



TUGAS AKHIR - KI141502

**KERANGKA KERJA SINKRONISASI BASIS DATA
RELASIONAL BERBASIS WEB PADA STUDI KASUS
SISTEM INFORMASI AKADEMIK**

**TOMMY NURWANTORO
NRP 5111100050**

**Dosen Pembimbing I
Umi Laili Yuhana, S.Kom., M.Sc.**

**Dosen Pembimbing II
Rizky Januar Akbar, S.Kom., M.Eng.**

**JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2015**



UNDERGRADUATE THESES - KI141502

WEB BASED RELATIONAL DATABASE SYNCHRONIZATION FRAMEWORK ON ACADEMIC INFORMATION SYSTEM CASE STUDY

TOMMY NURWANTORO
NRP 5111100050

Supervisor I
Umi Laili Yuhana, S.Kom., M.Sc.

Supervisor II
Rizky Januar Akbar, S.Kom., M.Eng.

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2015

LEMBAR PENGESAHAN

KERANGKA KERJA SINKRONISASI BASIS DATA RELASIONAL BERBASIS WEB PADA STUDI KASUS SISTEM INFORMASI AKADEMIK

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Rekayasa Perangkat Lunak
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh
TOMMY NURWANTORO
NRP : 5111 100 050

Disetujui oleh Dosen Pembimbing Tugas Akhir

1. Umi Laili Yuhana, S.Kom., M.Sc.
NIP: 19790626 200501 2 002 (Pembimbing 1)
2. Rizky Januar Akbar, S.Kom., M.Eng.
NIP: 19870103 201404 1 001 (Pembimbing 2)

SURABAYA
JUNI, 2015

KERANGKA KERJA SINKRONISASI BASIS DATA RELASIONAL BERBASIS WEB PADA STUDI KASUS SISTEM INFORMASI AKADEMIK

Nama Mahasiswa : TOMMY NURWANTORO
NRP : 5111100050
Jurusan : Teknik Informatika FTIF-ITS
Dosen Pembimbing 1 : Umi Laili Yuhana, S.Kom., M.Sc.
Dosen Pembimbing 2 : Rizky Januar Akbar, S.Kom., M.Eng.

Abstrak

Sinkronisasi basis data merupakan suatu proses untuk menjaga konsistensi data pada basis data satu dengan basis data lainnya. Sinkronisasi data perlu dilakukan pada basis data yang membutuhkan data dari basis data lainnya. Perlu adanya sinkronisasi data pada kondisi seperti ini agar tidak terjadi kesalahan pengolahan data. Contoh nyata kondisi ini ada pada sistem informasi akademik. Sistem informasi akademik menggunakan data yang sangat banyak. Sedangkan data-data tersebut bisa diperoleh dari sistem informasi lain yang khusus mengolah jenis data tersebut.

Melihat besarnya kebutuhan penggunaan data dari luar sistem informasi akademik, dibutuhkan metode yang handal untuk dapat melakukan pertukaran data secara benar. Metode tersebut diharapkan dapat mempermudah pertukaran data yang dibutuhkan sehingga tidak dibutuhkan waktu yang lama untuk melakukan proses pengolahan data.

Tugas akhir ini menerapkan metode Change Data Capture untuk melakukan pencatatan perubahan data pada tabel. Selain itu juga menerapkan metode Routing and Batching melakukan pencatatan serta validasi pengiriman data sehingga tidak terjadi kesalahan pengiriman data.

Uji coba metode pada tugas akhir ini menggunakan data pendidik dan peserta didik sistem informasi akademik. Jumlah data pendidik yang dilakukan ujicoba adalah 55 baris. Sedangkan jumlah data peserta didik yang dilakukan ujicoba adalah 207 baris. Ujicoba perubahan data juga dilakukan untuk menguji kesamaan data setelah dilakukan perubahan.

Dari hasil evaluasi ujicoba, perubahan data bisa terdeteksi dan hasil pengiriman data sesuai dengan data sebenarnya. Sehingga dapat disimpulkan bahwa dengan metode ini sistem informasi akademik dapat melakukan pertukaran data dengan benar.

Tugas akhir ini dapat dijadikan salah satu alternatif pengiriman data apapun dari basis data satu ke basis data lainnya dengan mudah tanpa melakukan perubahan struktur basis data.

Kata kunci: Basis data, Change Data Capture, Routing and Batching, Sinkronisasi, Sistem Informasi Akademik.

WEB BASED RELATIONAL DATABASE SYNCHRONIZATION FRAMEWORK ON ACADEMIC INFORMATION SYSTEM CASE STUDY

Student's Name : TOMMY NURWANTORO
Student's ID : 5111100050
Department : Teknik Informatika FTIF-ITS
First Advisor : Umi Laili Yuhana, S.Kom., M.Sc.
Second Advisor : Rizky Januar Akbar, S.Kom., M.Eng.

Abstract

Database synchronization is a process to maintain the consistency data on a single database to other databases. Database synchronization needs to be done on database that needs data from other databases. In this condition, data synchronization is needed in order to avoid data processing errors. An example of this condition is in academic information system. Academic information system using so many data. While these data can be obtained from the other information systems that process these data specifically.

Given the tremendous need for the use of data from outside the academic information system, it need a reliable method to synchronize data. The method is expected to facilitate the exchange of data needed so it does not take a long time to perform data processing.

This undergraduated thesis applies the method of Change Data Capture to record data changes in the table. It also applies the method of Routing and Batching to perform recording and validation of data transmission so that there is no data transmission errors.

Methods testing in this undergraduated thesis uses data of student and teacher in academic information system.

The amount of teacher data testings are 55 rows and the amount of student data testings are 207 rows. Testing of data changes are also done to test the similarity data after changes.

From the above test methods, data changes can be detected and result data deliveries are same with the actual data. So that with this method, academic information systems can exchange data with other databases correctly.

This final task can be used as one alternative to sending any data from one database to another database easily and securely.

Keywords: Academic System Information, Database, Change Data Capture, Synchronization

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“KERANGKA KERJA SINKRONISASI BASIS DATA RELASIONAL PADA SISTEM INFORMASI AKADEMIK BERBASIS WEB”**.

Pengerjaan tugas akhir ini merupakan suatu kesempatan yang sangat baik bagi penulis. Dengan pengerjaan tugas akhir ini, penulis bisa belajar lebih banyak untuk memperdalam dan meningkatkan apa yang telah didapatkan penulis selama menempuh perkuliahan di Teknik Informatika ITS. Dengan tugas akhir ini penulis juga dapat menghasilkan suatu implementasi dari apa yang telah penulis pelajari.

Selesainya tugas akhir ini tidak lepas dari bantuan dan dukungan beberapa pihak. Sehingga pada kesempatan ini penulis mengucapkan syukur dan terima kasih kepada:

1. Allah SWT dan Nabi Muhammad SAW.
2. Mama, Papa, Tante Nuryatimah, beserta keluarga besar yang telah memberikan dukungan moral dan material serta do'a yang tak terhingga untuk penulis. Serta selalu memberikan semangat dan motivasi pada penulis dalam mengerjakan tugas akhir ini.
3. Ibu Umi Laili Yuhana, S.Kom., M.Sc. selaku pembimbing I yang telah membantu, membimbing, dan memotivasi penulis dalam menyelesaikan tugas akhir ini dengan sabar.
4. Bapak Rizky Januar Akbar, S.Kom., M.Eng. selaku pembimbing II yang juga telah membantu memberikan konsep, membimbing, dan memotivasi kepada penulis dalam mengerjakan tugas akhir ini.
5. Ibu Dr. Ir. Siti Rochimah, MT. dan Ibu Ratih Nur Esti Angraini, S.Kom., M.Sc. selaku dosen pembimbing LBE

yang telah membantu dan menemani selama proses pengerjaan proyek SIAKAD ini.

6. Ibu Dr. Eng. Nanik Suciati, S.Kom., M.Kom. selaku Kepala Jurusan Teknik Informatika ITS, Bapak Radityo Anggoro, S.Kom.,M.Sc. selaku koordinator TA, dan segenap dosen Teknik Informatika yang telah memberikan ilmunya.
7. Nimas Olivia Oktarini, Amd.Gz. yang telah memberikan motivasi, bantuan, doa serta hiburan kepada penulis dalam mengerjakan tugas akhir.
8. Teman-teman tim SIAKAD: Galih, Bustan, Rahman, Agung, dan Amanda yang selalu saling berbagi dalam pengerjaan tugas akhir ini.
9. Teman-teman The Contracant: Rahman, Ruslan, Bestama, Faris, Punggi, Toto, Dapik, Andrie, Yunus, Tev, Risal yang selalu memberikan hiburan kepada penulis dalam mengerjakan tugas akhir.
10. Teman-teman group keren: Riri, Amanda, Rifi, Luthfan, Ghani yang selama perkuliahan selalu berbagi bersama-sama.
11. Teman-teman angkatan 2011 yang telah membantu, berbagi ilmu, menjaga kebersamaan, dan memberi motivasi kepada penulis, kakak-kakak angkatan 2009 dan 2010 serta adik-adik angkatan 2012 dan 2013 yang membuat penulis untuk selalu belajar.
12. Serta semua pihak yang telah turut membantu penulis dalam menyelesaikan tugas akhir ini.

Penulis menyadari bahwa tugas akhir ini masih memiliki banyak kekurangan. Sehingga dengan kerendahan hati, penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depannya.

Surabaya, Juni 2015

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
<i>Abstrak</i>	vii
<i>Abstract</i>	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xviii
DAFTAR KODE SUMBER	xx
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	3
1.5 Manfaat.....	4
1.6 Metodologi	4
1.7 Sistematika Penulisan Laporan Tugas Akhir.....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1 Basis Data Relasional	7
2.2 Sinkronisasi Basis Data	9
2.3 Metode <i>Change Data Capture</i>	10
2.4 Metode <i>Routing and Batching</i>	11
2.5 PostgreSQL	12
2.6 <i>Java Database Connectivity (JDBC)</i>	13
2.7 Sistem Informasi Akademik	14
BAB III DESAIN PERANGKAT LUNAK.....	17
3.1 Analisis	17
3.1.1 Analisis Permasalahan.....	17
3.1.2 Deskripsi Umum Perangkat Lunak	18
3.1.3 Arsitektur Sistem.....	20
3.1.4 Kebutuhan Fungsional Aplikasi	22
3.2 Perancangan.....	33

3.2.1	Perancangan Basis Data.....	34
3.2.2	Perancangan Proses	37
3.2.3	Perancangan Antarmuka.....	42
BAB IV IMPLEMENTASI.....		47
4.1	Lingkungan Implementasi	47
4.1.1	Lingkungan Implementasi Perangkat Keras	47
4.1.2	Lingkungan Implementasi Perangkat Lunak	47
4.2	Implementasi Basis Data	49
4.2.1	Implementasi Struktur Basis Data Kerangka Kerja.....	49
4.2.2	Implementasi Struktur Basis Data Kebutuhan Sinkronisasi	50
4.3	Implementasi Proses	55
4.3.1	Proses Menambah Pengaturan	55
4.3.2	Proses Mengubah Pengaturan.....	55
4.3.3	Proses Menghapus Pengaturan	56
4.3.4	Proses Sinkronisasi Basis Data.....	57
4.3.5	Proses Melanjutkan Sinkronisasi Basis Data yang Terputus	59
4.4	Implementasi Antarmuka.....	61
4.4.1	Antarmuka Halaman Tambah Pengaturan.....	61
4.4.2	Antarmuka Halaman Menu Utama	72
4.4.3	Antarmuka Halaman Ubah Pengaturan	73
BAB V UJI COBA DAN EVALUASI.....		75
5.1	Lingkungan Uji Coba	75
5.2	Dasar Pengujian	76
5.3	Pengujian Fungsionalitas	76
5.3.1	Pengujian Pengelolaan Pengaturan Sinkbd.....	76
5.3.2	Pengujian Sinkronisasi Basis Data	80
5.4	Evaluasi Pengujian.....	107
BAB VI KESIMPULAN DAN SARAN.....		111
6.1	Kesimpulan.....	111
6.2	Saran	112
DAFTAR PUSTAKA.....		113
BIODATA PENULIS.....		115

DAFTAR TABEL

Tabel 2. 1 Tabel Mahasiswa.....	8
Tabel 2. 2 Tabel Mata Kuliah.....	8
Tabel 2. 3 Tabel Nilai.....	8
Tabel 2. 4 Tabel Deskripsi Fitur PostgreSQL	12
Tabel 3. 1 Deskripsi Kasus Penggunaan Sinkbd.....	22
Tabel 3. 2 Rincian Alur Kasus Penggunaan UC-W01	23
Tabel 3. 3 Rincian Alur Kasus Penggunaan UC-W02	27
Tabel 3. 4 Rincian Alur Kasus Penggunaan UC-W03	31
Tabel 3. 5 Atribut Tabel sink_bd	35
Tabel 3. 6 Atribut Tabel sink_data, sinkbd_inbound, dan sinkbd_outbound.....	36
Tabel 5. 1 Tabel Spesifikasi Lingkungan Pengujian <i>Source</i> ..	75
Tabel 5. 2 Tabel Spesifikasi Lingkungan Pengujian <i>Target</i> ..	75
Tabel 5. 3 Tabel Spesifikasi Lingkungan Pengujian Server Sinkbd	76
Tabel 5. 4 Skenario Pengujian Membuat Pengaturan	77
Tabel 5. 5 Skenario Pengujian Mengubah Pengaturan.....	77
Tabel 5. 6 Skenario Pengujian Menghapus Pengaturan	78
Tabel 5. 7 Skenario Uji Coba 1	82
Tabel 5. 8 Skenario Uji Coba 2	85
Tabel 5. 9 Skenario Uji Coba 3	88
Tabel 5. 10 Skenario Uji Coba 4	92
Tabel 5. 11 Skenario Uji Coba 5	96
Tabel 5. 12 Skenario Uji Coba 6	97
Tabel 5. 13 Skenario Uji Coba 7 Bagian 1.....	100
Tabel 5. 14 Skenario Uji Coba 7 Bagian 2.....	100
Tabel 5. 15 Skenario Uji Coba 7 Bagian 3.....	101
Tabel 5. 16 Skenario Uji Coba 8.....	106

DAFTAR GAMBAR

Gambar 2. 1 Istilah dalam basis data relasional	9
Gambar 2. 2 <i>Change Data Capture</i>	10
Gambar 2. 3 <i>Routing and Batching</i>	11
Gambar 2. 4 Arsitektur JDBC	13
Gambar 3. 1 Arsitektur Sistem	20
Gambar 3. 2 Diagram Kasus Penggunaan Sinkbd	22
Gambar 3. 3 Diagram Aktifitas Kasus Penggunaan UC-W0126	
Gambar 3. 4 Diagram Aktifitas Kasus Penggunaan UC-W0230	
Gambar 3. 5 Diagram Aktifitas Kasus Penggunaan UC-W0233	
Gambar 3. 6 Conceptual Data Model (CDM) Kerangka Kerja Sinkbd	34
Gambar 3. 7 Physical Data Model (PDM) Kerangka Kerja Sinkbd	35
Gambar 3. 8 Rancangan Struktur <i>File</i> xml	38
Gambar 3. 9 Alur Proses Sinkronisasi Basis Data	40
Gambar 3. 10 Rancangan Antarmuka Halaman Menu Utama	42
Gambar 3. 11 Rancangan Antarmuka Sub Halaman Pengaturan Sambungan Basis Data	43
Gambar 3. 12 Rancangan Antarmuka Sub Halaman Pengaturan Tabel dan Kolom	44
Gambar 3. 13 Rancangan Antarmuka Sub Halaman Konfirmasi Penyimpanan Pengaturan	46
Gambar 4. 1 Isi <i>File configuration.properties</i>	48
Gambar 4. 2 Pengaturan <i>Super Set</i> dan <i>Sub Set</i>	52
Gambar 4. 3 Antarmuka Sub Halaman Sambungan Basis Data	62
Gambar 4. 4 Validasi <i>Form</i> Nama Pengaturan	63
Gambar 4. 5 Validasi <i>Form</i> Alamat Basis Data	64
Gambar 4. 6 Memasukkan <i>Username</i> dan <i>Password</i> Server	64
Gambar 4. 7 Antarmuka Sub Halaman Pengaturan Tabel dan Kolom	66

Gambar 4. 8 Validasi <i>Form</i> Tidak Ada Pengaturan Disimpan	67
Gambar 4. 9 Validasi Form Jumlah Kolom Dipilih Tidak Sama	68
Gambar 4. 10 Tampilan Setelah Pencocokan Tabel dan Kolom	70
Gambar 4. 11 Antarmuka Sub Halaman Konfirmasi Penyimpanan Pengaturan	71
Gambar 4. 12 Antarmuka Halaman Menu Utama	72
Gambar 4. 13 Antarmuka Halaman Ubah Pengaturan	74
Gambar 5. 1 (a,b,c) Ilustrasi Pengujian Proses Pengelolaan Pengaturan	80
Gambar 5. 2 Ilustrasi <i>File Xml</i> Dalam <i>Folder configuration</i>	80
Gambar 5. 3 Pengaturan Untuk Pengujian 1 dan 2.....	81
Gambar 5. 4 Pengaturan Untuk Pengujian 3 dan 4.....	81
Gambar 5. 5 Perubahan Jumlah Baris Tabel Sinkronisasi.....	83
Gambar 5. 6 Isi Tabel <i>sink_data</i> Pada <i>Source</i>	84
Gambar 5. 7 Isi Tabel <i>Target</i> Sebelum Sinkronisasi	84
Gambar 5. 8 Isi Tabel <i>Target</i> Setelah Sinkronisasi	84
Gambar 5. 9 Perubahan Jumlah Baris Tabel Sinkronisasi.....	86
Gambar 5. 10 Isi Tabel <i>sink_data</i> Pada <i>Source</i>	87
Gambar 5. 11 Isi Tabel <i>Target</i> Sebelum Sinkronisasi	87
Gambar 5. 12 Isi Tabel <i>Target</i> Setelah Sinkronisasi	87
Gambar 5. 13 Perubahan Jumlah Baris Tabel Sinkronisasi.....	89
Gambar 5. 14 Isi Tabel <i>sink_data</i> Pada <i>Source</i>	90
Gambar 5. 15 Isi Tabel <i>Target</i> Sebelum Sinkronisasi	90
Gambar 5. 16 Isi Tabel <i>Target</i> Setelah Sinkronisasi	91
Gambar 5. 17 Perubahan Jumlah Baris Tabel Sinkronisasi.....	93
Gambar 5. 18 Isi Tabel <i>sink_data</i> Pada <i>Source</i>	94
Gambar 5. 19 Isi Tabel <i>Target</i> Sebelum Sinkronisasi	94
Gambar 5. 20 Isi Tabel <i>Target</i> Setelah Sinkronisasi	95
Gambar 5. 21 Relasi Tabel Pengujian pd dan ptk	95
Gambar 5. 22 Pesan <i>Error</i> Memasukkan Data Valid	96
Gambar 5. 23 Isi Tabel ptk <i>Source</i> Sebelum Sinkronisasi	98
Gambar 5. 24 Isi Tabel ptk <i>Source</i> Setelah Sinkronisasi.....	98

