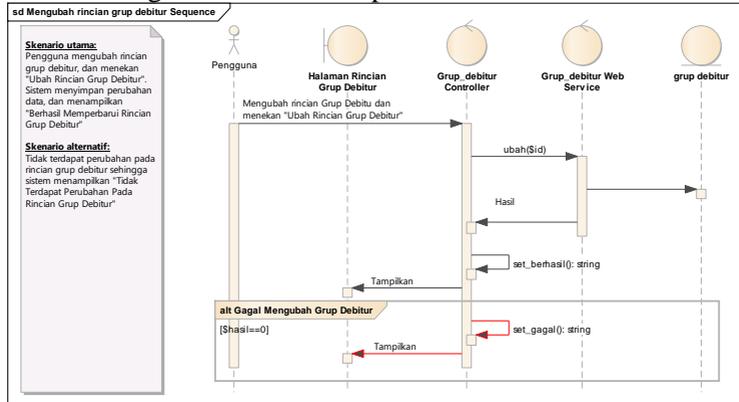
UC-16 Melihat Rincian Grup Debitur



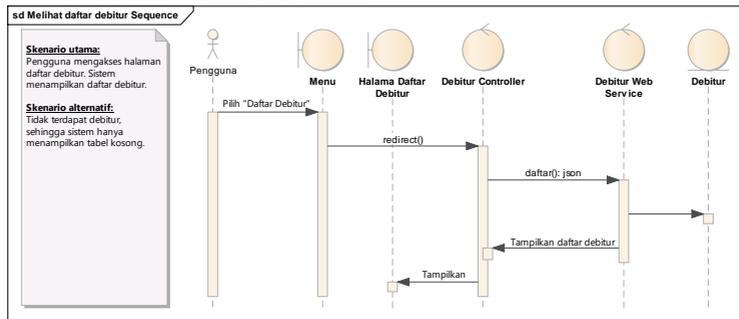
UC-17 Menambah Grup Debitur



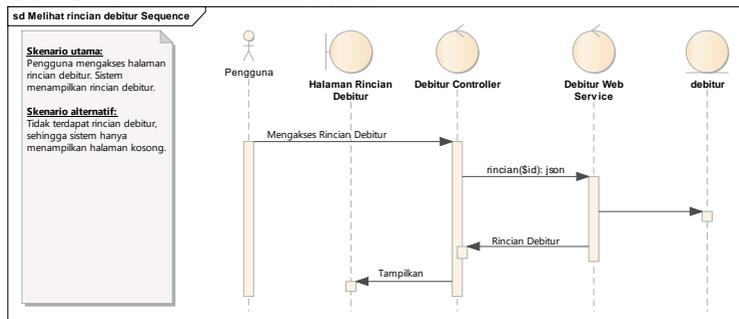
UC-18 Mengubah Rincian Grup Debitur



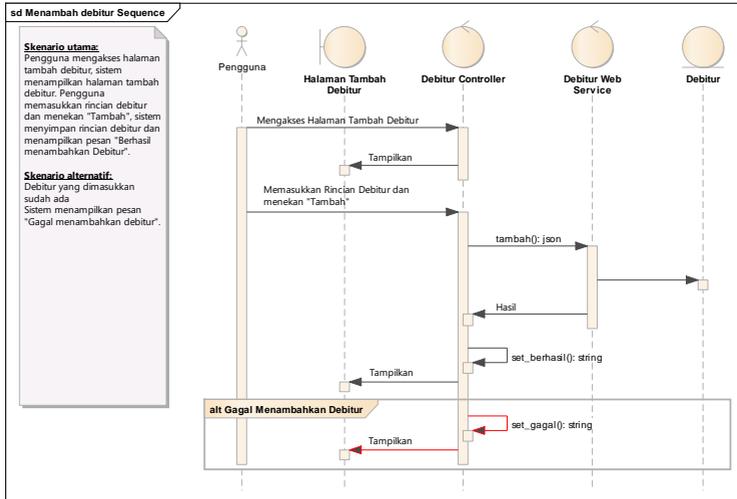
UC-19 Melihat Daftar Debitur