Gambar 5. 25 Isi Tabel ptk <i>Target</i> Sebelum Sinkronisasi	98
Gambar 5. 26 Isi Tabel ptk <i>Target</i> Setelah Sinkronisasi.....	99
Gambar 5. 27 Pengaturan Khusus Uji Coba 7.....	99
Gambar 5. 28 Isi Tabel ptk <i>Sub Set</i> 1 Sebelum Sinkronisasi	101
Gambar 5. 29 Isi Tabel ptk <i>Sub Set</i> 1 Setelah Sinkronisasi dummy_1.....	102
Gambar 5. 30 Isi Tabel ptk <i>Sub Set</i> 1 Setelah Sinkronisasi dummy_2.....	102
Gambar 5. 31 Isi Tabel ptk <i>Sub Set</i> 1 Setelah Sinkronisasi dummy_1 Kedua Kalinya.....	102
Gambar 5. 32 Isi Tabel ptk <i>Super Set</i> Sebelum Sinkronisasi	103
Gambar 5. 33 Isi Tabel ptk <i>Super Set</i> Setelah Sinkronisasi dummy_1.....	103
Gambar 5. 34 Isi Tabel ptk <i>Super Set</i> Setelah Sinkronisasi dummy_2.....	103
Gambar 5. 35 Isi Tabel ptk <i>Super Set</i> Setelah Sinkronisasi dummy_1 Kedua Kalinya.....	104
Gambar 5. 36 Isi Tabel ptk <i>Sub Set</i> 2 Sebelum Sinkronisasi	104
Gambar 5. 37 Isi Tabel ptk <i>Sub Set</i> 2 Setelah Sinkronisasi dummy_1.....	104
Gambar 5. 38 Isi Tabel ptk <i>Sub Set</i> 2 Setelah Sinkronisasi dummy_2.....	105
Gambar 5. 39 Isi Tabel ptk <i>Sub Set</i> 2 Setelah Sinkronisasi dummy_1 Kedua Kalinya.....	105
Gambar 5. 40 Tampilan Pesan Sinkronisasi Gagal	106
Gambar 5. 41 Isi Tabel sinkbd_outbound <i>Source</i>	107
Gambar 5. 42 Isi Tabel sinkbd_inbound <i>Target</i> Setelah Sinkronisasi Gagal	107
Gambar 5. 43 Isi Tabel sinkbd_inbound <i>Target</i> Setelah Melanjutkan Sinkronisasi.....	107

DAFTAR KODE SUMBER

Kode Sumber 4. 1 Implementasi Pembuatan Skema	49
Kode Sumber 4. 2 Implementasi Struktur Tabel sink_bd	50
Kode Sumber 4. 3 Implementasi Pembuatan Skema Pada <i>Source</i> dan <i>Target</i>	50
Kode Sumber 4. 4 Implementasi Struktur Tabel sink_data....	51
Kode Sumber 4. 5 Implementasi Struktur Tabel sinkbd_inbound.....	52
Kode Sumber 4. 6 Implementasi Struktur Tabel sinkbd_outbound.....	53
Kode Sumber 4. 7 Implementasi Fungsi get_pk	53
Kode Sumber 4. 8 Implementasi Fungsi process_sink_data..	54
Kode Sumber 4. 9 Implementasi Proses Menambah Pengaturan.....	55
Kode Sumber 4. 10 Implementasi Proses Mengubah Pengaturan.....	56
Kode Sumber 4. 11 Implementasi Proses Menghapus Pengaturan.....	57
Kode Sumber 4. 12 Implementasi Mendapatkan Waktu Terakhir Sinkronisasi	57
Kode Sumber 4. 13 Implementasi Mendapatkan Data Sinkronisasi	57
Kode Sumber 4. 14 Implementasi Pengiriman Data <i>Source</i> ke Pusat	58
Kode Sumber 4. 15 Implementasi Pengiriman Data Pusat ke <i>Target</i>	58
Kode Sumber 4. 16 Implementasi Memasukkan Data Valid ...	59
Kode Sumber 4. 17 Implementasi Melanjutkan Sinkronisasi Basis Data yang Terputus.....	61
Kode Sumber 4. 18 Kode Untuk Memeriksa Nama Pengaturan	63
Kode Sumber 4. 19 Kode Untuk Mendapatkan Seluruh Nama Tabel.....	65

Kode Sumber 4. 20 Kode Untuk Mendapatkan Seluruh Nama Kolom.....	70
Kode Sumber 4. 21 Kode Untuk Menyimpan Pengaturan	72
Kode Sumber 4. 22 Kode Untuk Menangani Fungsi Hapus Pengaturan	73
Kode Sumber 4. 23 Kode Untuk Menangani Ubah Pengaturan	74

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi saat ini sudah semakin berkembang pesat. Salah satunya adalah pemanfaatan teknologi WWW (*World Wide Web*) atau yang biasa disebut web yang dapat menyediakan informasi dalam bentuk teks, gambar, dan suara. Begitu juga pada pemanfaatan basis data. Setiap web saat ini sudah memiliki basis datanya masing-masing dengan data yang khusus dikelola oleh web tersebut. Tidak heran banyak web yang mengelola data, namun data tambahannya diambil dari sistem web lain yang khusus mengelola data tersebut. Namun data yang digunakan harus dilakukan pembaruan secara berkala untuk menjaga kesamaan data.

Untuk menangani permasalahan tersebut, dibutuhkan sebuah aplikasi yang dapat melakukan sinkronisasi basis data yang satu sebagai basis data sumber dengan basis data lainnya sebagai basis data tujuan. Basis data yang digunakan mempunyai struktur yang berbeda namun data-data yang dibutuhkan dapat saling berhubungan. Dengan begitu basis data tujuan dapat mengambil data dari basis data sumber tanpa mengubah struktur basis data, begitu pula sebaliknya

Contoh studi kasus yang cocok untuk menerapkan sinkronisasi data adalah sistem informasi akademik berbasis web. Dengan memanfaatkan teknologi web, sistem informasi akademik yang merupakan pusat pengolahan dan penyajian data pada institusi pendidikan dapat semakin mudah untuk diakses. Sistem informasi akademik berbasis web tidak hanya mengelola data inputan, namun juga data-data dosen, karyawan, keuangan, organisasi, dan lain-lain. Data-data ini tidak bersumber dari sistem informasi akademik langsung, melainkan didapat dari

sistem informasi yang lain. Dengan kata lain, sistem informasi tersebut bisa saja menggunakan basis data yang berbeda dari basis data yang digunakan sistem informasi akademik. Sebagai contoh, sistem informasi akademik untuk mengelola data dosen dan karyawan harus mendapatkan data terbaru dari sistem kepegawaian. Jika bukan data terbaru, maka keabsahan data diragukan. Seiring berjalannya masa pembelajaran, perubahan data akan sering terjadi. Perubahan data tersebut dapat berupa penambahan data pegawai baru hingga perubahan informasi yang berhubungan dengan keuangan dan akademik siswa. Akan sangat sulit apabila pengambilan data dari basis data yang berbeda seperti ini dilakukan dengan cara membandingkan data satu persatu.

Dalam tugas akhir ini, sistem akan merancang kerangka kerja yang dapat melakukan sinkronisasi data antara dua basis data tanpa melakukan perubahan struktur pada masing-masing basis data. Aplikasi akan menggunakan bahasa pemrograman Java dan menggunakan basis data PostgreSQL.

Tujuan dari dikembangkannya kerangka kerja ini adalah untuk memberikan solusi kemudahan dalam melakukan sinkronisasi data antara dua basis data tanpa mengubah struktur utamanya sehingga tidak terjadi kesalahan pada aplikasi yang menggunakan data tersebut.

1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana spesifikasi kebutuhan kerangka kerja yang dapat melakukan sinkronisasi antara dua basis data?
2. Bagaimana metode sinkronisasi basis data yang tidak mengubah struktur basis data yang sudah ada?

3. Bagaimana membangun purwarupa kerangka kerja sinkronisasi basis data relasional pada sistem informasi akademik berbasis web?
4. Bagaimana menjamin keabsahan data ketika melakukan sinkronisasi?

1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, diantaranya sebagai berikut:

1. Kerangka kerja sinkronisasi basis data relasional ini menggunakan bahasa pemrograman Java dan menggunakan basis data PostgreSQL.
2. Aplikasi ini hanya dapat diakses oleh administrator basis data.
3. Aplikasi ini hanya digunakan untuk kebutuhan Sistem informasi akademik.
4. Struktur basis data sudah diketahui sebelumnya.
5. Uji coba dilakukan pada dua basis data PostgreSQL dengan struktur basis data yang berbeda.

1.4 Tujuan

Tujuan dari pengerjaan tugas akhir ini adalah sebagai berikut:

1. Melakukan spesifikasi kebutuhan kerangka kerja yang dapat melakukan sinkronisasi antara dua basis data.
2. Menentukan metode sinkronisasi basis data yang tidak mengubah struktur basis data yang sudah ada.
3. Membuat purwarupa aplikasi sinkronisasi basis data relasional pada sistem informasi akademik berbasis web.
4. Melakukan pengujian kerangka kerja untuk menjamin keabsahan data ketika melakukan sinkronisasi.

1.5 Manfaat

Manfaat yang diharapkan dari dikembangkannya kerangka kerja sinkronisasi ini antara lain:

1. Memudahkan pembaruan data dari basis data satu ke basis data lainnya tanpa mengubah struktur basis data utama.
2. Menjaga keabsahan data terbaru yang digunakan.
3. Menunjang basis data sistem informasi akademik berbasis web agar dapat melakukan sinkronisasi data dengan basis data sistem informasi lain yang dibutuhkan.

1.6 Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1. Penyusunan proposal tugas akhir.

Tahap awal untuk memulai pengerjaan tugas akhir adalah penyusunan proposal tugas akhir. Proposal tugas akhir yang diajukan memiliki gagasan yang sama dengan tugas akhir ini, yaitu kerangka kerja sinkronisasi basis data relasional pada sistem informasi akademik berbasis web.

2. Studi literatur

Pada tahap ini dilakukan pencarian, pengumpulan, pembelajaran dan pemahaman informasi dan literatur yang diperlukan untuk pembuatan kerangka kerja sinkronisasi basis data relasional pada sistem informasi akademik berbasis web. Informasi dan literatur didapatkan dari literatur buku dan sumber-sumber informasi lain yang berhubungan.

3. Analisis dan desain perangkat lunak

Tahap ini meliputi perancangan sistem berdasarkan studi literatur dan pembelajaran konsep teknologi dari perangkat lunak yang ada. Tahap ini mendefinisikan alur dari implementasi. Langkah-langkah yang dikerjakan juga

didefinisikan pada tahap ini. Pada tahapan ini dibuat *prototype* sistem, yang merupakan rancangan dasar dari sistem yang akan dibuat. Serta dilakukan desain suatu sistem dan desain proses-proses yang ada.

4. Implementasi perangkat lunak

Implementasi merupakan tahap membangun rancangan program yang telah dibuat. Pada tahapan ini merealisasikan apa yang terdapat pada tahapan sebelumnya, sehingga menjadi sebuah program yang sesuai dengan apa yang telah direncanakan.

5. Pengujian dan evaluasi

Pada tahapan ini dilakukan uji coba pada data yang telah dikumpulkan. Tahapan ini dimaksudkan untuk mengevaluasi kesesuaian data dan program serta mencari masalah yang mungkin timbul dan mengadakan perbaikan jika terdapat kesalahan.

6. Penyusunan buku tugas akhir.

Pada tahapan ini disusun buku yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

1.7 Sistematika Penulisan Laporan Tugas Akhir

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini:

Bab I Pendahuluan

Bab yang berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu

permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

Bab II Dasar Teori

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

Bab III Perancangan Perangkat Lunak

Bab ini membahas tentang desain dan rancangan dari perangkat lunak. Rancangan dan desain meliputi data, proses, arsitektur.

Bab IV Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa *code* yang digunakan untuk proses implementasi.

Bab V Uji Coba Dan Evaluasi

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

Bab VI Kesimpulan Dan Saran

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

Lampiran

Merupakan bab tambahan yang berisi kode-kode sumber yang penting pada aplikasi.

BAB II

TINJAUAN PUSTAKA

Bab ini berisi penjelasan dasra teori dan literatur yang digunakan pada pengimplementasian program. Dasar teori yang dibahas di dalam bab ini adalah penjelasan basis data relasional, sinkronisasi basis data, metode *Change Data Capture*, metode *Routing and Batching*, penjelasan mengenai PostgreSQL, *Java Database Connectivity* (JDBC), serta sistem informasi akademik. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap program yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

2.1 Basis Data Relasional

Basis data adalah kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan yang tidak perlu untuk memenuhi kebutuhan. Data sendiri merupakan fakta mengenai objek, orang, dan lain-lain. Data dinyatakan dengan nilai (angka, deretan karakter, atau symbol). Basis data bertujuan untuk mengatur data sehingga diperoleh kemudahan, ketepatan, dan kecepatan dalam pengambilan data [1].

Sistem basis data memiliki banyak model, salah satunya adalah basis data relasional. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris dan kolom. Untuk membuat hubungan antara dua atau lebih tabel, digunakan *key* (atribut kunci) yaitu *primary key* di salah satu tabel dan *foreign key* di tabel yang lain.

Basis data relasional mudah untuk dilakukan berbagai operasi karena bentuk hubungan antar tabelnya yang sederhana. Tabel 2. 1, Tabel 2. 2, dan Tabel 2. 3 merupakan contoh tabel yang saling berhubungan. Kolom NRP pada Tabel 2. 1

merupakan *primary key* dan kolom KD_MK pada Tabel 2. 2 juga merupakan *primary key*. Kedua *primary key* ini terdapat pula pada Tabel 2. 3 sebagai *foreign key*. Hal inilah yang menyebabkan tabel-tabel ini memiliki hubungan atau relasi.

Tabel 2. 1 Tabel Mahasiswa

NRP	NAMA	ALAMAT	TGL_LAHIR
5111100019	Bustan Amal Firdaus	Pati	29 Agustus 1993
5111100032	Galih Putera Nugraha Suminto	Pati	16 April 1994
5111100050	Tommy Nurwantoro	Lumajang	18 Juni 1993
5111100152	Amanda Tiara Averousi	Bandung	27 Juni 1993
5111100171	Hawari Rahman	Jakarta	3 Januari 1994
5111100191	Salahudin Agung Wijaya	Jakarta	15 April 1993

Tabel 2. 2 Tabel Mata Kuliah

KD_MK	NAMA_MK	SKS
BD012	Basis Data	4
DP001	Dasar Pemrograman	4
PW002	Pemrograman Web	3

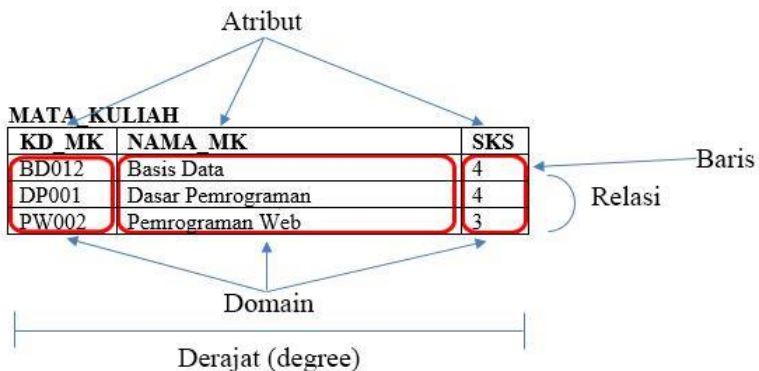
Tabel 2. 3 Tabel Nilai

NRP	KD_MK	NIL_ETS	NIL_EAS
5111100019	BD012	75	80
5111100032	DP001	80	90
5111100050	PW002	85	85
5111100032	PW002	75	80

Berikut ini merupakan penjelasan istilah-istilah dalam basis data relasional yang ditunjukkan pada Gambar 2. 1 [2].

- Relasi : Sebuah tabel yang terdiri dari beberapa kolom dan beberapa baris

- Atribut : Kolom pada sebuah relasi
- Tupel : Baris pada sebuah relasi
- Domain : Kumpulan nilai yang valid untuk satu atau lebih atribut
- Derajat (degree) : Jumlah atribut dalam sebuah relasi
- Cardinality : Jumlah tupel dalam sebuah relasi



Gambar 2. 1 Istilah dalam basis data relasional

2.2 Sinkronisasi Basis Data

Sinkronisasi basis data merupakan sebuah proses untuk menjaga konsistensi data yang dimiliki oleh basis data satu dengan basis data yang lainnya. Sinkronisasi dilakukan untuk menjaga agar data identik dengan data yang lainnya. Dalam proses sinkronisasi basis data, terdapat proses penyalinan data yang disimpan ke dalam suatu tabel dan skema yang berada pada database yang lain. Dengan adanya proses sinkronisasi memungkinkan suatu data yang berada pada basis data tertentu akan dapat diperbarui secara langsung maupun berkala pada basis data yang lain.

Ada beberapa hal yang harus diperhatikan selama proses sinkronisasi database berlangsung antara lain.

1. Kesamaan antara struktur data yang akan dilakukan proses sinkronisasi.
2. Frekuensi proses sinkronisasi yang dilakukan.
3. Situasi kompleks dimana kedua buah database dilakukan proses update data secara bersamaan [3].

2.3 Metode *Change Data Capture*

Change Data Capture adalah metode yang sudah dikembangkan oleh pengembang SymetricDS. Metode ini mencatat perubahan data yang terjadi dari sebuah basis data berupa *insert*, *update*, dan *delete* ke dalam tabel SYM_DATA. Gambaran umum metode ini adalah seperti pada Gambar 2. 2 [4].

MAHASISWA			
NRP	NAMA	ALAMAT	TGL_LAHIR
5111100050	Tommy Nurwantoro	Lumajang	18 Juni 1993

MATA KULIAH		
KD_MK	NAMA_MK	SKS
BD012	Basis Data	4

SYM_DATA					
data_id	table_name	event_type	row_data	pk_data	old_data
1	mahasiswa	I	5111100050,"Tommy Nurwantoro","Lumajang",4		
2	mata_kuliah	I	"BD012","Basis Data",3		
3	mata_kuliah	U	"BD012","Basis Data",4	BD012	"BD012","Basis Data",3
4	mata_kuliah	D		BD012	"BD012","Basis Data",4

Gambar 2. 2 *Change Data Capture*

Gambar 2. 2 merupakan contoh penerapan metode *Change Data Capture* dimana terjadi masukan data pada tabel MAHASISWA yang kemudian dicatat ke dalam tabel SYM_DATA berdasarkan *event type* I sebagai *event input*. Begitu juga pada tabel MATA_KULIAH terjadi *event* I sebagai *event input*, U sebagai *event update*, dan D sebagai *event delete*. Perubahan data yang terjadi juga dicatat pada kolom *row_data* untuk data baru

sedangkan *old_data* untuk data lama dan *pk_data* menunjukkan data *primary key* pada baris tersebut.

2.4 Metode *Routing and Batching*

Metode ini adalah metode yang digunakan pada perangkat lunak SymetricDS dimana perubahan data yang terjadi pada basis data dikumpulkan ke dalam suatu kumpulan yang di namakan *batch*. *Batch* akan membagi datanya menjadi beberapa bagian yang disebut *node*. *Node* ini yang satu persatu akan dikirimkan ke basis data lain. Perubahan data dimasukkan ke dalam *batch* yang sama hingga memenuhi kapasitas maksimum *batch* tersebut. Penempatan data ke dalam *batch* dicatat pada tabel SYM_DATA_EVENT beserta ID *router*. Sedangkan penempatan *node* ke dalam *batch* dicatat pada tabel SYM_OUTGOING_BATCH. Penjelasan dalam bentuk tabel seperti pada Gambar 2. 3 [4].