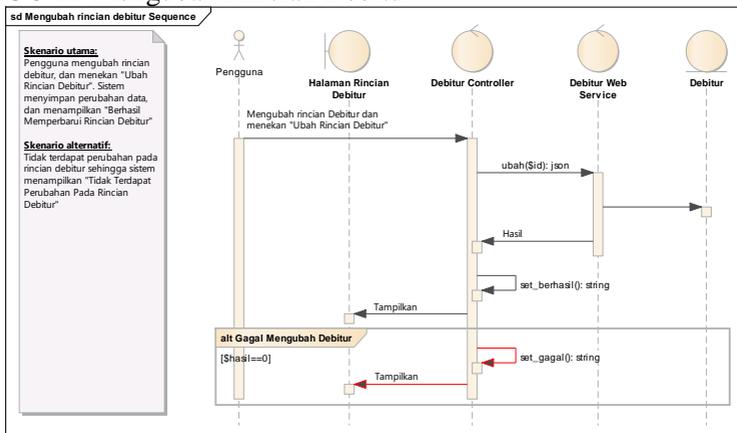
UC-20 Melihat Rincian Debitur



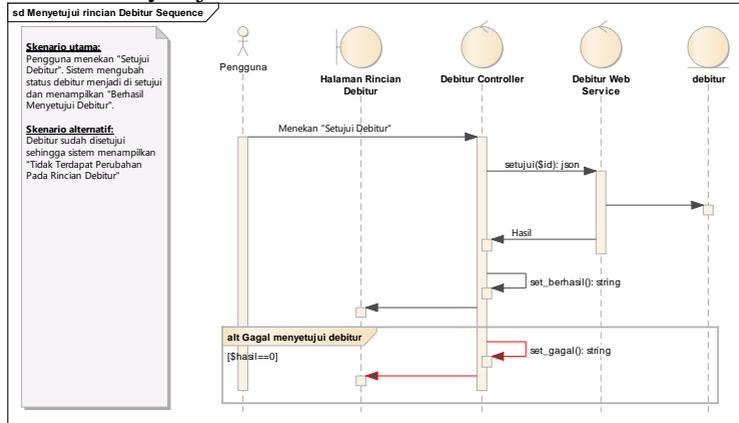
UC-21 Menambah Debitur



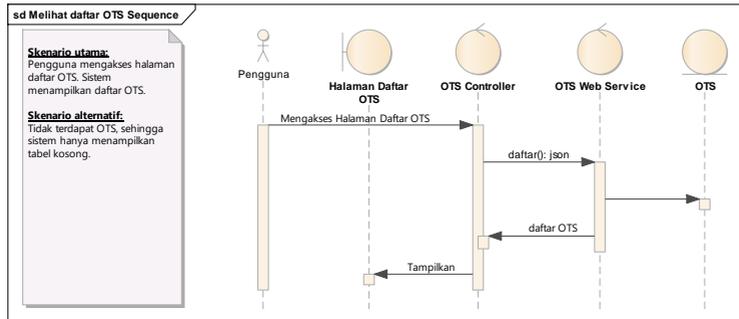
UC-22 Mengubah Rincian Debitur



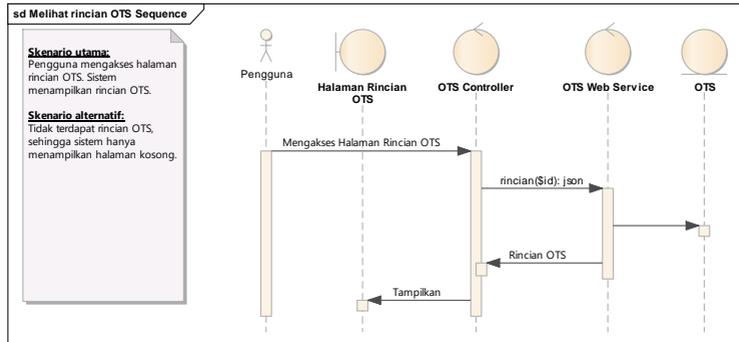
UC-23 Menyetujui Debitur