SYM_DATA_EVENT

data_id	batch_id	router_id
1	101	12

MAHASISWA

batch_id	node_id	status	error_flag
101	00001	OK	0
101	00002	OK	0

Gambar 2. 3 Routing and Batching

Pada Gambar 2. 3, tabel SYM_DATA_EVENT berisi 1 baris data dimana data tersebut merupakan informasi pengiriman data berdasarkan rute pengirimannya. Tabel MAHASISWA berisi 2 baris data yang merupakan informasi bagian pengiriman yang dicatat pada tabel SYM_DATA_EVENT dengan status pengiriman data juga dicatat pada kolom status.

2.5 PostgreSQL

PostgreSQL merupakan *object-relational database management system* (ORDBMS). PostgreSQL adalah *open source* sistem basis data relasional yang sangat powerful. PostgreSQL dapat berjalan di semua sistem operasi. PostgreSQL mendukung penuh *foreign keys, joins, views, triggers, dan stored procedures*. PostgreSQL memiliki hampir semua data *type* SQL92 dan SQL99, termasuk *integer, numeric, boolean, char, varchar, data, interval, dan timestamp*. PostgreSQL juga memiliki kemampuan menyimpan objek *binary* yang cukup besar, termasuk gambar, suara, dan video. Selain itu PostgreSQL memiliki *native programming interface* untuk C/C++, Java, Perl, Python, Ruby, Tcl, ODBC, dll.

PostgreSQL memiliki fitur-fiturnya yang mutakhir seperti yang dijelaskan pada Tabel 2. 4.

Tabel 2. 4 Tabel Deskripsi Fitur PostgreSQL

No	Fitur	Deskripsi
1	<i>Multi Version Concurrency Control (MVCC)</i>	Fitur yang digunakan untuk menyediakan akses secara bersamaan ke basis data dan transaksi pemrograman.
2	<i>Point in time recovery</i>	Fitur yang dapat mengembalikan atau memulihkan data pada waktu tertentu
3	<i>Tablespaces</i>	Fitur penyimpanan data objek
4	<i>Asynchronous replication</i>	Fitur replikasi yang tidak mengganggu jalannya proses replikasi ketika PostgreSQL melakukan penyimpanan data sendiri.
5	<i>Nested transactions (savepoints)</i>	Fitur yang memungkinkan sebuah transaksi dijalankan di dalam transaksi lainnya.
6	<i>online/hot backups</i>	Fitur yang memungkinkan untuk melakukan <i>backup</i> meskipun data masih aktif digunakan oleh pengguna.
7	<i>Sophisticated</i>	Fitur yang dapat mengoptimalkan proses

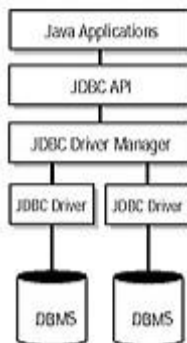
	<i>query planner/optimizer</i>	menjalankan query.
8	<i>Write ahead logging for fault tolerance</i>	Fitur pencatatan modifikasi ke dalam <i>log</i> sebelum aktifitas dijalankan.

PostgreSQL juga mendukung *international character sets*, *multibyte character encodings*, *Unicode*, and *locale-aware for sorting, case-sensitivity*, dan *formatting*. Selain itu, PostgreSQL dapat diukur baik kualitas datanya maupun jumlah *concurrent user* yang dapat ditangani [5].

2.6 Java Database Connectivity (JDBC)

JDBC merupakan Java API (*Application Programming Interface*) yang digunakan untuk melakukan koneksi ke basis data melalui bahasa pemrograman Java [6]. Dengan menggunakan JDBC API, pengembang dapat mengakses ke beberapa jenis basis data dengan satu bahasa Java. Selain itu pengembang juga dapat mengubah *host* dan *port* pada basis data yang digunakan [7].

Tujuan dibuatnya JDBC adalah untuk membuat sebuah basis data yang generik sebagai antar muka untuk berhubungan dengan sumber data yang lainnya. JDBC memiliki arsitektur seperti ditunjukkan pada Gambar 2. 4.



Gambar 2. 4 Arsitektur JDBC

Pada Gambar 2. 4 dapat ditunjukkan bahwa koneksi yang menghubungkan basis data adalah JDBC *Driver Manager*. *Driver Manager* membuka koneksi dan mengakses basis data menggunakan JDBC *Driver*. Oleh karena itu JDBC *Driver* harus terdaftar di dalam *Driver Manager*. JDBC menggunakan URL (*Uniform Resource Locator*) *syntax* untuk menentukan koneksi basis data. Contoh URL yang digunakan pada PostgreSQL adalah sebagai berikut [6].

- jdbc:postgresql://localhost/databasename

2.7 Sistem Informasi Akademik

Sistem informasi adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen [8]. Dengan memanfaatkan teknologi yang ada, aktivitas operasi manajemen akan semakin tertata dan berjalan sesuai aturan yang ada. Maka dapat diartikan bahwa sistem informasi akademik adalah pemanfaatan teknologi informasi dalam mendukung operasi dan manajemen aktivitas dalam lingkup akademik. Sistem informasi akademik merupakan alat bantu kegiatan pembelajaran yang sangat bermanfaat. Dengan sistem informasi akademik, civitas akademik dapat melihat segala yang berhubungan dengan pendidikannya dengan mudah.

Sistem informasi akademik tidak berdiri sendiri, melainkan berhubungan dengan sistem informasi lainnya. Data pegawai, keuangan, inventaris, organisasi, semua berasal dari sistem informasi yang berbeda. Namun untuk menjalankan sistem informasi akademik, mutlak dibutuhkan data-data tersebut.

Data-data ini berasal dari luar sistem informasi akademik, namun sistem informasi akademik membutuhkan data-data ini untuk menjalankan berbagai aktifitas sistem. Oleh karena itu data-data dari luar sistem informasi akademik ini perlu untuk disinkronisasikan secara berkala agar mendapatkan data terbaru

dan valid. Sehingga aktifitas sistem informasi akademik dapat berjalan sesuai dengan semestinya.

BAB III

DESAIN PERANGKAT LUNAK

Pada bab ini dibahas analisis kebutuhan, proses bisnis dan desain dari perangkat lunak yang dibangun dalam tugas akhir ini. Bagian awal bab akan dibahas tentang analisis permasalahan dan kebutuhan masyarakat. Berikutnya dibahas fungsional sistem yang berdasarkan hasil analisis kebutuhan. Bagian berikutnya akan dibahas rancangan perangkat lunak yang ditujukan untuk memberikan gambaran tentang perangkat lunak yang dibuat.

3.1 Analisis

Pada sub bab berikut akan dijelaskan analisis pembuatan kerangka kerja sinkronisasi basis data relasional pada sistem informasi akademik berbasis web. Analisis yang dilakukan meliputi analisis permasalahan, kebutuhan umum perangkat lunak, deskripsi umum sistem, arsitektur dan kebutuhan fungsional sistem.

3.1.1 Analisis Permasalahan

Sistem informasi akademik merupakan sistem informasi yang menangani segala hal yang berlangsung di dalam kegiatan akademik. Mulai dari kegiatan belajar mengajar hingga status dosen pengajar atau pendidik dan mahasiswa. Dengan kata lain, sistem informasi akademik menggunakan data yang sangat banyak. Sedangkan data-data tersebut bisa diperoleh dari sistem informasi lain yang khusus mengolah jenis data tersebut. Contohnya data pendidik adalah data yang didapatkan dari sistem informasi kepegawaian. Status pembayaran mahasiswa juga didapatkan dari sistem informasi pembayaran.

Dengan data yang saling berhubungan ini, sangat mungkin terjadi perubahan data yang mempengaruhi data yang lainnya. Misalkan terjadi perubahan data di sistem informasi kepegawaian

pada salah satu pendidik maka pada sistem informasi akademik juga harus diubah. Jika perubahan data hanya satu, maka mudah untuk menyalin data tersebut ke sistem informasi akademik. Namun jika perubahan datanya sangat banyak hingga sulit untuk diamati, maka penyalinan data secara manual tidak efektif.

Dengan perkembangan teknologi informasi, kondisi-kondisi seperti ini dapat diselesaikan dengan memanfaatkan kerangka kerja tambahan yang dapat mendeteksi perubahan data pada basis data untuk dikirimkan ke basis data lain yang sudah diatur sebelumnya. Dengan memanfaatkan metode *change data capture*, kerangka kerja ini dapat mencatat setiap perubahan data pada tabel yang telah dipilih untuk dikirim datanya ke tabel pada basis data yang lain.

3.1.2 Deskripsi Umum Perangkat Lunak

Kerangka kerja yang dibangun pada tugas akhir ini bernama Sinkbd. Sinkbd merupakan kependekan dari sinkronisasi basis data. Sinkbd adalah kerangka kerja yang dibangun menggunakan bahasa pemrograman *Java* dan memanfaatkan metode *change data capture* untuk melakukan pencatatan perubahan data.

Pencatatan perubahan data terjadi akibat *trigger* yang sudah dibuat oleh Sinkbd pada tabel yang telah dipilih. *Trigger* ini dibuat berdasarkan pengaturan yang dibuat pengguna melalui halaman web yang sudah disediakan. Pengaturan yang dibuat pengguna disimpan ke dalam *file* xml. Seluruh pencatatan perubahan data disimpan ke dalam tabel yang bernama *sink_data*. Sinkbd mengambil informasi yang ada di dalam tabel *sink_data* untuk dikirimkan ke basis data lain.

Dalam pengiriman data, Sinkbd tidak melakukan pengiriman data begitu saja. Sinkbd melakukan dua kali pencatatan dan pemeriksaan data agar tidak terjadi kesalahan pada pengiriman data. Pengiriman data dilakukan pada satu waktu tertentu sehingga perubahan data yang terjadi selama pengiriman data

berlangsung tidak akan mempengaruhi data yang sedang dikirimkan, namun perubahan tersebut tetap dicatat untuk pengiriman data selanjutnya.

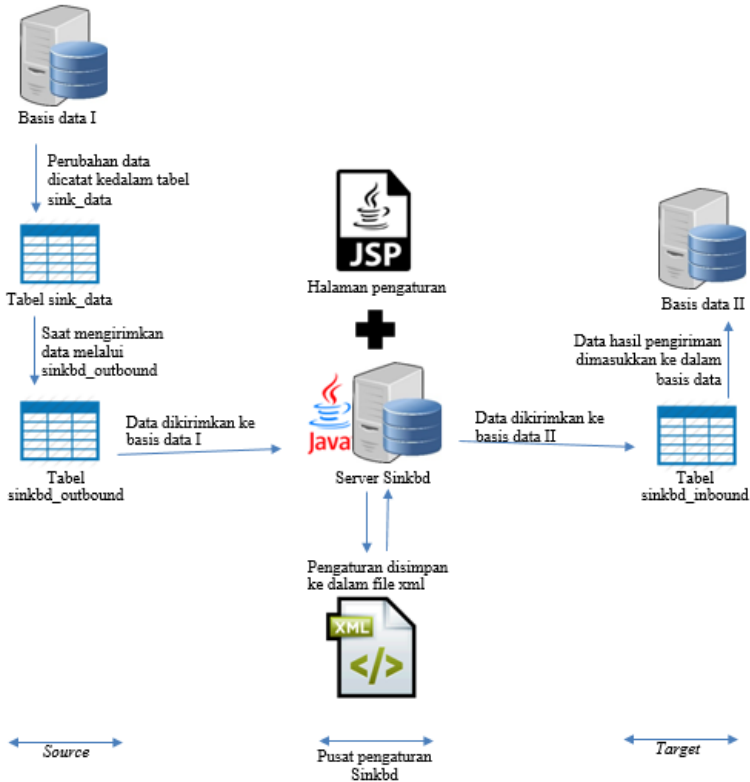
Sinkbd dibangun dengan menggunakan *Spring Web Application* sebagai antar muka pengguna sekaligus halaman untuk membuat serta melihat pengaturan yang dibuat. Sinkbd juga dibangun menggunakan PostgreSQL sebagai tempat menyimpan status pengaturan serta mencatat pengiriman data masuk dan keluar. Sinkbd juga memanfaatkan *Java Database Connectivity* untuk membuat koneksi ke basis data untuk memasukkan, merubah, dan menghapus data pada basis data yang disambungkan.

Fungsionalitas Sinkbd apabila diringkas adalah sebagai berikut.

1. Sinkbd dapat membuat, mengubah, dan menghapus pengaturan sinkronisasi.
2. Sinkbd dapat menampilkan semua pengaturan yang sudah dibuat.
3. Sinkbd dapat mengirimkan isi tabel dari basis data sumber ke basis data tujuan.
4. Sinkbd dapat memeriksa kesamaan data yang dikirim dari basis data sumber dengan yang sudah diterima di basis data tujuan.
5. Sinkbd dapat melakukan pencatatan sinkronisasi yang berhasil maupun gagal.

3.1.3 Arsitektur Sistem

Berikut ini adalah arsitektur sistem dari Sinkbd.



Gambar 3. 1 Arsitektur Sistem

Kerangka kerja Sinkbd menggunakan arsitektur seperti pada Gambar 3. 1. Sistem ini dibagi menjadi 3 bagian yaitu.

1. *Source*

Merupakan bagian yang menangani sumber data yang akan dikirimkan. *Source* terdiri dari *trigger*, tabel *sink_data*, dan tabel *sinkbd_outbound*. *Trigger* berperan mendeteksi perubahan data yang terjadi

pada tabel. Tabel *sink_data* berperan menampung semua perubahan data yang dideteksi oleh *trigger*. Tabel *sink_data* merupakan penerapan dari metode *Change Data Capture*. Tabel *sinkbd_outbound* berperan menampung data dari tabel *sink_data* yang akan dikirimkan ke luar. Tabel *sinkbd_outbound* merupakan pengembangan dari metode *Routing and Batching*.

2. Pusat pengaturan Sinkbd

Merupakan bagian yang menangani segala hal yang berhubungan dengan pengaturan sinkronisasi. Pusat pengaturan Sinkbd terdiri dari tampilan antar muka pengguna, tabel *sink_bd*, tabel *sinkbd_inbound*, dan *file xml*. Tampilan antar muka pengguna dibangun menggunakan *Spring Web Application*. Tampilan antar muka ini dibutuhkan bagi pengguna untuk mengelola pengaturan sinkronisasi. Pengguna dapat membuat, mengubah, menghapus, dan menjalankan pengaturan melalui antar muka ini. Kemudian tabel *sink_bd* berperan mencatat waktu terakhir sinkronisasi dilakukan. Tabel *sinkbd_inbound* berperan mencatat data yang dikirimkan dari *source* untuk dilakukan pemeriksaan dan kemudian dikirimkan ke *target*. *File xml* merupakan pengaturan sinkronisasi yang sudah dibuat. *File xml* ini berisi tabel apa saja yang akan disinkronisasikan.

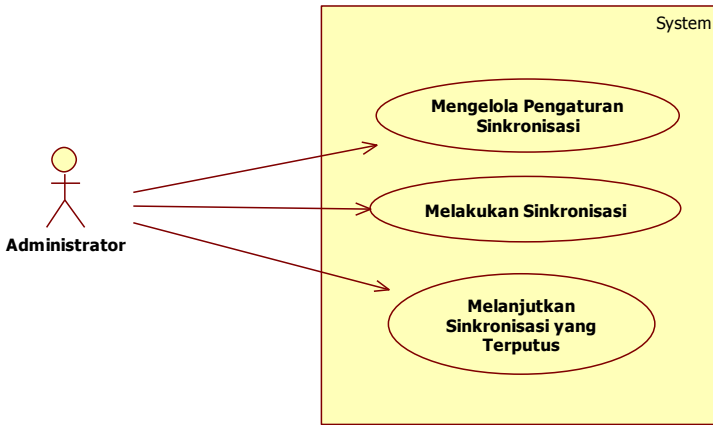
3. Target

Merupakan bagian yang menangani tujuan pengiriman data. Target hanya terdiri dari tabel *sinkbd_inbound*. Tabel *sinkbd_inbound* memiliki fungsi yang sama dengan yang ada pada pusat

pengaturan Sinkbd, namun data yang sudah dilakukan pemeriksaan akan dimasukkan ke dalam tabel sesuai dengan pengaturan yang sudah dibuat oleh pengguna.

3.1.4 Kebutuhan Fungsional Aplikasi

Kebutuhan fungsional kerangka kerja Sinkbd digambarkan pada Gambar 3. 2.



Gambar 3. 2 Diagram Kasus Penggunaan Sinkbd

Hak akses penggunaan kerangka kerja Sinkbd hanya diberikan kepada Administrator basis data. Administrator dapat mengelola pengaturan sinkronisasi, melakukan sinkronisasi, dan melanjutkan sinkronisasi yang terputus. Penjelasan lengkap mengenai kasus penggunaan pada kerangka kerja Sinkbd berada pada Tabel 3. 1.

Tabel 3. 1 Deskripsi Kasus Penggunaan Sinkbd

No	Kode	Nama	Keterangan
1	UC-W01	Mengelola Pengaturan Sinkronisasi	Administrator dapat mengelola data pengaturan yaitu

			menambah, mengubah, serta menghapus pengaturan.
2	UC-W02	Melakukan Sinkronisasi	Administrator dapat memulai sinkronisasi berdasarkan pengaturan yang sudah dibuat
3	UC-W03	Melanjutkan Sinkronisasi yang Terputus	Administrator dapat melanjutkan sinkronisasi apabila terjadi koneksi putus ketika sinkronisasi sedang berlangsung.

3.1.4.1 Deskripsi Kasus Kebutuhan UC-W01

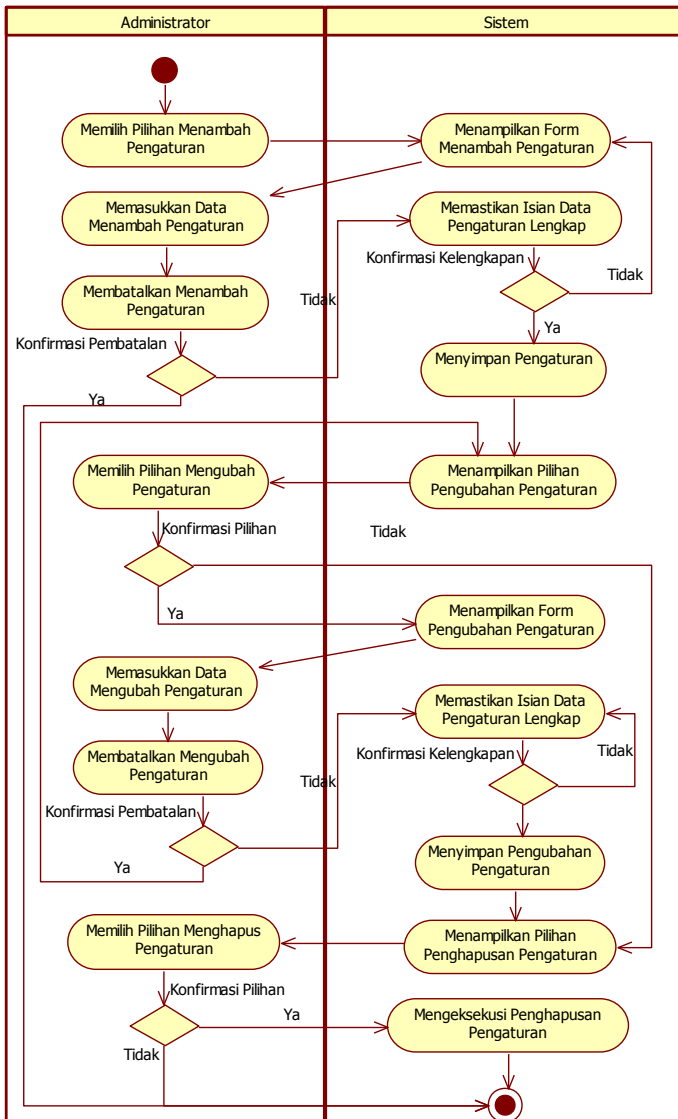
Kasus kebutuhan kode UC-W01 merupakan kasus kebutuhan mengelola pengaturan sinkronisasi. Rincian alur kasus mengelola pengaturan sinkronisasi dijelaskan pada Tabel 3. 2 dan diagram aktifitas kasus yang dijelaskan pada Gambar 3. 3.

Tabel 3. 2 Rincian Alur Kasus Penggunaan UC-W01

Nama Use Case	Mengelola pengaturan sinkronisasi
Nomor	UC-W01
Aktor	Administrator
Kondisi Awal	Data pengaturan belum terisi dan <i>form</i> isian belum tampil
Kondisi akhir	Informasi data tempat tersimpan
Alur Normal	<ol style="list-style-type: none"> 1. Administrator memilih pilihan menambah pengaturan. 2. Sistem menampilkan <i>form</i> menambah pengaturan. 3. Administrator memasukkan data ke dalam <i>form</i> menambah pengaturan. <p>A1. Administrator membatalkan</p>

	<p>menambah pengaturan.</p> <p>4. Sistem memastikan isian data pengaturan lengkap. A2. Sistem mendeteksi isian data pengaturan tidak lengkap.</p> <p>5. Sistem menyimpan pengaturan.</p> <p>6. Sistem menampilkan pilihan perubahan pengaturan. A3. Administrator tidak memilih pilihan perubahan pengaturan.</p> <p>7. Administrator memilih pilihan mengubah pengaturan.</p> <p>8. Sistem menampilkan form perubahan pengaturan.</p> <p>9. Administrator mengisi <i>form</i> perubahan pengaturan. A4. Administrator membatalkan proses perubahan pengaturan.</p> <p>10. Sistem memastikan isian data pengaturan lengkap. A5. Sistem mendeteksi isian data pengaturan tidak lengkap.</p> <p>11. Sistem mengeksekusi perubahan pengaturan.</p> <p>12. Sistem menampilkan pilihan penghapusan pengaturan. A6. Administrator tidak memilih pilihan penghapusan pengaturan.</p> <p>13. Administrator memilih pilihan penghapusan pengaturan. A7. Administrator membatalkan proses penghapusan pengaturan.</p> <p>14. Sistem mengeksekusi penghapusan</p>
--	--

	<p>pengaturan.</p> <p>15. Kasus penggunaan mengelola pengaturan sinkronisasi berakhir.</p>
	<p>A1. Administrator membatalkan pembuatan pengaturan. A1.1. Menuju alur normal nomor 15.</p> <p>A2. Sistem mendeteksi isian data pengaturan tidak lengkap A2.1. Menuju alur normal nomor 2.</p> <p>A3. Administrator tidak memilih pilihan pengubahan pengaturan. A3.1. Menuju alur normal nomor 12.</p> <p>A4. Administrator membatalkan proses perubahan pengaturan. A4.1. Menuju alur normal nomor 6.</p> <p>A5. Sistem mendeteksi isian data pengaturan tidak lengkap. A5.1. Menuju alur normal nomor 8.</p> <p>A6. Administrator tidak memilih pilihan penghapusan pengaturan. A6.1. Menuju alur normal nomor 15.</p> <p>A7. Administrator membatalkan proses penghapusan data. A7.1. Menuju alur normal nomor 12.</p>



Gambar 3. 3 Diagram Aktifitas Kasus Penggunaan UC-W01

3.1.4.2 Deskripsi Kasus Kebutuhan UC-W02

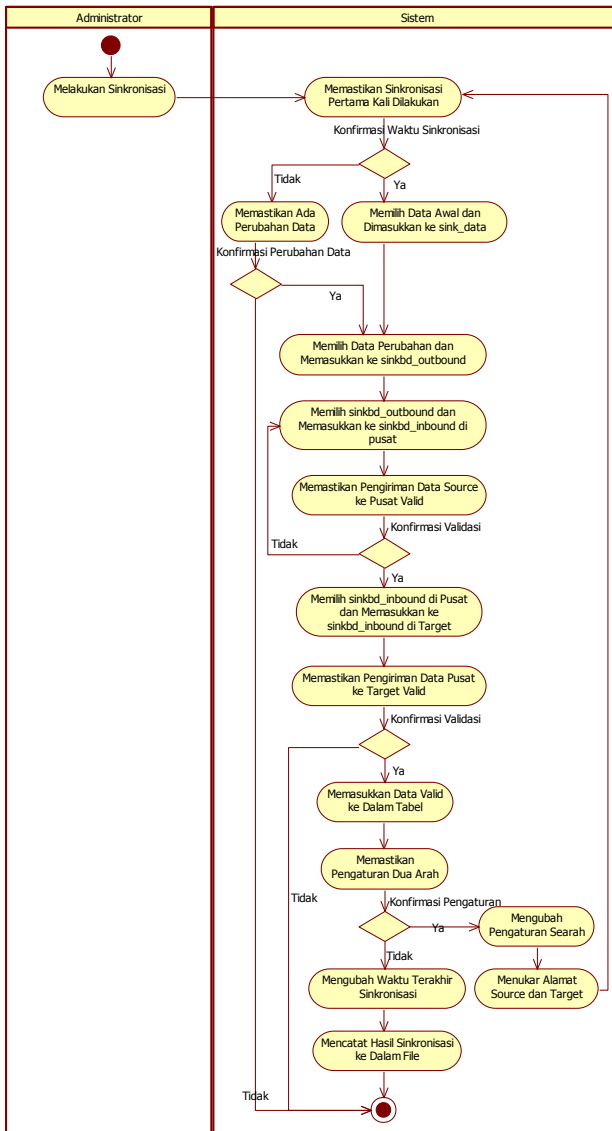
Kasus kebutuhan kode UC-W02 merupakan kasus kebutuhan melakukan sinkronisasi. Rincian alur kasus melakukan sinkronisasi dijelaskan pada Tabel 3. 3 dan diagram aktifitas kasus yang dijelaskan pada Gambar 3. 4.

Tabel 3. 3 Rincian Alur Kasus Penggunaan UC-W02

Nama Use Case	Melakukan sinkronisasi
Nomor	UC-W02
Aktor	Administrator
Kondisi Awal	Data belum ada yang dikirimkan dari <i>source</i> menuju <i>target</i>
Kondisi akhir	Data yang sudah valid dimasukkan ke dalam tabel.
Alur Normal	<ol style="list-style-type: none"> 1. Administrator memilih lakukan sinkronisasi. 2. Sistem memastikan sinkronisasi pertama kali dilakukan. A1. Sistem mendeteksi sinkronisasi pertama kali dilakukan. 3. Sistem memastikan ada perubahan data. A2. Sistem mendeteksi tidak ada perubahan data 4. Sistem memilih data perubahan yang ada pada tabel <i>sink_data</i> berdasarkan waktu terakhir sinkronisasi dan memasukkan ke dalam tabel <i>sinkbd_outbound</i> di <i>source</i>. 5. Sistem memilih semua data pada <i>sinkbd_outbound</i> di <i>source</i> dan memasukkan ke dalam tabel <i>sinkbd_inbound</i> di basis data pusat. 6. Sistem memastikan data pada <i>sinkbd_outbound</i> di <i>source</i> sama

	<p>dengan data pada sinkbd_inbound di pusat.</p> <p>A3. Sistem mendeteksi data pada sinkbd_outbound di <i>source</i> tidak sama dengan data pada sinkbd_inbound di pusat.</p> <p>7. Sistem memilih seluruh data pada tabel sinkbd_inbound di pusat dan memasukkan data ke dalam tabel sinkbd_inbound di <i>target</i>.</p> <p>8. Sistem memastikan data di sinkbd_inbound di pusat sama dengan data di sinkbd_inbound di <i>target</i>.</p> <p>A4. Sistem mendeteksi data pada sinkbd_inbound di pusat tidak sama dengan data pada sinkbd_inbound di <i>target</i>.</p> <p>9. Sistem memasukkan data valid ke dalam tabel di <i>target</i>.</p> <p>10. Sistem memastikan pengaturan sinkronisasi satu arah.</p> <p>A5. Sistem mendeteksi pengaturan sinkronisasi dua arah.</p> <p>11. Sistem mengubah waktu terakhir sinkronisasi menjadi waktu sekarang.</p> <p>12. Sistem mencatat hasil sinkronisasi ke dalam sebuah <i>file</i>.</p> <p>13. Kasus penggunaan melakukan sinkronisasi berakhir.</p>
	<p>A1. Sistem mendeteksi sinkronisasi bukan pertama kali dilakukan</p> <p>A1.1. Sistem memilih data awal dari</p>

	<p>tabel yang ada di <i>source</i> dan dimasukkan ke dalam tabel <i>sink_data</i>.</p> <p>A1.2. Menuju alur normal nomor 4.</p> <p>A2. Sistem mendeteksi tidak ada perubahan data.</p> <p>A2.1. Menuju alur normal nomor 11.</p> <p>A3. Sistem mendeteksi data pada <i>sinkbd_outbound</i> di <i>source</i> tidak sama dengan data pada <i>sinkbd_inbound</i> di pusat.</p> <p>A3.1. Sistem mengubah nilai validasi terakhir sesuai index baris yang sudah diperiksa.</p> <p>A3.2. Menuju alur normal nomor 4.</p> <p>A4. Sistem mendeteksi data pada <i>sinkbd_inbound</i> di pusat tidak sama dengan data pada <i>sinkbd_inbound</i> di <i>target</i>.</p> <p>A4.1. Sistem mengubah nilai validasi terakhir sesuai index baris yang sudah diperiksa.</p> <p>A4.2. Menuju alur normal nomor 11.</p> <p>A5. Sistem mendeteksi pengaturan sinkronisasi dua arah.</p> <p>A5.1. Sistem mengubah pengaturan menjadi satu arah</p> <p>A5.2. Sistem menukar alamat <i>source</i> dan <i>target</i>.</p> <p>A5.3. Menuju alur normal nomor 2</p>
--	---



Gambar 3. 4 Diagram Aktifitas Kasus Penggunaan UC-W02

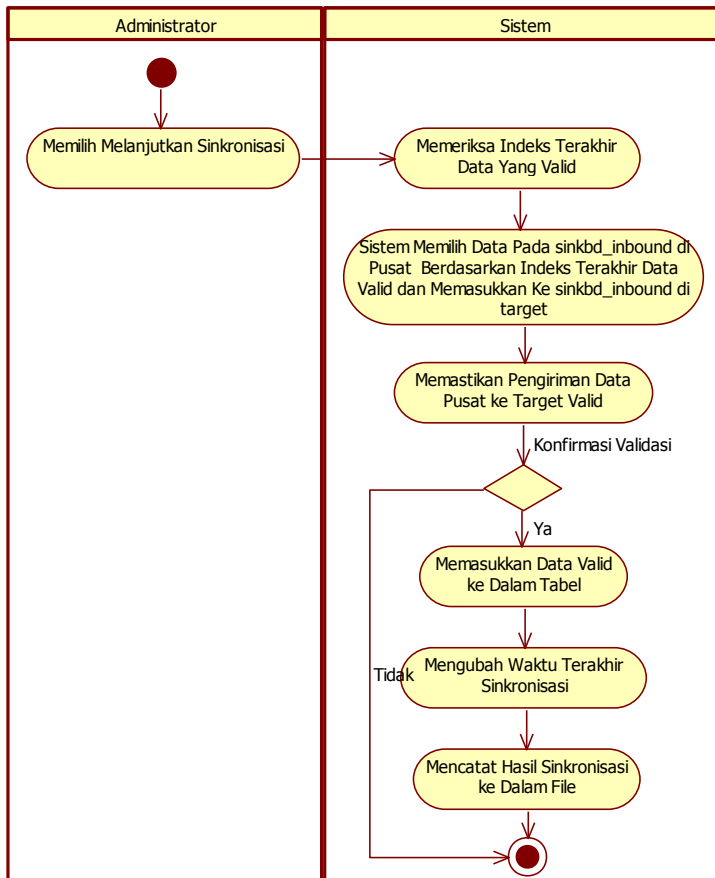
3.1.4.3 Deskripsi Kasus Kebutuhan UC-W03

Kasus kebutuhan kode UC-W03 merupakan kasus kebutuhan melanjutkan sinkronisasi yang terputus. Rincian alur kasus melanjutkan sinkronisasi yang terputus dijelaskan pada Tabel 3. 4 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 5.

Tabel 3. 4 Rincian Alur Kasus Penggunaan UC-W03

Nama Use Case	Melanjutkan sinkronisasi yang terputus
Nomor	UC-W03
Aktor	Administrator
Kondisi Awal	Data yang dikirimkan dari pusat ke <i>target</i> tidak valid
Kondisi akhir	Data yang sudah valid dimasukkan ke dalam tabel.
Alur Normal	<ol style="list-style-type: none"> 1. Administrator memilih melanjutkan sinkronisasi. 2. Sistem memeriksa indeks terakhir data yang valid. 3. Sistem memilih data pada tabel <i>sinkbd_inbound</i> di pusat berdasarkan indeks terakhir data valid dan memasukkan data ke dalam tabel <i>sinkbd_inbound</i> di <i>target</i>. 4. Sistem memastikan data di <i>sinkbd_inbound</i> di pusat sama dengan data di <i>sinkbd_inbound</i> di <i>target</i>. A1. Sistem mendeteksi data pada <i>sinkbd_inbound</i> di pusat tidak sama dengan data pada <i>sinkbd_inbound</i> di <i>target</i>. 5. Sistem memasukkan data valid ke dalam tabel di <i>target</i>. 6. Sistem mengubah waktu terakhir sinkronisasi menjadi waktu sekarang.

	<p>7. Sistem mencatat hasil sinkronisasi ke dalam sebuah <i>file</i>.</p> <p>8. Kasus penggunaan melakukan sinkronisasi berakhir.</p>
	<p>A1. Sistem mendeteksi data pada <i>sinkbd_inbound</i> di pusat tidak sama dengan data pada <i>sinkbd_inbound</i> di <i>target</i>.</p> <p>A1.1. Sistem mengubah nilai validasi terakhir sesuai index baris yang sudah diperiksa.</p> <p>A4.2. Menuju alur normal nomor 6.</p>



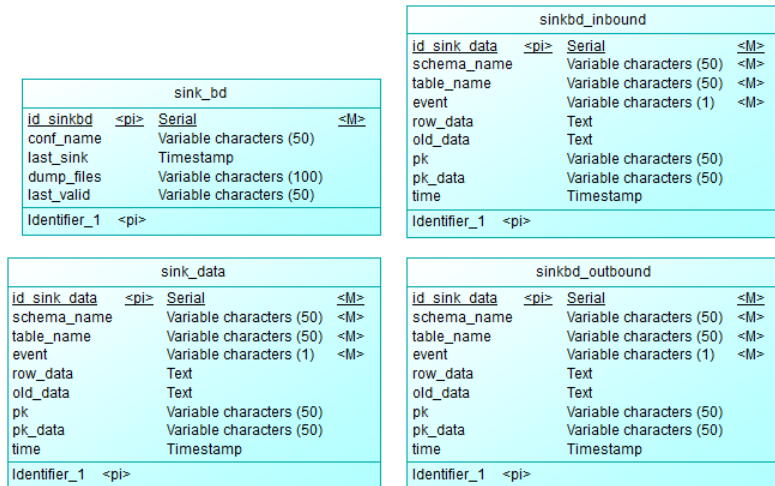
Gambar 3. 5 Diagram Aktifitas Kasus Penggunaan UC-W02

3.2 Perancangan

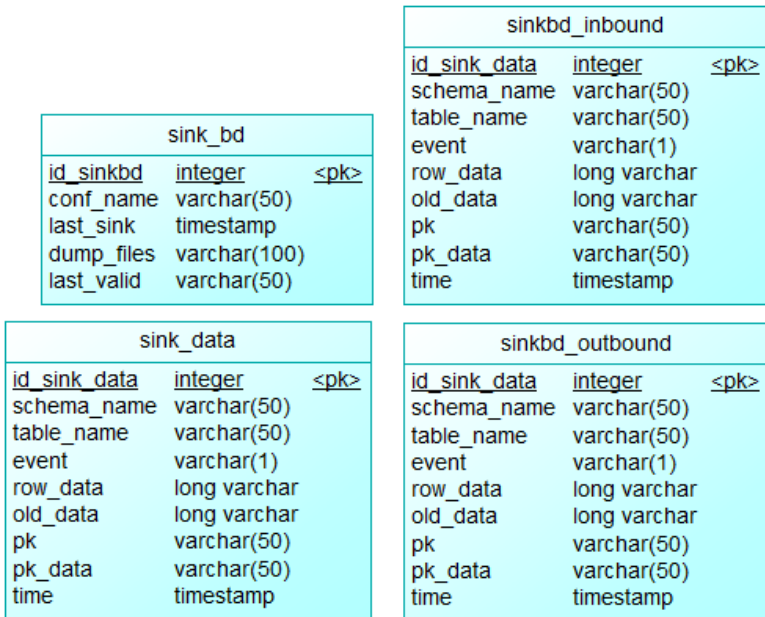
Sub bab berikut membahas tentang perancangan dari kerangka kerja Sinkbd. Sub bab berikut membahas rancangan basis data, rancangan proses, dan rancangan antarmuka.

3.2.1 Perancangan Basis Data

Pada sub bab ini akan dijelaskan bagaimana rancangan basis data yang digunakan pada kerangka kerja Sinkbd. Basis data yang digunakan pada sistem yang dibangun pada tugas akhir ini menggunakan sistem manajemen basis data relasional PostgreSQL. PostgreSQL digunakan untuk menyimpan keterangan pengaturan sinkronisasi dan penyimpanan pengiriman data. *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM) dari basis data sistem ini dapat dilihat pada Gambar 3. 6 dan Gambar 3. 7.



Gambar 3. 6 Conceptual Data Model (CDM) Kerangka Kerja Sinkbd



Gambar 3. 7 Physical Data Model (PDM) Kerangka Kerja Sinkbd

3.2.1.1 Rancangan Tabel sink_bd

Tabel sink_bd digunakan untuk menyimpan status pengaturan sinkronisasi basis data. Detail atribut tabel sink_bd dijelaskan pada Tabel 3. 5.

Tabel 3. 5 Atribut Tabel sink_bd

Nama Kolom	Tipe Data	Keterangan
id_sink_bd	Integer	<i>Primary Key</i> dari tabel sink_bd
conf_name	Varchar (50)	Nama pengaturan
last_sink	Timestamp	Waktu terakhir sinkronisasi dilakukan pada

		pengaturan ini
dump_files	Varchar (100)	Nama file hasil dump tabel sebelum data valid dimasukkan
last_valid	Varchar (50)	Index baris terakhir data dinyatakan valid. Berisi 0 apabila data yang dikirimkan sudah valid

3.2.1.2 Rancangan Tabel sink_data, sinkbd_inbound, dan sinkbd_outbound

Tabel sink_data, sinkbd_inbound, dan sinkbd_outbound mempunyai rancangan yang sama. Tabel sink_data digunakan untuk menyimpan perubahan data oleh *trigger* yang sudah dibuat pada tabel tertentu. Tabel sinkbd_inbound dan sinkbd_outbound digunakan untuk menyimpan pengiriman data ketika sinkronisasi dilakukan. Detail atribut ketiga tabel ini dijelaskan pada Tabel 3.6.