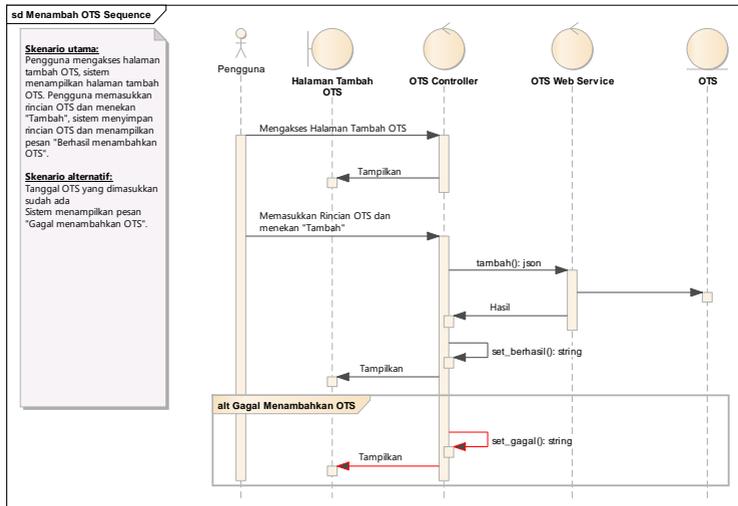
UC-24 Melihat Daftar OTS



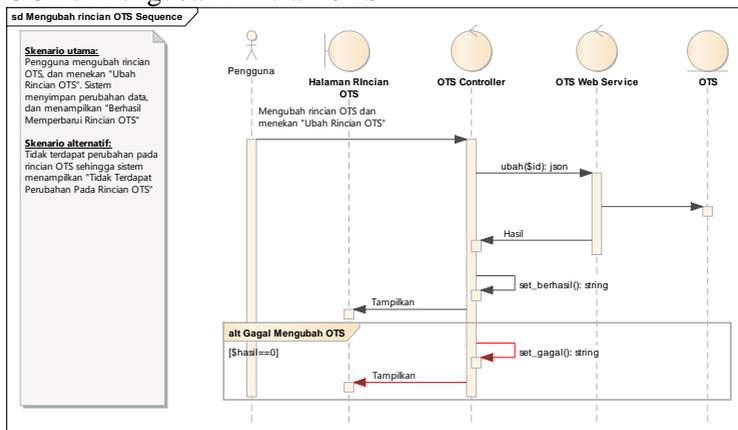
UC-25 Melihat Rincian OTS



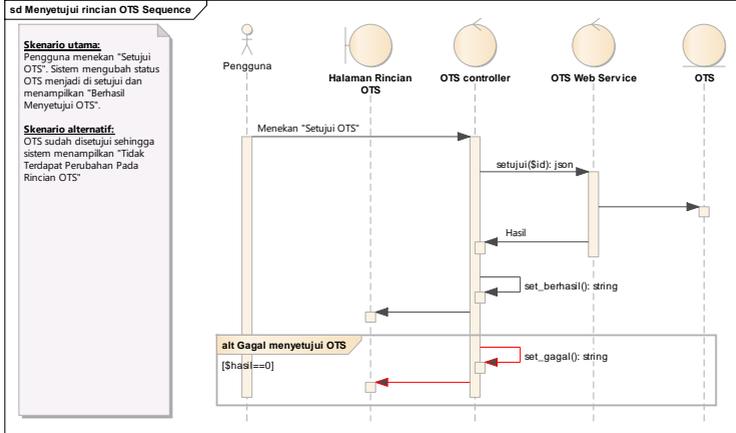
UC-26 Menambah OTS



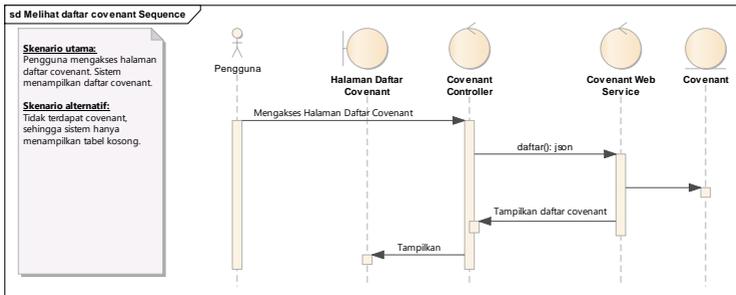
UC-27 Mengubah Rincian OTS



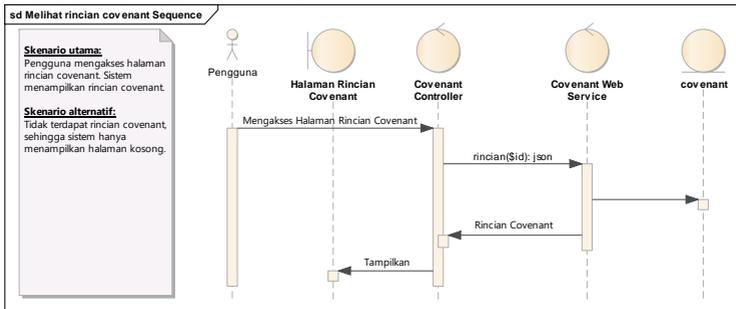