Tabel 3.6 Atribut Tabel sink_data, sinkbd_inbound, dan sinkbd_outbound

Nama Kolom	Tipe Data	Keterangan
id_sink_data	Integer	<i>Primary Key</i> dari tabel
schema_name	Varchar (50)	Nama skema tabel
table_name	Varchar (50)	Nama tabel

Event	Varchar(1)	Nama event trigger. Dapat berisi I, U, dan D maksudnya Insert, Update, Delete
row_data	Text	Satu baris data baru setelah terjadi perubahan data pada tabel
old_data	Text	Satu baris data lama sebelum terjadi perubahan data pada tabel
pk	Varchar (50)	Nama kolom <i>primary key</i> pada tabel
pk_data	Varchar(50)	Data kolom <i>primary key</i> pada tabel
time	Timestamp	Waktu perubahan data terjadi

3.2.2 Perancangan Proses

Pada subbab ini akan dibahas secara mendetail dari rancangan proses Sinkbd untuk memenuhi kebutuhan fungsionalnya.

3.2.2.1 Proses Menambah Pengaturan

Proses menambah pengaturan merupakan proses yang dilakukan ketika Administrator membutuhkan pengaturan

sinkronisasi untuk mengirim data dari basis data satu ke basis data lainnya. Pengaturan baru dibuat melalui antarmuka Sinkbd. Pengaturan disimpan ke dalam *file* xml untuk kemudian dibaca ketika melakukan sinkronisasi. Perancangan struktur file xml yang akan digunakan ditunjukkan pada gambar Gambar 3. 8.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<configuration name="" reverse="">
  <source>
    <address></address>
    <username></username>
    <password></password>
    <table-list></table-list>
    <tables>
      <table name="">
        <column-list></column-list>
        <column-unselected/>
        <column-selected></column-selected>
      </table>
    </tables>
  </source>
  <target>
    <address></address>
    <username></username>
    <password></password>
    <table-list></table-list>
    <tables>
      <table name="">
        <column-list></column-list>
        <column-unselected/>
        <column-selected></column-selected>
      </table>
    </tables>
  </target>
</configuration>
```

Gambar 3. 8 Rancangan Struktur *File* xml

Pada Gambar 3. 8, *file xml* terdiri dari 2 bagian yaitu *source* dan *target*. Kedua bagian ini memiliki struktur sub bagian yang sama namun isinya berbeda. Sub bagiannya terdapat alamat basis data, *username*, *password*, tabel apa saja yang tersedia, serta kolom yang dipilih untuk disinkronisasi.

Pengaturan yang ditambahkan juga dicatat pada basis data milik kerangka kerja Sinkbd. Pengaturan ini disimpan di dalam tabel *sink_bd*. Tabel ini menampung nama pengaturan beserta waktu terakhir pengaturan ini dilakukan sinkronisasi.

Pada akhir proses menambah pengaturan ini, kerangka kerja Sinkbd memasang *trigger* dan membuat beberapa tabel di basis data *source* dan *target*. *Trigger* dan tabel dibuat pada kedua bagian untuk mengantisipasi pengaturan dua arah sehingga basis data *target* juga bisa menjadi sumber data. *Trigger* yang dipasang berfungsi mencatat perubahan data apa saja yang terjadi. Tabel yang dibuat adalah tabel *sink_data* untuk menerapkan metode *Change Data Capture*, tabel *sinkbd_inbound* untuk mengantisipasi sinkronisasi dua arah, dan tabel *sinkbd_outbound* untuk menerapkan metode *Routing and Batching*. Semua perubahan data disimpan di tabel *sink_data*.

3.2.2.2 Proses Mengubah Pengaturan

Proses mengubah pengaturan merupakan proses yang dilakukan ketika informasi pengaturan akan diubah. Seluruh isi pengaturan dapat diubah kecuali nama pengaturan tidak dapat diubah. Perubahan pengaturan dilakukan melalui antarmuka Sinkbd. Pengaturan yang sudah diubah akan disimpan kembali ke dalam *file xml* dengan nama yang sama dengan pengaturan sebelumnya.

3.2.2.3 Proses Menghapus Pengaturan

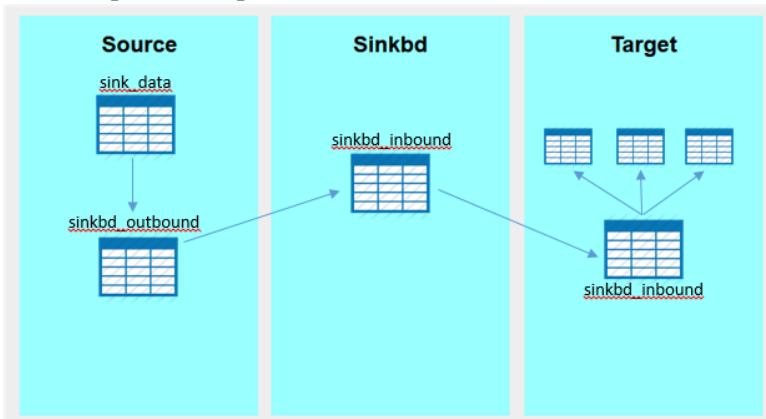
Proses menghapus pengaturan merupakan proses yang dilakukan ketika pengaturan tidak lagi dibutuhkan atau pun dibutuhkan pengaturan yang baru namun berbeda jauh dari

sebelumnya. Proses menghapus pengaturan dilakukan melalui antarmuka Sinkbd. Menghapus pengaturan berarti menghapus *file* xml yang sudah disimpan sebelumnya.

3.2.2.4 Proses Sinkronisasi Basis Data

Proses sinkronisasi basis data merupakan proses yang dilakukan ketika terjadi pengiriman data dari basis data *source* ke basis data *target*. Proses ini melalui beberapa tahapan pengiriman data karena kerangka kerja Sinkbd juga melakukan pencatatan data apa saja yang sedang dipertukarkan.

Gambaran pembagian *source*, Sinkbd, dan *target* secara umum dapat dilihat pada Gambar 3. 9.



Gambar 3. 9 Alur Proses Sinkronisasi Basis Data

Pada Gambar 3. 9, bagian *source* terdapat tabel *sink_data* dan tabel *sinkbd_outbound*. Bagian pusat Sinkbd hanya terdapat tabel *sinkbd_inbound*. Sedangkan bagian *target* terdapat tabel *sinkbd_inbound*. Secara umum, urutan proses sinkronisasi basis data adalah sebagai berikut.

1. Sinkbd mengambil informasi dari tabel *sink_bd* mengenai waktu terakhir sinkronisasi dilakukan pada pengaturan yang dipilih.
2. Sinkbd memilih data pada tabel *sink_data* pada bagian *source* berdasarkan data yang waktu perubahannya melebihi waktu terakhir sinkronisasi dilakukan.
3. Data-data tersebut disalin ke tabel *sinkbd_outbound* untuk siap dikirimkan.
4. Data dari *sinkbd_outbound* dimasukkan ke tabel *sinkbd_inbound* untuk dicatat. Kemudian divalidasi agar tidak ada kesalahan pengiriman data.
5. Data dari *sinkbd_inbound* ini diambil dan dikirimkan ke *sinkbd_inbound* pada basis data *target*. Kemudian dilakukan validasi kembali untuk memeriksa kesalahan pengiriman.
6. Apabila data sudah valid, maka data di *sinkbd_inbound* dimasukkan ke dalam tabel sesuai pengaturan yang sudah dibuat.
7. Waktu terakhir sinkronisasi pada tabel *sink_bd* diubah menjadi waktu sekarang.

3.2.2.5 Proses Melanjutkan Sinkronisasi Basis Data yang Terputus

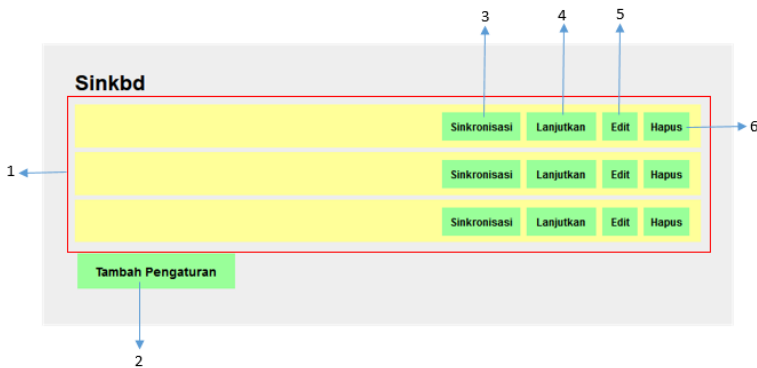
Proses melanjutkan sinkronisasi basis data yang terputus merupakan proses yang berlangsung ketika pengiriman data dari bagian Sinkbd menuju *target* dinyatakan tidak valid. Sinkbd menyimpan index terakhir data dinyatakan valid sehingga dari nilai tersebut Sinkbd dapat melanjutkan pengiriman data, bukan mengirimkan dari awal. Prosesnya sama dengan proses sinkronisasi basis data dari bagian Sinkbd menuju *target*.

3.2.3 Perancangan Antarmuka

Pada sub bab ini akan dibahas secara mendetail dari rancangan antarmuka kerangka kerja Sinkbd.

3.2.3.1 Antarmuka Halaman Menu Utama

Gambar 3. 10 merupakan gambar rancangan antarmuka halaman menu utama. Pada halaman tersebut menampilkan list pengaturan yang sudah dibuat beserta tombol aksi untuk mengelola pengaturan.



Gambar 3. 10 Rancangan Antarmuka Halaman Menu Utama

Gambar 3. 10 merupakan gambar rancangan antarmuka halaman menu utama. Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3. 10.

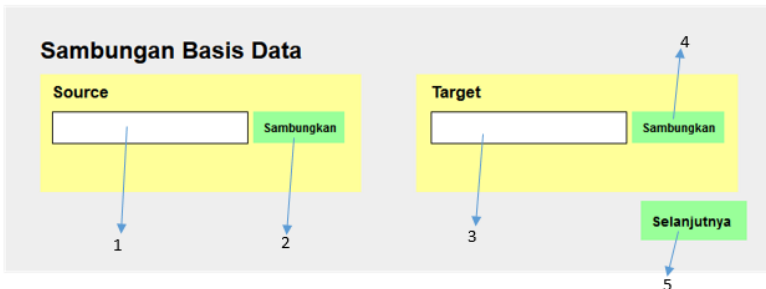
1. Berupa daftar pengaturan sinkronisasi yang sudah dibuat sebelumnya.
2. Berupa tombol menuju halaman tambah pengaturan sinkronisasi.
3. Berupa tombol untuk memulai sinkronisasi.
4. Berupa tombol untuk melanjutkan sinkronisasi yang terputus.
5. Berupa tombol menuju halaman mengubah pengaturan
6. Berupa tombol untuk menghapus pengaturan

3.2.3.2 Antarmuka Halaman Tambah Pengaturan

Halaman tambah pengaturan dibagi menjadi 3 sub halaman yaitu sub halaman pengaturan sambungan basis data, sub halaman pengaturan tabel dan kolom, dan sub halaman konfirmasi penyimpanan pengaturan.

3.2.3.2.1 Antarmuka Sub Halaman Pengaturan Sambungan Basis Data

Gambar 3. 11 merupakan gambar rancangan antarmuka sub halaman pengaturan sambungan basis data. Pada halaman tersebut menampilkan *form* yang harus diisi untuk melakukan sambungan basis data ke *source* dan *target*.



Gambar 3. 11 Rancangan Antarmuka Sub Halaman Pengaturan Sambungan Basis Data

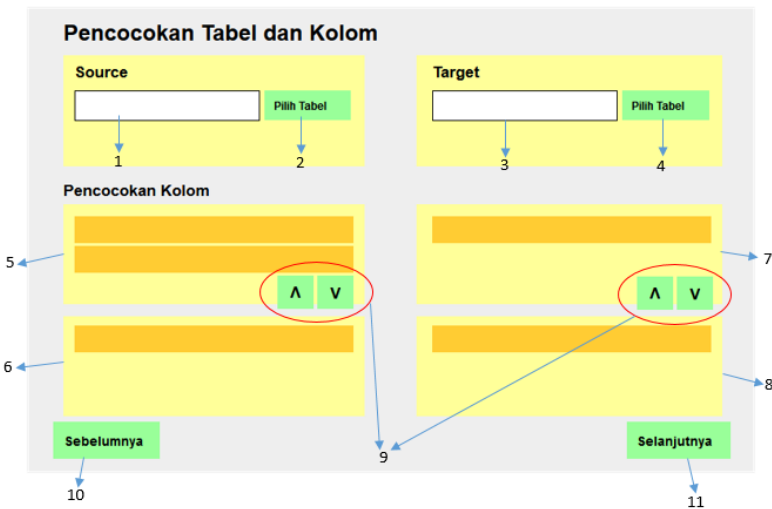
Gambar 3. 11 merupakan gambar rancangan antarmuka sub halaman pengaturan sambungan basis data. Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3. 11.

1. Berupa *textbox* yang digunakan untuk menampung alamat basis data *source*.
2. Berupa tombol untuk menyambungkan dengan basis data source.
3. Berupa *textbox* yang digunakan untuk menampung alamat basis data target.

4. Berupa tombol untuk menyambungkan dengan basis data target.
5. Berupa tombol untuk menuju sub halaman selanjutnya.

3.2.3.2.2 Antarmuka Sub Halaman Pengaturan Tabel dan Kolom

Gambar 3. 12 merupakan gambar rancangan antarmuka sub halaman pengaturan tabel dan kolom. Pada halaman tersebut menampilkan daftar tabel yang harus diisi untuk menampilkan kolom-kolom yang tersedia pada tabel tersebut. Kemudian kolom dicocokkan untuk dilakukan pengiriman data.



Gambar 3. 12 Rancangan Antarmuka Sub Halaman Pengaturan Tabel dan Kolom

Gambar 3. 12 merupakan gambar rancangan antarmuka sub halaman pengaturan tabel dan kolom. Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3. 12.

1. Berupa *combobox* yang digunakan untuk memilih tabel *source*.

2. Berupa tombol yang digunakan untuk memilih tabel *source* dan menampilkan kolom-kolom yang tersedia.
3. Berupa combobox yang digunakan untuk memilih tabel target.
4. Berupa tombol yang digunakan untuk memilih tabel target dan menampilkan kolom-kolom yang tersedia.
5. Berupa daftar kolom-kolom tidak dipilih yang dimiliki tabel *source*.
6. Berupa daftar kolom-kolom dipilih yang dimiliki tabel *source*.
7. Berupa daftar kolom-kolom tidak dipilih yang dimiliki tabel target.
8. Berupa daftar kolom-kolom dipilih yang dimiliki tabel target.
9. Berupa tombol untuk memindahkan daftar kolom tidak dipilih ke dalam daftar kolom dipilih.
10. Berupa tombol untuk kembali ke sub halaman sebelumnya.
11. Berupa tombol untuk menuju ke sub halaman selanjutnya.

3.2.3.2.3 Antarmuka Sub Halaman Konfirmasi Penyimpanan Pengaturan

Gambar 3. 13 merupakan gambar rancangan antarmuka sub halaman konfirmasi penyimpanan pengaturan. Pada halaman tersebut menampilkan pertanyaan pengaturan dua arah atau satu arah. Kemudian Administrator dapat membatalkan pembuatan pengaturan.



Gambar 3. 13 Rancangan Antarmuka Sub Halaman Konfirmasi Penyimpanan Pengaturan

Gambar 3. 13 merupakan gambar rancangan antarmuka sub halaman konfirmasi penyimpanan pengaturan. Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3. 13.

1. Berupa *radio button* untuk memilih pengaturan berjenis satu arah atau dua arah.
2. Berupa tombol menuju sub halaman sebelumnya.
3. Berupa tombol untuk membatalkan pembuatan pengaturan.
4. Berupa tombol untuk menyimpan pengaturan.

3.2.3.3 Antarmuka Halaman Mengubah Pengaturan

Antarmuka halaman mengubah pengaturan sama dengan antarmuka halaman tambah pengaturan. Adapun yang membedakan adalah pada halaman mengubah pengaturan data sudah terisi berdasarkan pengaturan yang sudah disimpan sebelumnya.

BAB IV IMPLEMENTASI

Pada bab ini dijelaskan mengenai implementasi dari perancangan perangkat lunak. Implementasi yang dijelaskan meliputi lingkungan pembangunan perangkat lunak, implementasi antarmuka pengguna, dan implementasi proses-proses yang terjadi pada masing-masing kasus penggunaan pada perangkat lunak. Implementasi sistem mengacu pada perancangan yang ditulis pada Bab 3. Namun, tidak menutup kemungkinan adanya perubahan-perubahan dari rancangan tersebut apabila memang diperlukan.

4.1 Lingkungan Implementasi

Dalam merancang perangkat lunak ini digunakan beberapa perangkat pendukung sebagai berikut.

4.1.1 Lingkungan Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan sistem perangkat lunak berupa komputer dengan spesifikasi PC Lenovo, Intel Core i3-3240 3,40 Ghz dan 4GB *memory*.

4.1.2 Lingkungan Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan untuk pengembangan sistem adalah sebagai berikut.

1. Microsoft Windows 8 sebagai sistem operasi.
2. Spring Tool Suite 3.6.2.RELEASE sebagai IDE untuk implementasi kerangka kerja.
3. Apache Tomcat versi 8.0 sebagai *web server* tempat kerangka kerja dijalankan.
4. PostgreSQL versi 9.4 sebagai basis data kerangka kerja.

5. pgAdmin versi 1.20.0 sebagai antarmuka pengolah basis data PostgreSQL.
6. Firefox versi 38.0.5 sebagai *web browser* antarmuka kerangka kerja Sinkbd.
7. PowerDesigner versi 15.0 untuk merancang basis data.
8. StarUML versi 5.0.2 untuk merancang diagram perencanaan perangkat lunak.

Di dalam implementasinya nanti, Sinkbd membutuhkan beberapa variabel tetap sebagai pengaturan server. Variabel ini diletakkan di dalam sebuah *file configuration.properties* sehingga kerangka kerja hanya tinggal membaca variabel yang ada di dalam *file* tersebut. Isi dari *configuration.properties* ditunjukkan pada Gambar 4. 1.