UC-28 Menyetujui OTS



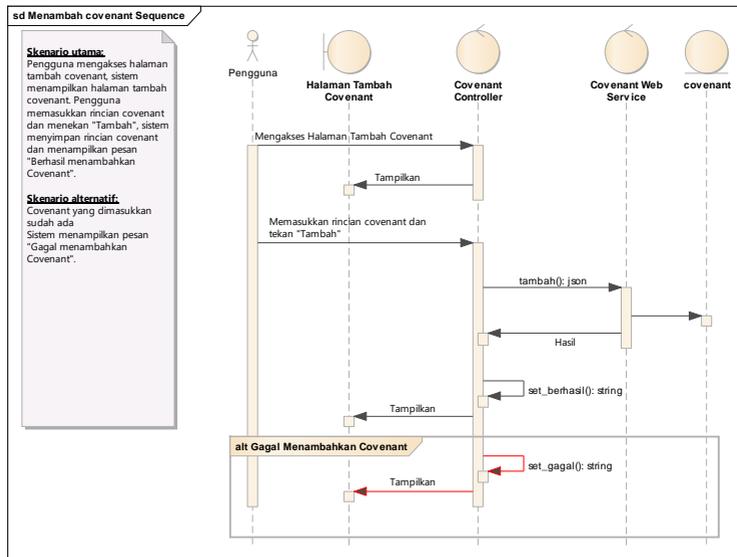
UC-29 Melihat Daftar Covenant



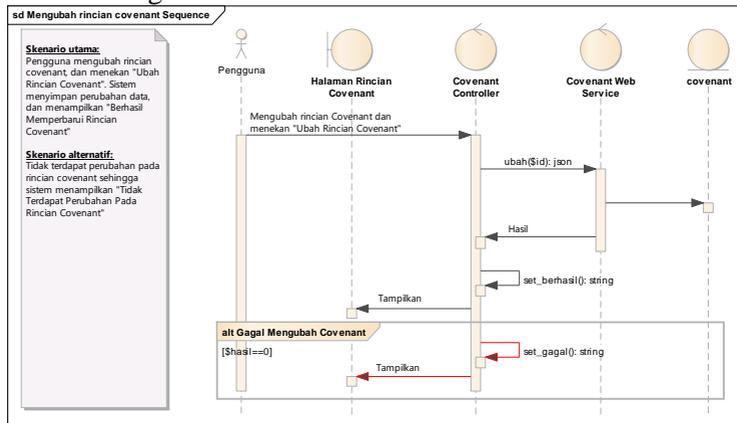
UC-30 Melihat Rincian Covenant



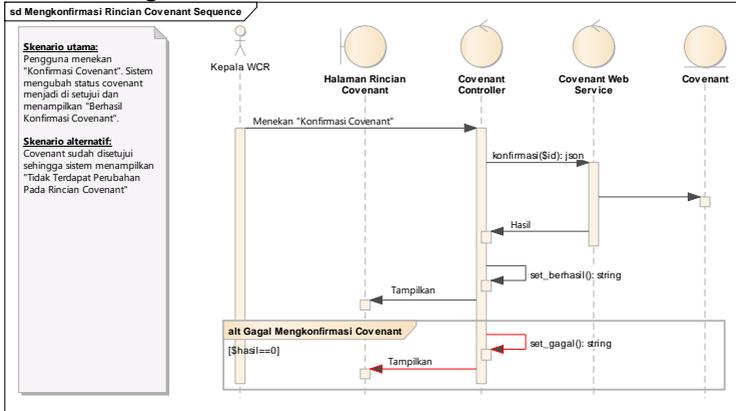
UC-31 Menambah Covenant



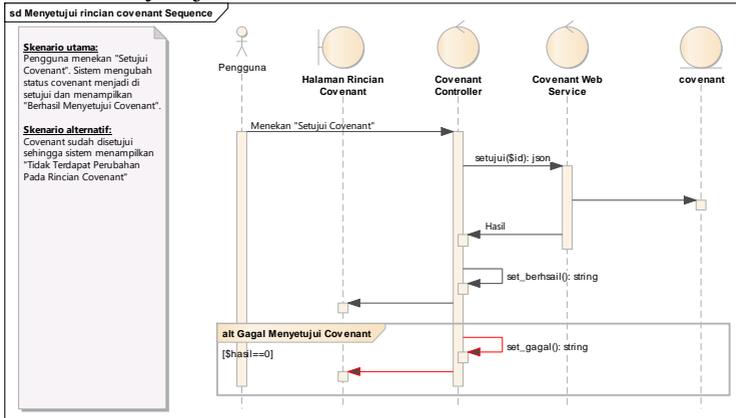
UC-32 Mengubah Rincian Covenant