```
sinkbd_address=127.0.0.1/sinkbd
sinkbd_username=postgres
sinkbd_pass=admin
dir_sinkbd_conf=E:\\Important\\Tugas Akhir\\ta-sinkbd\\Java Project 2\\SinkBDSiakad\\configuration
dir_sinkbd_dump=E:\\Important\\Tugas Akhir\\ta-sinkbd\\Java Project 2\\SinkBDSiakad\\dumpFile
dir_sinkbd_pg_dump=C:\\Program Files\\PostgreSQL\\9.4\\bin\\pg_dump.exe
```

Gambar 4. 1 Isi *File configuration.properties*

Pada Gambar 4. 1 terdapat 7 variabel yang harus disesuaikan sebelum kerangka kerja dijalankan. Masing-masing variabel penjelasannya sebagai berikut.

1. sinkbd_address
Merupakan variabel alamat basis data server Sinkbd.
2. sinkbd_username
Merupakan variabel *username* untuk login ke basis data Sinkbd
3. sinkbd_password
Merupakan variabel *password* untuk login ke basis data Sinkbd

4. `dir_sinkbd_conf`
Merupakan variabel posisi *directory* penyimpanan pengaturan yang dibuat melalui antarmuka Sinkbd
5. `dir_sinkbd_dump`
Merupakan variabel posisi *directory* penyimpanan hasil *backup* basis data.
6. `dir_sinkbd_pg_dump`
Merupakan variabel posisi fitur aplikasi `pg_dump` milik PostgreSQL untuk melakukan *backup* basis data.

4.2 Implementasi Basis Data

Pada subbab ini akan dibahas implementasi dari rancangan basis data yang telah dibahas pada Bab 3. Terdapat dua implementasi pada subbab ini yakni implementasi struktur basis data kerangka kerja dan implementasi struktur basis data kebutuhan sinkronisasi.

4.2.1 Implementasi Struktur Basis Data Kerangka Kerja

Implementasi struktur basis data kerangka kerja merupakan implementasi sintaks yang digunakan untuk membangun tabel-tabel yang dibutuhkan oleh kerangka kerja `sink_bd` dalam pencatatan pengaturan sinkronisasi. Implementasi basis data menggunakan sintaks SQL.

4.2.1.1 Implementasi Pembuatan Skema

Tabel yang dibutuhkan oleh kerangka kerja Sinkbd diletakkan pada skema tersendiri dengan nama `sinkbd`. Oleh karena itu *query* pertama yang dibutuhkan adalah seperti yang ditunjukkan oleh Kode Sumber 4. 1.

```
CREATE SCHEMA IF NOT EXISTS sinkbd;
```

Kode Sumber 4. 1 Implementasi Pembuatan Skema

4.2.1.2 Implementasi Tabel `sink_bd`

Tabel `sink_bd` merupakan tabel yang digunakan untuk menyimpan status pengaturan sinkronisasi basis data. Implementasi tabel `sink_bd` ditunjukkan pada Kode Sumber 4. 2.

```
CREATE TABLE sinkbd.sink_bd (
  id_sink_bd serial PRIMARY KEY,
  conf_name varchar(50),
  last_sink timestamp DEFAULT current_timestamp,
  dump_files varchar(100),
  last_valid varchar(50)
);
```

Kode Sumber 4. 2 Implementasi Struktur Tabel `sink_bd`

4.2.2 Implementasi Struktur Basis Data Kebutuhan Sinkronisasi

Implementasi struktur basis data kebutuhan sinkronisasi merupakan implementasi sintaks yang digunakan untuk membangun tabel-tabel yang dibutuhkan oleh kerangka kerja `sink_bd` dalam melakukan proses pengiriman data perubahan. Implementasi basis data menggunakan sintaks SQL dan dijalankan bersamaan dengan ketika pengaturan baru dibuat.

4.2.2.1 Implementasi Pembuatan Skema

Skema `sinkbd` dibuat pada *source* dan *target* agar proses tidak mengganggu kegiatan yang dilakukan basis data. Implementasi pembuatan skema ditunjukkan pada Kode Sumber 4. 3.

```
CREATE SCHEMA IF NOT EXISTS sinkbd;
```

Kode Sumber 4. 3 Implementasi Pembuatan Skema Pada *Source* dan *Target*

4.2.2.2 Implementasi Tabel `sink_data`

Tabel `sink_data` merupakan tabel yang digunakan untuk menyimpan perubahan data oleh *trigger* yang sudah dibuat pada tabel tertentu. Implementasi tabel `sink_data` ditunjukkan pada Kode Sumber 4. 4.

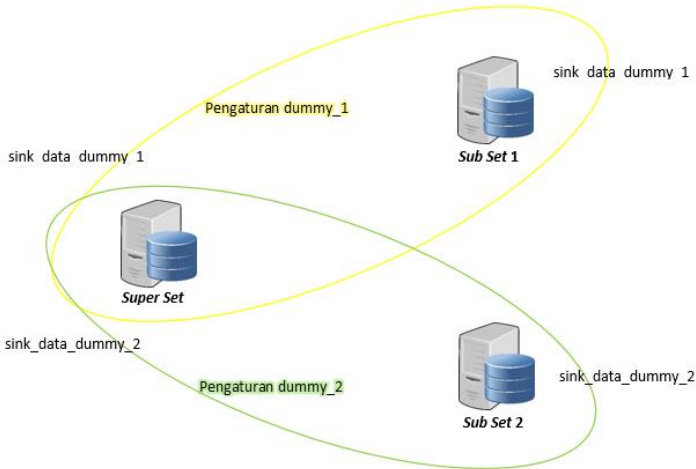
```

CREATE TABLE IF NOT EXISTS sinkbd.sink_data_+"confName_"
(
  id_sink_data SERIAL PRIMARY KEY,
  schema_name varchar(50) NOT NULL,
  table_name varchar(50) NOT NULL,
  event varchar(1) NOT NULL,
  row_data text,
  old_data text,
  pk varchar(50),
  pk_data varchar(50),
  time timestamp DEFAULT current_timestamp
);

```

Kode Sumber 4. 4 Implementasi Struktur Tabel sink_data

Pada Kode Sumber 4. 4, nama tabel sink_data diberikan akhiran nama pengaturan. Hal ini dimaksudkan untuk mengantisipasi pengaturan tambahan seperti pada Gambar 4. 2. Gambar 4. 2 merupakan gambaran pengaturan yang saling berhubungan. Ketika sinkronisasi dummy_1 dijalankan, *Sub Set* 1 akan mengirimkan perubahan data menuju *Super Set*, maka *trigger* pada pengaturan dummy_1 harus dimatikan untuk sementara tetapi *trigger* pengaturan_2 tetap berjalan untuk mencatat perubahan data. Sehingga pengaturan_2 tetap dapat mengirim perubahan data dari *Super Set* menuju *Sub Set* 2 begitu pula sebaliknya. Penambahan akhiran nama pengaturan ini juga diimplementasikan pada pembuatan *trigger* seperti pada Kode Sumber 4. 8.



Gambar 4. 2 Pengaturan Super Set dan Sub Set

4.2.2.3 Implementasi Tabel sinkbd_inbound

Tabel sinkbd_inbound merupakan tabel yang digunakan untuk menyimpan pengiriman data masuk ketika sinkronisasi dilakukan. Implementasi tabel sinkbd_inbound ditunjukkan pada Kode Sumber 4. 5.

```
CREATE TABLE IF NOT EXISTS sinkbd.sinkbd_inbound (
  id_sink_data SERIAL PRIMARY KEY,
  schema_name varchar(50) NOT NULL,
  table_name varchar(50) NOT NULL,
  event varchar(1) NOT NULL,
  row_data text,
  old_data text,
  pk varchar(50),
  pk_data varchar(50),
  time varchar(50)
);
```

Kode Sumber 4. 5 Implementasi Struktur Tabel sinkbd_inbound

4.2.2.4 Implementasi Tabel sinkbd_outbound

Tabel sinkbd_outbound merupakan tabel yang digunakan untuk menyimpan pengiriman data keluar ketika sinkronisasi

dilakukan. Implementasi tabel `sinkbd_inbound` ditunjukkan pada Kode Sumber 4. 6.

```
CREATE TABLE IF NOT EXISTS sinkbd.sinkbd_outbound (
  id_sink_data SERIAL PRIMARY KEY,
  schema_name varchar(50) NOT NULL,
  table_name varchar(50) NOT NULL,
  event varchar(1) NOT NULL,
  row_data text,
  old_data text,
  pk varchar(50),
  pk_data varchar(50),
  time varchar(50)
);
```

Kode Sumber 4. 6 Implementasi Struktur Tabel `sinkbd_outbound`

4.2.2.5 Implementasi Fungsi `get_pk`

Fungsi `get_pk` merupakan fungsi yang digunakan untuk mendapatkan *primary key* pada tabel yang diinginkan. Fungsi ini dibutuhkan oleh fungsi *trigger*. Implementasi fungsi `get_pk` ditunjukkan pada Kode Sumber 4. 7.

```
CREATE OR REPLACE FUNCTION get_pk(table_name varchar(50))
RETURNS text AS $$
  DECLARE
    result text;
  BEGIN
    SELECT a.attname
    FROM   pg_index i
    JOIN   pg_attribute a ON a.attrelid = i.indrelid
    AND a.attnum = ANY(i.indkey)
    WHERE  i.indrelid = table_name::regclass
    AND    i.indisprimary INTO result;
    RETURN result;
  END;
$$ LANGUAGE plpgsql;
```

Kode Sumber 4. 7 Implementasi Fungsi `get_pk`

4.2.2.6 Implementasi Fungsi `process_sink_data`

Fungsi `process_sink_data` merupakan fungsi yang berjalan ketika *trigger* perubahan data pada tabel yang sudah diatur berjalan. Fungsi ini bermaksud memasukkan segala perubahan data tabel

ke dalam tabel sink_data. Implementasi fungsi get_pk ditunjukkan pada Kode Sumber 4. 8.

```

CREATE OR REPLACE FUNCTION
process_sink_data_+"confName"+() RETURNS TRIGGER AS
$sink_data$
  DECLARE
    schema_name text;
    new_row text;
    old_row text;
    primary_key text;
    pk_data text;
  BEGIN
    SELECT get_pk(concat('\', (CAST (TG_TABLE_NAME AS
text)), '\')) INTO primary_key;
    IF (TG_OP = 'UPDATE') THEN
      new_row = NEW.*::text;
      pk_data = substring(split_part(new_row, ',',1) from
2);
      old_row = OLD.*::text;
      INSERT INTO sinkbd.sink_data_+"confName"+
(schema_name, table_name, event, row_data, old_data, pk,
pk_data) values (TG_TABLE_SCHEMA, TG_TABLE_NAME, 'U',
new_row, old_row, primary_key, pk_data);
      RETURN NEW;
    ELSIF (TG_OP = 'INSERT') THEN
      new_row = NEW.*::text;
      pk_data = substring(split_part(new_row, ',',1) from
2);
      INSERT INTO sinkbd.sink_data_+"confName"+
(schema_name, table_name, event, row_data, old_data, pk,
pk_data) values (TG_TABLE_SCHEMA, TG_TABLE_NAME, 'I',
new_row, '', primary_key, pk_data);
      RETURN NEW;
    ELSIF (TG_OP = 'DELETE') THEN
      old_row = OLD.*::text;
      pk_data = substring(split_part(old_row, ',',1) from
2);
      INSERT INTO sinkbd.sink_data_+"confName"+
(schema_name, table_name, event, row_data, old_data, pk,
pk_data) values (TG_TABLE_SCHEMA, TG_TABLE_NAME, 'D', '',
old_row, primary_key, pk_data);
      RETURN OLD;
    END IF;
    RETURN NULL;
  END;
$sink_data$ LANGUAGE plpgsql;

```

Kode Sumber 4. 8 Implementasi Fungsi process_sink_data

4.3 Implementasi Proses

Pada subbab ini akan dibahas tentang fungsi atau proses yang berjalan pada kerangka kerja Sinkbd.

4.3.1 Proses Menambah Pengaturan

Proses ini merupakan proses yang dieksekusi ketika Administrator melakukan penambahan pengaturan melalui halaman antarmuka. Kode Sumber 4. 9 merupakan implementasi dari proses menambah pengaturan baru.

Kode Sumber 4. 9 menjelaskan bahwa nama pengaturan disimpan ke dalam tabel `sink_bd`. Kolom `last_sink` dimasukkan nilai "0001-01-01 00:00:00.000" sebagai tanda pengaturan belum pernah dilakukan sinkronisasi. Kolom `last_valid` dimasukkan nilai 0. Tahap selanjutnya menuliskan pengaturan ke dalam *file* xml menggunakan fungsi pada kelas `XmlService.java`. Tahap terakhir *program* membuat pengaturan *trigger* menggunakan dungsi pada kelas `SqlService.java`.

```
@RequestMapping(value="/newConf", method=RequestMethod.POST)
public String saveConf(@ModelAttribute("configuration")
Configuration conf){
    String sql = "INSERT INTO sinkbd.sink_bd (conf_name,
last_sink, last_valid) VALUES ('"+conf.getConfName()+"', '0001-
01-01 00:00:00.000', '0')";

    sqlService.dbConnectInsert(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);
    xmlService.writeXML(conf);
    sqlService.createTrigger(conf);
    return "redirect:/";
}
```

Kode Sumber 4. 9 Implementasi Proses Menambah Pengaturan

4.3.2 Proses Mengubah Pengaturan

Proses ini merupakan proses yang dieksekusi ketika Administrator mengubah pengaturan melalui halaman antarmuka. Kode Sumber 4. 10 merupakan implementasi dari proses

mengubah pengaturan. Kode ini hampir sama dengan fungsi simpan pengaturan baru, namun *program* tidak membuat perubahan pada tabel `sink_bd`.

```
@RequestMapping(value="/editConf/{name}",
method=RequestMethod.POST)
public String updateConf(@ModelAttribute("configuration")
Configuration conf){
    xmlService.writeXML(conf);
    sqlService.createTrigger(conf);
    return "redirect:/";
}
```

Kode Sumber 4. 10 Implementasi Proses Mengubah Pengaturan

4.3.3 Proses Menghapus Pengaturan

Proses ini merupakan proses yang dieksekusi ketika Administrator menghapus pengaturan melalui halaman menu utama. Kode Sumber 4. 11 merupakan implementasi dari proses menghapus pengaturan.

Kode Sumber 4. 11 menjelaskan bahwa *program* menghapus pengaturan di dalam tabel `sink_bd`. Kemudian isi tabel `sink_data` pada basis data *source* dihapus sesuai nama tabel di dalam pengaturan. Tahap selanjutnya menghapus *file* xml berdasarkan nama pengaturan.

```
@RequestMapping(value="/deleteConf", method=RequestMethod.GET)
public @ResponseBody String deleteConf(@RequestParam String
confName) throws JsonGenerationException, JsonMappingException,
IOException{
    String sql = "DELETE FROM sinkbd.sink_bd WHERE conf_name =
"+"+confName+"";

    sqlService.dbConnectInsert(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);
    Configuration conf = xmlService.readOneXml(confName);
    sqlService.truncateSinkData(conf);
    xmlService.deleteXml(confName);
    return mapper.writeValueAsString("success");
}
```


Kode Sumber 4. 11 Implementasi Proses Menghapus Pengaturan

4.3.4 Proses Sinkronisasi Basis Data

Proses ini merupakan proses yang dieksekusi ketika Administrator menekan tombol sinkronisasi melalui halaman menu utama. Proses sinkronisasi basis data dibagi menjadi beberapa bagian.

1. Sinkbd mengambil informasi dari tabel sink_bd mengenai waktu terakhir sinkronisasi dilakukan. Kode Sumber 4. 12 merupakan sebagian dari implementasi tahap ini.

```
String sql = "SELECT last_sink, last_valid FROM sinkbd.sink_bd
WHERE conf_name = '"+confName+"'";

queryResult =
sqlService.dbConnectSelect(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);

last_sink = sqlService.queryToList(queryResult,
"last_sink").get(0);

last_valid = sqlService.queryToList(queryResult,
"last_valid").get(0);
```

Kode Sumber 4. 12 Implementasi Mendapatkan Waktu Terakhir Sinkronisasi

2. Sinkbd memilih data pada tabel sink_data pada bagian *source* berdasarkan data yang waktu perubahannya melebihi waktu terakhir sinkronisasi dilakukan. Kode Sumber 4. 13 merupakan sebagian dari implementasi tahap ini.

```
sql = "TRUNCATE TABLE sinkbd.sinkbd_outbound RESTART
IDENTITY;INSERT INTO sinkbd.sinkbd_outbound SELECT * FROM
sinkbd.sink_data WHERE time > '"+last_sink+"' AND table_name IN
('"+tables_sink+"');";

sqlService.dbConnectInsert(conf.getSourceAddress(),
conf.getSourceUsername(), conf.getSourcePassword(), sql);
```

Kode Sumber 4. 13 Implementasi Mendapatkan Data Sinkronisasi

3. Data-data tersebut disalin ke tabel `sinkbd_outbound` untuk siap dikirimkan. Tahap ini juga diimplementasikan pada Kode Sumber 4. 13.
4. Data dari `sinkbd_outbound` dimasukkan ke tabel `sinkbd_inbound` untuk dicatat. Kemudian divalidasi agar tidak ada kesalahan pengiriman data. Kode Sumber 4. 14 merupakan sebagian dari implementasi tahap ini.

```
sqlService.dbSourceToSinkbd(conf, 0);
int valid = sqlService.sinkbdInboundCheck(conf);
```

Kode Sumber 4. 14 Implementasi Pengiriman Data *Source* ke Pusat

5. Data dari `sinkbd_inbound` ini diambil dan dikirimkan ke `sinkbd_inbound` pada basis data target. Kemudian dilakukan validasi kembali untuk memeriksa kesalahan pengiriman. Kode Sumber 4. 15 merupakan sebagian dari implementasi tahap ini.

```
sqlService.dbSinkbdToTarget(conf, 0);
valid = sqlService.targetInboundCheck(conf);
```

Kode Sumber 4. 15 Implementasi Pengiriman Data Pusat ke *Target*

6. Apabila data sudah valid, maka data di `sinkbd_inbound` dimasukkan ke dalam tabel sesuai pengaturan yang sudah dibuat. Seperti potongan kode pada Kode Sumber 4. 16.

```
if(valid==0){
    LOGGER.info("VALID!");
    String dumpFiles = "";

    boolean unchanged = false;
    sql = "SELECT COUNT(*) AS jumlah FROM sinkbd.sinkbd_inbound";
    queryResult =
    sqlService.dbConnectSelect(configurationService.getAddress(),
    configurationService.getUserName(),
    configurationService.getPassword(), sql);

    LOGGER.info(queryResult.get(0).get("jumlah"));
```

```

    if(queryResult.get(0).get("jumlah").compareToIgnoreCase("0")
    == 0){
        unchanged = true;
    }

    ...
    ...
    ...

    if(unchanged){
        sql = "UPDATE sinkbd.sink_bd SET last_sink=now() WHERE
        conf_name = '"+confName+"'";

        sqlService.dbConnectInsert(configurationService.getAddress(),
        configurationService.getUserName(),
        configurationService.getPassword(), sql);
        return mapper.writeValueAsString("Unchanged");
    }

    dumpFiles = sqlService.insertValidData(conf);
    sql = "UPDATE sinkbd.sink_bd SET last_sink=now(),
    dump_files='"+dumpFiles+"', last_valid='"+valid+"' WHERE
    conf_name = '"+confName+"'";
    sqlService.dbConnectInsert(configurationService.getAddress(),
    configurationService.getUserName(),
    configurationService.getPassword(), sql);

    return mapper.writeValueAsString("Success");
}

```

Kode Sumber 4. 16 Implementasi Memasukkan Data Valid

7. Waktu terakhir sinkronisasi pada tabel sink_bd diubah menjadi waktu sekarang.

4.3.5 Proses Melanjutkan Sinkronisasi Basis Data yang Terputus

Proses melanjutkan sinkronisasi basis data yang terputus merupakan proses yang berlangsung ketika pengiriman data dari bagian Sinkbd menuju *target* dinyatakan tidak valid. Sinkbd menyimpan index terakhir data dinyatakan valid pada tabel sink_bd di kolom last_valid sehingga dari nilai tersebut Sinkbd dapat melanjutkan pengiriman data, bukan mengirimkan dari awal. Prosesnya sama dengan proses sinkronisasi basis data dari

bagian Sinkbd menuju *target*. Potongan implementasinya terdapat pada Kode Sumber 4. 17.

```

if(last_valid == "0"){
    sqlService.dbSinkbdToTarget(conf, 0);
}
else{
    LOGGER.info("Last valid :"+ last_valid);
    if(last_valid.charAt(0) == '~'){
        rev = true;
        conf = reverse;
        last_valid = last_valid.substring(1);
    }
    sqlService.dbSinkbdToTarget(conf,
Integer.parseInt(last_valid)+1);
}

int valid = sqlService.targetInboundCheck(conf);

if(valid==0){
    LOGGER.info("VALID!");
    String dumpFiles = sqlService.insertValidData(conf);
    sql = "UPDATE sinkbd.sink_bd SET last_sink='"+last_sink+',
dump_files="'+dumpFiles+', last_valid="'+valid+' WHERE
conf_name = '"+confName+'";

    sqlService.dbConnectInsert(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);

    return mapper.writeValueAsString("Success");
}
else{
    LOGGER.error("NOT VALID!");
    if(rev){
        sql = "UPDATE sinkbd.sink_bd SET last_valid=~"+valid+'
WHERE conf_name = '"+confName+'";
    }
    else{
        sql = "UPDATE sinkbd.sink_bd SET last_valid='"+valid+'
WHERE conf_name = '"+confName+'";
    }
    sqlService.dbConnectInsert(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);
}

```

```
    return mapper.writeValueAsString(valid);  
}
```

Kode Sumber 4. 17 Implementasi Melanjutkan Sinkronisasi Basis Data yang Terputus

4.4 Implementasi Antarmuka

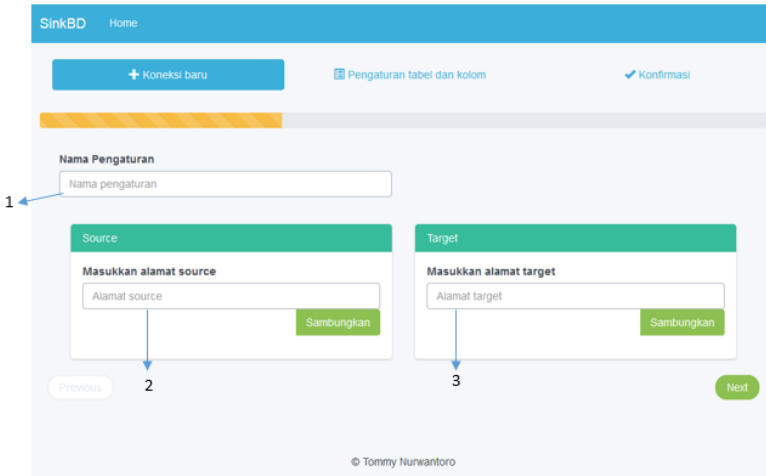
Pada subbab ini akan dijelaskan implementasi antarmuka berdasarkan rancangan antarmuka yang telah dibahas pada bab 3. Pada bagian implementasi antarmuka ini juga akan dijelaskan mengenai fungsi-fungsi yang digunakan dalam program tugas akhir ini dan disertai dengan kode sumber masing-masing fungsi utama.

4.4.1 Antarmuka Halaman Tambah Pengaturan

Halaman tambah pengaturan merupakan halaman yang diakses Administrator melalui halaman menu utama untuk menambahkan pengaturan baru. Halaman tambah pengaturan dibagi menjadi tiga tahapan. Pengaturan sambungan basis data, pengaturan tabel dan kolom, dan konfirmasi penyimpanan pengaturan.

4.4.1.1 Antarmuka Sub Halaman Pengaturan Basis Data

Antarmuka sub halaman pengaturan basis data ditunjukkan pada Gambar 4. 3. Pada sub halaman ini, Administrator diminta memasukkan nama pengaturan, alamat basis data *source*, dan alamat basis data *target*.



Gambar 4. 3 Antarmuka Sub Halaman Sambungan Basis Data

Gambar 4. 3 Merupakan antarmuka sub halaman pengaturan basis data. Berikut penjelasan masing-masing nomor yang tertera pada Gambar 4. 3.

1. Merupakan tempat memasukkan nama pengaturan.
2. Merupakan tempat memasukkan alamat basis data *source*.
3. Merupakan tempat memasukkan alamat basis data *target*.

Nama pengaturan yang dimasukkan tidak boleh sama dengan pengaturan yang sudah disimpan sebelumnya. Jika nama pengaturan sama, maka akan muncul peringatan seperti yang ditunjukkan lingkaran merah pada Gambar 4. 4.

Gambar 4. 4 Validasi Form Nama Pengaturan

Kode Sumber 4. 18 merupakan implementasi dari fungsi validasi nama pengaturan. Kode Sumber 4. 18 diimplementasikan pada kelas *service* yaitu *XmlService.java*. Pada kode ini menjelaskan bahwa program mencari nama *file* di dalam *folder* yang tempatnya didapat dari kelas *service* yaitu *ConfigurationService.java*.

```

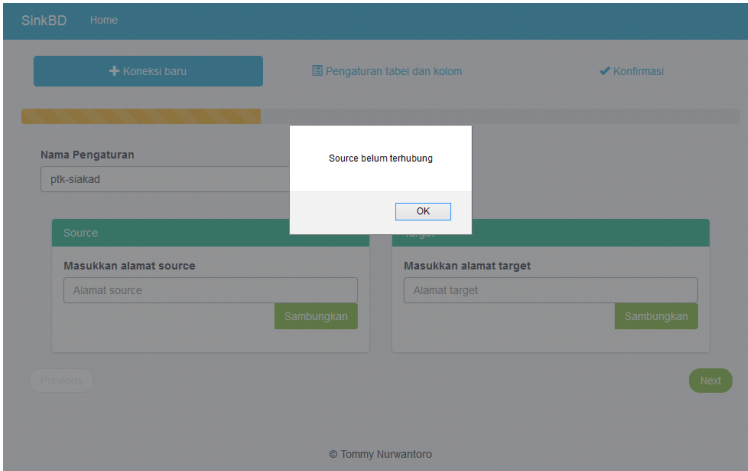
public boolean fileNameExist(String fileName){
    boolean exist = true;
    File folder = new File(configurationService.getDirConf());
    File[] files = folder.listFiles();
    for(int i=0; i<files.length; i++){
        if(files[i].getName().equalsIgnoreCase(fileName+".xml"))
            exist = false;
    }
    return exist;
}

```

Kode Sumber 4. 18 Kode Untuk Memeriksa Nama Pengaturan

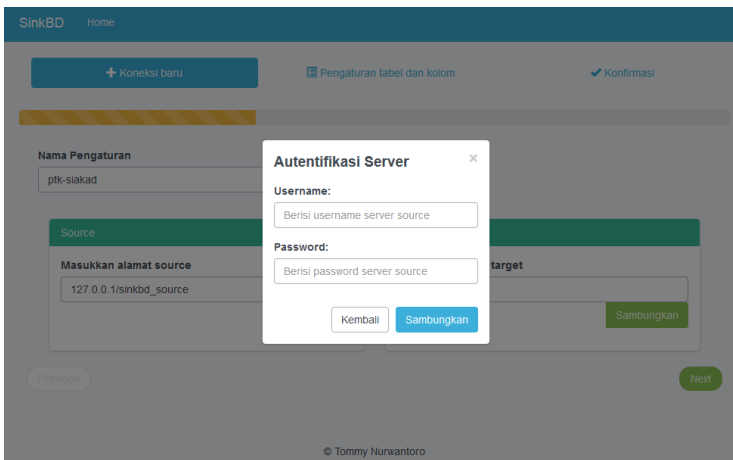
Administrator juga harus memasukkan alamat basis data *source* dan alamat basis data *target*. Jika *textbox* tersebut dikosongkan maka pengaturan tidak dapat dilanjutkan ke halaman

selanjutnya dan akan muncul peringatan seperti ditunjukkan pada Gambar 4. 5.



Gambar 4. 5 Validasi *Form* Alamat Basis Data

Administrator juga diminta untuk memasukkan *username* dan *password* sambungan ke basis data seperti ditunjukkan pada Gambar 4. 6.



Gambar 4. 6 Memasukkan *Username* dan *Password* Server

Kode Sumber 4. 19 merupakan implementasi dari fungsi menghubungkan Sinkbd ke basis data untuk mendapatkan tabel yang tersedia. Kode Sumber 4. 19 diimplementasikan pada kelas *controller* yaitu *FormController.java*. Pada kode ini menjelaskan bahwa *program* memilih nama tabel dan dimasukkan ke dalam sebuah *list*.

```

@RequestMapping(value="/databaseConnect",
method=RequestMethod.GET)
public @ResponseBody String databaseConnect(@RequestParam String
address, @RequestParam String user, @RequestParam String
password, Configuration conf) throws JsonGenerationException,
JsonMappingException, IOException{
    List<Map<String, String>> queryResult = new
ArrayList<Map<String,String>>();
    List<String> tableList = new ArrayList<String>();
    ObjectMapper mapper = new ObjectMapper();

    String sql = "SELECT table_name FROM
information_schema.tables WHERE table_schema='public' AND
table_type='BASE TABLE' ORDER BY table_name ASC";

    queryResult = sqlService.dbConnectSelect(address, user,
password, sql);

    if(queryResult == null){
        LOGGER.error("NULL");
        return "";
    }
    else{
        tableList = sqlService.queryToList(queryResult,
"table_name");
        conf.setTargetTableList(tableList);
        LOGGER.info(tableList.toString());
        return mapper.writeValueAsString(tableList);
    }
}

```

Kode Sumber 4. 19 Kode Untuk Mendapatkan Seluruh Nama Tabel

4.4.1.2 Antarmuka Sub Halaman Pengaturan Tabel dan Kolom

Antarmuka sub halaman pengaturan basis data ditunjukkan pada Gambar 4. 7. Pada sub halaman ini, Administrator diminta memilih tabel dan mencocokkan kolom untuk disinkronisasikan.



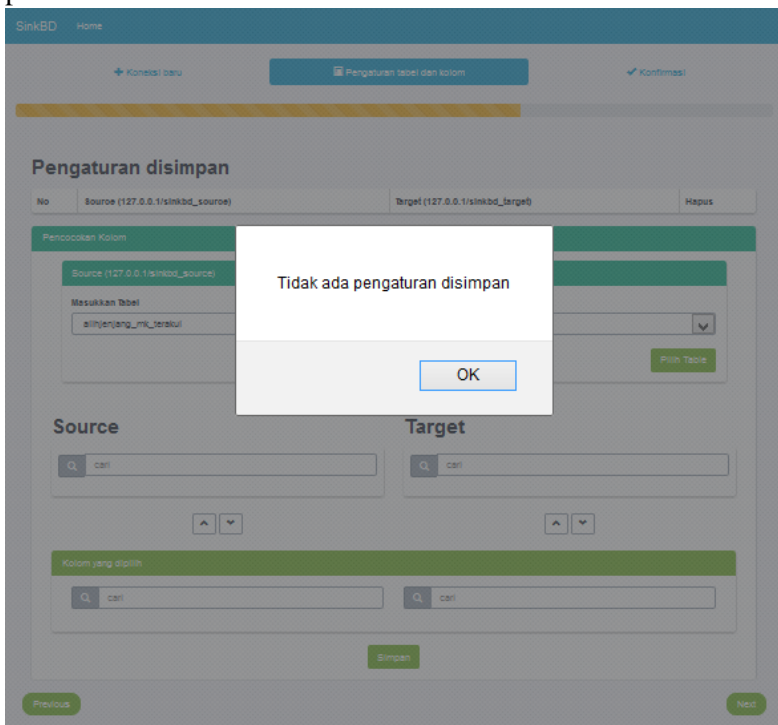
Gambar 4. 7 Antarmuka Sub Halaman Pengaturan Tabel dan Kolom

Gambar 4. 7 Merupakan antarmuka sub halaman pengaturan basis data. Berikut penjelasan masing-masing nomor yang tertera pada Gambar 4. 7.

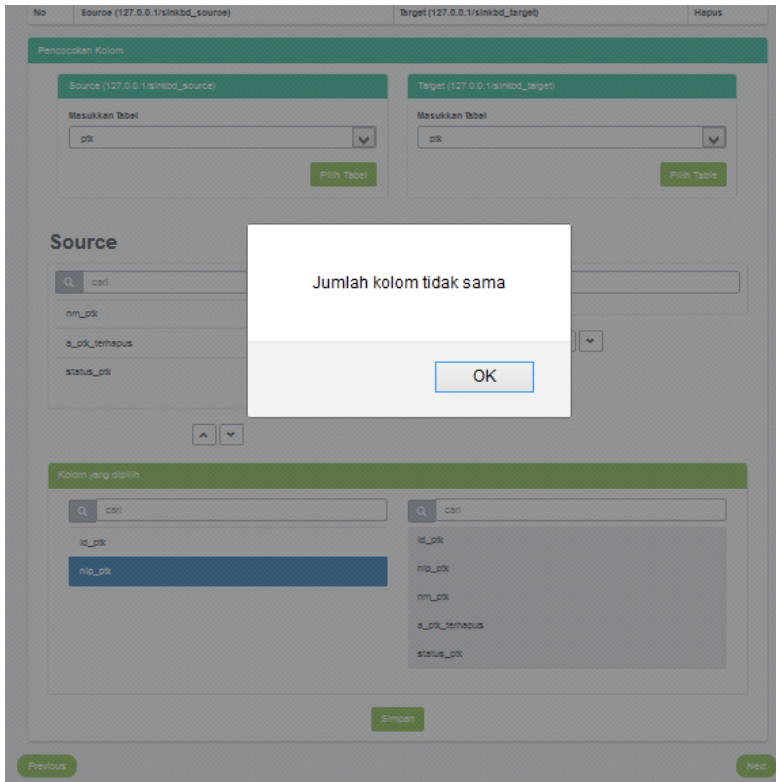
1. Merupakan tempat memilih tabel *source*.
2. Merupakan tempat memilih tabel *target*.
3. Merupakan tempat mencocokkan kolom *source* dan *target*.
4. Merupakan tempat menyimpan pengaturan tabel sementara.

5. Merupakan tombol untuk menyimpan pengaturan kolom.

Apabila tidak ada pengaturan tabel dan kolom yang disimpan, maka sub halaman pengaturan selanjutnya tidak dapat dibuka dan akan muncul pesan validasi *form* seperti pada Gambar 4. 8. Apabila jumlah kolom yang dipilih pada bagian *source* dan *target* tidak sama, maka akan muncul pesan validasi *form* seperti pada Gambar 4. 9.



Gambar 4. 8 Validasi *Form* Tidak Ada Pengaturan Disimpan



Gambar 4. 9 Validasi Form Jumlah Kolom Dipilih Tidak Sama

Kode Sumber 4. 20 merupakan implementasi fungsi mendapatkan seluruh kolom yang tersedia. Kode Sumber 4. 20 diimplementasikan pada kelas *controller* yaitu *FormController.java*. Kode ini menjelaskan bahwa *program* memilih seluruh nama kolom. Selain itu, kolom juga dibedakan menurut batasan boleh berisi null atau tidak. Hal ini untuk mengantisipasi kesalahan memasukkan data ketika melakukan sinkronisasi data. Nama-nama kolom ini dikembalikan dalam bentuk *list*.

```

@RequestMapping(value="/tableConnect", method=RequestMethod.GET)
public @ResponseBody String tableConnect(@RequestParam String
address, @RequestParam String table, @RequestParam String user,
@RequestParam String password) throws JsonGenerationException,
JsonMappingException, IOException{
    List<Map<String, String>> queryResult = new
ArrayList<Map<String,String>>();
    List<String> columnList = new ArrayList<String>();
    List<String> columnListNotNull = new ArrayList<String>();
    ObjectMapper mapper = new ObjectMapper();

    String sql = "SELECT column_name FROM information_schema.columns
WHERE table_schema = 'public' AND table_name = '"+table+"'";
    queryResult = sqlService.dbConnectSelect(address, user,
password, sql);

    if(queryResult == null){
        return "";
    }
    else{
        columnList = sqlService.queryToList(queryResult,
"column_name");
    }

    sql = "select column_name from INFORMATION_SCHEMA.COLUMNS where
is_nullable = 'NO' AND table_name = '"+table+"'";
    queryResult = sqlService.dbConnectSelect(address, user,
password, sql);

    if(queryResult == null){
        return mapper.writeValueAsString(columnList);
    }
    else{
        columnListNotNull = sqlService.queryToList(queryResult,
"column_name");
        for(int i=0; i<columnList.size(); i++){
            for(int j=0; j<columnListNotNull.size(); j++){
                if(columnList.get(i).compareToIgnoreCase(columnListNotNull.get(j))
== 0){
                    String temp = "~" + columnList.get(i);
                    columnList.set(i, temp);
                }
            }
        }
    }
    return mapper.writeValueAsString(columnList);
}

```

Kode Sumber 4. 20 Kode Untuk Mendapatkan Seluruh Nama Kolom

Pengaturan tabel yang disimpan akan muncul di bagian “Pengaturan Disimpan.” Gambar 4. 10 merupakan gambaran setelah tabel dan kolom dicocokkan dan menekan tombol simpan. Pengaturan yang dibuat bisa terdiri dari beberapa tabel.

Pengaturan disimpan

No	Source (127.0.0.1/simkdb_source)	Target (127.0.0.1/simkdb_target)	Hapus
1	ptk	ptk	✖
2	pd	pd	✖

Pencocokan Kolom

Source (127.0.0.1/simkdb_source)

Masukkan Tabel

pd

Pilih Tabel

Target (127.0.0.1/simkdb_target)

Masukkan Tabel

pd

Pilih Tabel

Source

id_ptk

Target

id_ptk

Kolom yang dipilih

id_pd

nim_pd

nim_pd

s_pd_terhapus

angkatan_pd

id_pd

nim_pd

nim_pd

s_pd_terhapus

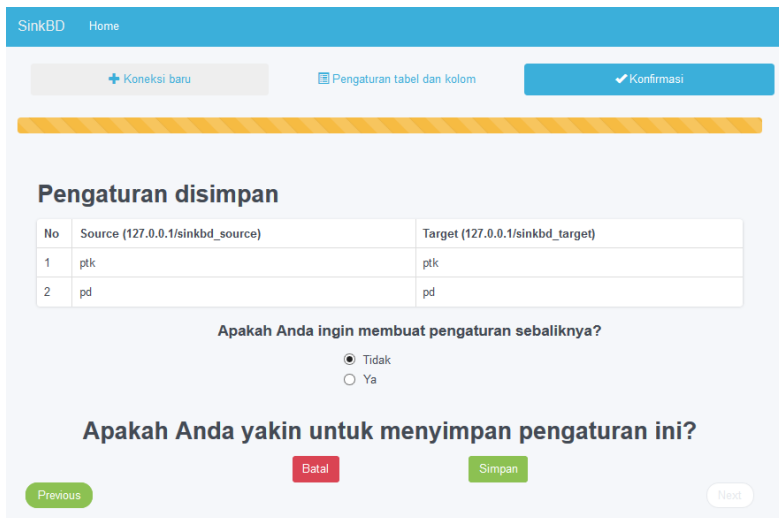
angkatan_pd

Previous
Next

Gambar 4. 10 Tampilan Setelah Pencocokan Tabel dan Kolom

4.4.1.3 Antarmuka Sub Halaman Pengaturan Penyimpanan Pengaturan

Antarmuka sub halaman pengaturan penyimpanan pengaturan ditunjukkan pada Gambar 4. 11. Pada sub halaman ini, Administrator diminta memilih jenis pengaturan satu arah atau dua arah melalui *radio button* yang telah disediakan. Pengaturan satu arah adalah jenis pengiriman data dari *source* ke *target* saja, sedangkan pengaturan dua arah adalah jenis pengiriman data dari *source* ke *target* kemudian sebaliknya dari *target* ke *source*. Jika Administrator menekan tombol simpan, maka pengaturan akan disimpan. Sebaliknya jika menekan tombol batal, maka pengaturan tidak disimpan.



The screenshot shows a web interface for SinkBD. At the top, there is a navigation bar with 'SinkBD' and 'Home'. Below it, there are three buttons: '+ Koneksi baru', 'Pengaturan tabel dan kolom', and 'Konfirmasi'. The main content area is titled 'Pengaturan disimpan' and contains a table with the following data:

No	Source (127.0.0.1/sinkbd_source)	Target (127.0.0.1/sinkbd_target)
1	ptk	ptk
2	pd	pd

Below the table, there is a question: 'Apakah Anda ingin membuat pengaturan sebaliknya?' with two radio buttons: 'Tidak' (selected) and 'Ya'. At the bottom, there is another question: 'Apakah Anda yakin untuk menyimpan pengaturan ini?' with three buttons: 'Previous', 'Batal', and 'Simpan'. There is also a 'Next' button in the bottom right corner.

Gambar 4. 11 Antarmuka Sub Halaman Konfirmasi Penyimpanan Pengaturan

Kode Sumber 4. 21 merupakan implementasi dari fungsi penyimpanan pengaturan. Kode Sumber 4. 21 diimplementasikan pada kelas *controller* yaitu kelas *FormController.java*.