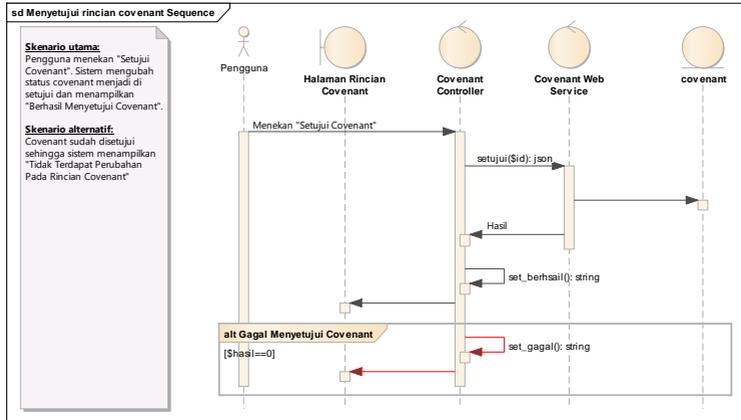
UC-33 Mengkonfirmasi Rincian Covenant



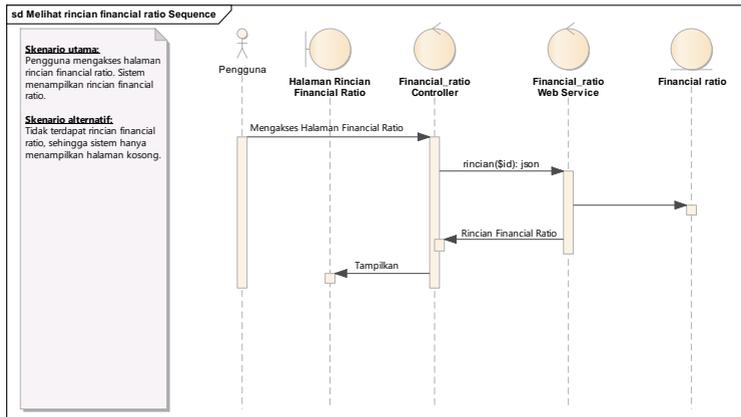
UC-34 Menyetujui Covenant



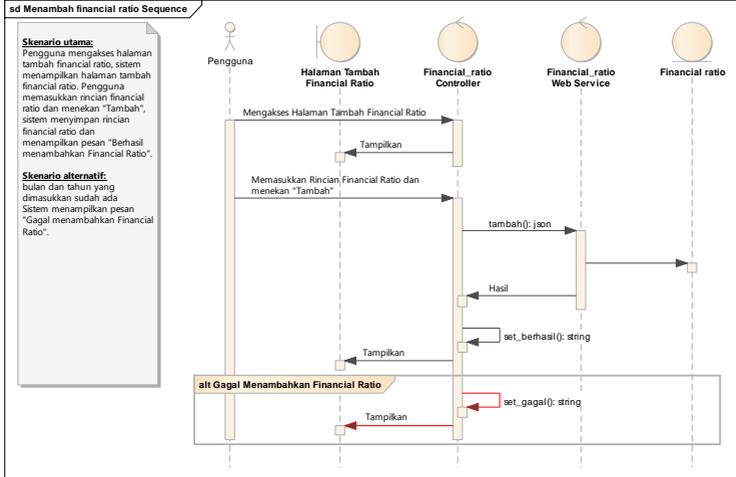
UC-35 Melihat Daftar Financial Ratio



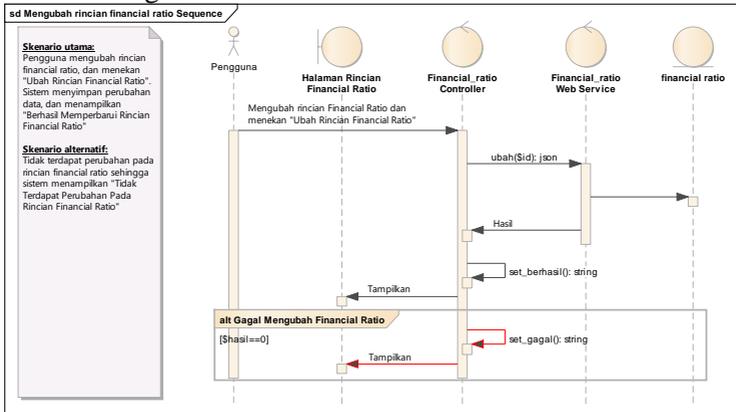
UC-36 Melihat Rincian Financial Ratio



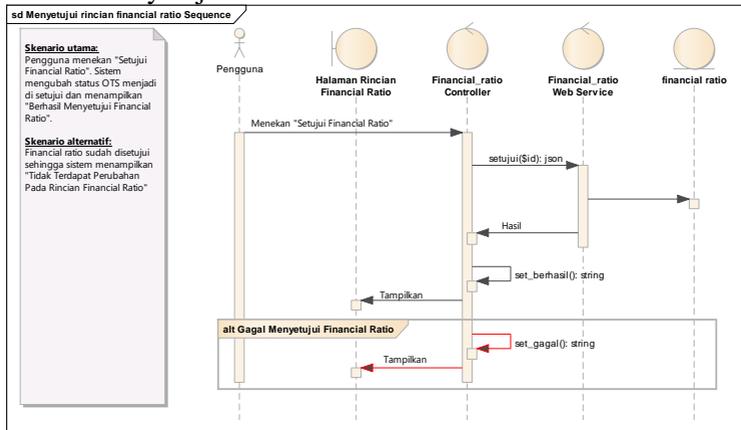
UC-37 Menambah Financial Ratio



UC-38 Mengubah Rincian Financial Ratio

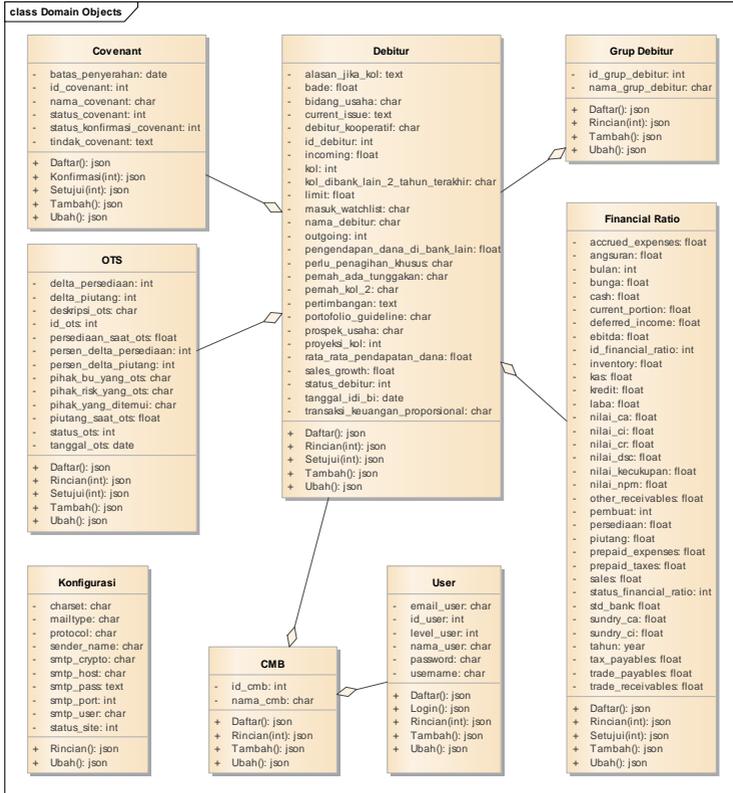


UC-39 Menyetujui Financial Ratio



Halaman ini sengaja dikosongkan

LAMPIRAN G CLASS DIAGRAM



G-2

Halaman ini sengaja dikosongkan