```

@RequestMapping(value="/newConf", method=RequestMethod.POST)
public String saveConf(@ModelAttribute("configuration")
Configuration conf){
    String sql = "INSERT INTO sinkbd.sink_bd (conf_name, last_sink,
last_valid) VALUES ('"+conf.getConfName()+"', '0001-01-01
00:00:00.000', '0')";

    sqlService.dbConnectInsert(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);
xmlService.writeXML(conf);
sqlService.createTrigger(conf);
return "redirect:/";
}

```

Kode Sumber 4. 21 Kode Untuk Menyimpan Pengaturan

4.4.2 Antarmuka Halaman Menu Utama

Antarmuka halaman menu utama merupakan tempat Administrator melihat pengaturan apa saja yang sudah dibuat. Halaman menu utama Sinkbd ditunjukkan pada Gambar 4. 12. Pada Gambar 4. 12 Administrator dapat menjalankan sinkronisasi dan menghapus pengaturan dengan menekan tombol yang disediakan.

No	Nama Pengaturan	Alamat Source	Alamat Target	Sinkronisasi	Lanjutkan	Edit	Hapus
1	ptk-pd-siakad	127.0.0.1/sinkbd_source	127.0.0.1/sinkbd_target				

+ Tambah pengaturan

© Tommy Nurwantoro

Gambar 4. 12 Antarmuka Halaman Menu Utama

Kode Sumber 4. 22 merupakan implementasi dari fungsi hapus pengaturan. Kode Sumber 4. 22 diimplementasikan pada *controller* yaitu kelas *HomeController.java*.

```

@RequestMapping(value="/deleteConf", method=RequestMethod.GET)
public @ResponseBody String deleteConf(@RequestParam String
confName) throws JsonGenerationException, JsonMappingException,
IOException{
    String sql = "DELETE FROM sinkbd.sink_bd WHERE conf_name =
'" + confName + "'";

    sqlService.dbConnectInsert(configurationService.getAddress(),
configurationService.getUserName(),
configurationService.getPassword(), sql);
    Configuration conf = xmlService.readOneXml(confName);
    sqlService.truncateSinkData(conf);
    xmlService.deleteXml(confName);
    return mapper.writeValueAsString("success");
}

```

Kode Sumber 4. 22 Kode Untuk Menangani Fungsi Hapus Pengaturan

4.4.3 Antarmuka Halaman Ubah Pengaturan

Antarmuka halaman ubah pengaturan merupakan halaman yang diakses Administrator melalui halaman menu utama untuk mengubah pengaturan sesuai kebutuhan. Halaman ubah pengaturan memiliki tampilan yang sama dengan antarmuka halaman tambah pengaturan. Namun pada halaman ini data sudah ada sehingga Administrator cukup mengubah sesuai keinginan. Antarmuka halaman ubah pengaturan diimplementasikan seperti pada Gambar 4. 13.

SinkBD Home

+ Koneksi baru Pengaturan tabel dan kolom ✓ Konfirmasi

Pengaturan disimpan

No	source (127.0.0.1/sinkbd_source)	target (127.0.0.1/sinkbd_target)	Hapus
1	ptk	ptk	<input type="button" value="x"/>
2	pd	pd	<input type="button" value="x"/>

Pencocokan Kolom

Source (127.0.0.1/sinkbd_source)

Masukkan Tabel

Target (127.0.0.1/sinkbd_target)

Masukkan Tabel

Source

Target

Kolom yang dipilih

Gambar 4. 13 Antarmuka Halaman Ubah Pengaturan

Kode Sumber 4. 23 merupakan implementasi dari fungsi ubah pengaturan. Kode Sumber 4. 23 diimplementasikan pada *controller* yaitu kelas `HomeController.java`.

```
@RequestMapping(value="/editConf/{name}",
method=RequestMethod.POST)
public String updateConf(@ModelAttribute("configuration")
Configuration conf){
    xmlService.writeXML(conf);
    sqlService.createTrigger(conf);
    return "redirect:/";
}
```

Kode Sumber 4. 23 Kode Untuk Menangani Ubah Pengaturan

BAB V UJI COBA DAN EVALUASI

Pada bab ini akan dijelaskan uji coba yang dilakukan pada aplikasi yang telah dikerjakan serta analisis dari uji coba yang telah dilakukan. Pembahasan pengujian meliputi lingkungan uji coba, skenario uji coba yang meliputi uji kebenaran dan uji kinerja serta analisis setiap pengujian.

5.1 Lingkungan Uji Coba

Lingkungan uji coba menjelaskan lingkungan yang digunakan untuk menguji pengiriman data berdasarkan pengaturan yang dibuat tugas akhir ini. Lingkungan uji coba meliputi perangkat keras dan perangkat lunak yang dibedakan menjadi 3 jenis yaitu *source* seperti pada Tabel 5. 1, *target* pada Tabel 5. 2, dan server Sinkbd pada Tabel 5. 3.

Tabel 5. 1 Tabel Spesifikasi Lingkungan Pengujian *Source*

Spesifikasi	Deskripsi
Prosesor	Intel® Core™ i3-3240 CPU @3.4GHz
Memory (RAM)	4GB
Tipe sistem	64-bit
Sistem operasi	Windows 8 Professional
Basis data	PostgreSQL

Tabel 5. 2 Tabel Spesifikasi Lingkungan Pengujian *Target*

Spesifikasi	Deskripsi
Prosesor	Intel® Core™ i3-3240 CPU @3.4GHz
Memory (RAM)	4GB
Tipe sistem	64-bit
Sistem operasi	Windows 8 Professional
Basis data	PostgreSQL

Tabel 5. 3 Tabel Spesifikasi Lingkungan Pengujian Server Sinkbd

Spesifikasi	Deskripsi
Prosesor	Intel® Core™ i3-3240 CPU @3.4GHz
Memory (RAM)	4GB
Tipe sistem	64-bit
Sistem operasi	Windows 8 Professional
Basis data	PostgreSQL

5.2 Dasar Pengujian

Pengujian yang dilakukan berupa pengujian fungsionalitas. Pengujian fungsionalitas dilakukan dengan model *black box* untuk menguji pengelolaan pengaturan Sinkbd dan proses sinkronisasi basis data. Pengujian ini dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap kebutuhan benar-benar diimplementasikan dan bekerja seperti yang diharapkan.

5.3 Pengujian Fungsionalitas

Subbab ini menjelaskan tentang skenario pengujian fungsionalitas perangkat lunak pada tugas akhir ini. Pengujian didokumentasikan secara sistematis sebagai tolok ukur keberhasilan sistem.

5.3.1 Pengujian Pengelolaan Pengaturan Sinkbd

Pengujian pengelolaan pengaturan dibagi menjadi beberapa proses yaitu membuat, mengubah, dan menghapus pengaturan. Skenario pengujian membuat pengaturan dijelaskan pada Tabel 5. 4. Skenario pengujian mengubah pengaturan dijelaskan pada Tabel 5. 5. Sementara skenario pengujian menghapus pengaturan dijelaskan pada Tabel 5. 6. Ilustrasi proses pengelolaan pengaturan ditunjukkan pada Gambar 5. 1, dan Gambar 5. 2.

Tabel 5. 4 Skenario Pengujian Membuat Pengaturan

Nomor	PW-01
Nama	Membuat pengaturan baru.
Use Case	UC-W01
Tujuan	Memeriksa fungsi pembuatan pengaturan baru berfungsi atau tidak.
Kondisi awal	Tidak ada pengaturan yang disimpan.
Skenario	<ol style="list-style-type: none"> 1. Administrator memilih tombol tambah pengaturan. 2. Administrator mengisi nama pengaturan, alamat basis data <i>source</i>, dan alamat basis data <i>target</i>. 3. Administrator memilih tabel dan kolom untuk dicocokkan. 4. Administrator memilih pengaturan satu arah atau dua arah. 5. Administrator menyimpan pengaturan dengan menekan tombol simpan.
Masukan	Pengaturan baru
Keluaran yang diharapkan	Pengaturan tersimpan dan terdapat <i>file</i> xml di dalam <i>folder configuration</i> dengan nama yang sama dengan nama pengaturan baru.
Hasil pengujian	Berhasil

Tabel 5. 5 Skenario Pengujian Mengubah Pengaturan

Nomor	PW-02
Nama	Mengubah pengaturan.
Use Case	UC-W01
Tujuan	Memeriksa fungsi mengubah pengaturan berfungsi atau tidak.
Kondisi awal	Pengaturan sebelumnya sudah disimpan.
Skenario	<ol style="list-style-type: none"> 1. Administrator memilih tombol ubah pengaturan berlambang pensil pada salah satu pengaturan.

	<ol style="list-style-type: none"> 2. Administrator memilih tabel dan kolom yang berbeda dari sebelumnya. 3. Administrator memilih pengaturan satu arah atau dua arah. 4. Administrator menyimpan pengaturan dengan menekan tombol simpan.
Masukan	Perubahan pengaturan
Keluaran yang diharapkan	Perubahan pengaturan tersimpan dan terdapat isi <i>file</i> xml di dalam <i>folder configuration</i> dengan nama yang sama namun isinya berbeda.
Hasil pengujian	Berhasil

Tabel 5. 6 Skenario Pengujian Menghapus Pengaturan

Nomor	PW-03
Nama	Mengubah pengaturan.
Use Case	UC-W01
Tujuan	Memeriksa fungsi menghapus pengaturan berfungsi atau tidak.
Kondisi awal	Pengaturan sebelumnya sudah disimpan.
Skenario	<ol style="list-style-type: none"> 1. Administrator memilih tombol hapus pengaturan berwarna merah dengan simbol tanda X pada salah satu pengaturan. 2. Administrator “ya” saat ditanya oleh sistem untuk menghapus pengaturan tersebut.
Masukan	Penghapusan pengaturan
Keluaran yang diharapkan	Tidak terdapat <i>file</i> xml di dalam <i>folder configuration</i> dengan nama yang sama dengan nama pengaturan yang dihapus.
Hasil pengujian	Berhasil

+ Koneksi baru Pengaturan tabel dan kolom Konfirmasi

Nama Pengaturan

ptk-siakad

Source

Masukkan alamat source

10.151.31.33/sia-modul

Sambungkan

Target

Masukkan alamat target

10.151.31.220/siaj

Sambungkan

Previous Next

(a)

+ Koneksi baru Pengaturan tabel dan kolom Konfirmasi

Pengaturan disimpan

No	Source (10.151.31.33/sia-modul)	Target (10.151.31.220/sia)	Hapus
1	ptk	ptk	✖

Pencocokan Kolom

Source (10.151.31.33/sia-modul)

Masukkan Tabel

ptk

Pilih Tabel

Target (10.151.31.220/sia)

Masukkan Tabel

ptk

Pilih Table

Source

🔍 cari

⬆ ⬇

Target

🔍 cari

⬆ ⬇

Kolom yang dipilih

🔍 cari

id_ptk

nip_ptk

nm_ptk

a_ptk_terhapus

status_ptk

🔍 cari

id_ptk

nip_ptk

nm_ptk

a_ptk_terhapus

status_ptk

Simpan

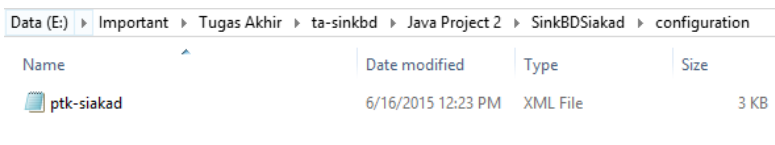
Previous Next

(b)



(c)

Gambar 5. 1 (a,b,c) Ilustrasi Pengujian Proses Pengelolaan Pengaturan



Gambar 5. 2 Ilustrasi File Xml Dalam Folder configuration

5.3.2 Pengujian Sinkronisasi Basis Data

Uji coba ini dilakukan untuk menguji apakah fungsionalitas pengiriman data telah diimplementasikan dengan benar dan berjalan sebagaimana mestinya. Uji coba akan didasarkan pada beberapa skenario untuk menguji validasi pengiriman data Sinkbd. Skenario pengujian terdiri dari 8 bagian yaitu sebagai berikut.

1. Skenario pengiriman seluruh data awal dari satu tabel.
2. Skenario pengiriman perubahan data dari satu tabel.
3. Skenario pengiriman seluruh data awal lebih dari satu tabel.
4. Skenario pengiriman perubahan data lebih dari satu tabel.

5. Skenario pengiriman data menuju tabel yang memiliki relasi dengan tabel lain.
6. Skenario pengiriman data dua arah.
7. Skenario pengiriman data dengan pengaturan khusus.
8. Skenario melanjutkan sinkronisasi yang terputus.

Pengaturan yang digunakan pada skenario pengujian 1 dan 2 ditunjukkan pada Gambar 5. 3. Sedangkan pengaturan yang digunakan pada skenario pengujian 3 dan 4 ditunjukkan pada Gambar 5. 4.

Pengaturan Sinkronisasi Basis Data

No	Nama Pengaturan	Alamat Source	Alamat Target	Sinkronisasi	Lanjutkan	Edit	Hapus
1	ptk-siakad	10.151.31.33/sia-modul	10.151.31.220/sia				

[+ Tambah pengaturan](#)

© Tommy Nurwantoro

Gambar 5. 3 Pengaturan Untuk Pengujian 1 dan 2

Pengaturan Sinkronisasi Basis Data

No	Nama Pengaturan	Alamat Source	Alamat Target	Sinkronisasi	Lanjutkan	Edit	Hapus
1	ptk-pd-siakad	10.151.31.33/sia-modul	10.151.31.220/sia				

[+ Tambah pengaturan](#)

© Tommy Nurwantoro

Gambar 5. 4 Pengaturan Untuk Pengujian 3 dan 4

5.3.2.1 Skenario Uji Coba 1

Skenario uji coba 1 merupakan skenario pengiriman data awal dari salah satu tabel di *source* menuju *target*. Kondisi ini merupakan kondisi dimana pengaturan baru pertama kali dilakukan sinkronisasi. Tabel yang digunakan adalah tabel ptk dengan jumlah data awal 55 baris. Skenario pengujian dijelaskan pada Tabel 5. 7. Ilustrasi perubahan isi tabel diperlihatkan pada Gambar 5. 5, Gambar 5. 6, Gambar 5. 7, dan Gambar 5. 8.

Tabel 5. 7 Skenario Uji Coba 1

Nomor	PS-01
Nama	Sinkronisasi data awal satu tabel
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data awal satu tabel dari <i>source</i> menuju <i>target</i> berfungsi atau tidak.
Kondisi awal	Pengaturan baru dibuat dan isi tabel belum sama.
Skenario	Administrator memilih tombol sinkronisasi.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Berhasil

The image displays three examples of database table synchronization. Each example consists of two screenshots of a table's properties, connected by a blue arrow. In each screenshot, the 'Rows (counted)' value is circled in red. The first screenshot of each pair shows 0 rows, and the second shows 55 rows.

Table Name	Property	Value (Before)	Value (After)
sink_data_pdk_siakad	Name	sink_data_pdk_siakad	sink_data_pdk_siakad
	OID	36643	36643
	Owner	postgres	postgres
	Tablespace	pg_default	pg_default
	ACL		
	Of type		
	Primary key	id_sink_data	id_sink_data
	Rows (estimated)	0	0
	PK factor		
	Rows (counted)	0	55
sinkbd_outbound	Name	sinkbd_outbound	sinkbd_outbound
	OID	36592	36592
	Owner	postgres	postgres
	Tablespace	pg_default	pg_default
	ACL		
	Of type		
	Primary key	id_sink_data	id_sink_data
	Rows (estimated)	0	0
	PK factor		
	Rows (counted)	0	55
sinkbd_inbound	Name	sinkbd_inbound	sinkbd_inbound
	OID	37771	37771
	Owner	postgres	postgres
	Tablespace	pg_default	pg_default
	ACL		
	Of type		
	Primary key	id_sink_data	id_sink_data
	Rows (estimated)	0	0
	PK factor		
	Rows (counted)	0	55

Gambar 5. 5 Perubahan Jumlah Baris Tabel Sinkronisasi

	id_sink_data [PK] serial	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
47	47	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
48	48	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
49	49	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
50	50	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
51	51	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
52	52	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
53	53	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
54	54	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
55	55	public	ptk	I	"92da596d-d72f-48c9-bd38-34a1-2a745e9a-0da9-4bd4-adf1-003-1478ad44-01be-4dd7-b910-17f1-30104c0ff-c89e-4e22-8db7-b7a1-bae98028-9702-43f1-be88-4151-20ccf0ed-1147-4a6e-b8e3-51f1-b32393ac-e32a-4cfc-9b12-b961-34ec113c-6d85-43e3-914d-faf1-9cc8eaaa-e528-4b4c-9700-222"
*					

Gambar 5. 6 Isi Tabel sink_data Pada Source

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)	a_ptk_terhapus boolean	status_ptk boolean
*					

Gambar 5. 7 Isi Tabel Target Sebelum Sinkronisasi

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
47	c30b03b6-93ce-4171-940a-b3a7dfbbe501	0	Sarwoeri, S.Kom, MT.
48	c9fc8d1e-50a9-44d2-b329-d8d45d3478f4	0	Dwi Sunaryono, S.Kom, M.Kom
49	ca9de696-4ede-4ddc-97e5-a6690c57466e	0	Harul Fajrin Ariyani, S.Kom, M.Sc
50	c0b95963-03ca-4bff-bf7e-d3be6342aa33	0	Dr. Royyena Muslim Iyithadi, S.Kom, M
51	d803df1e-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amallah, S.Kom, M.Kom
52	de2590ca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Faticah, S.Kom, M.Kom
53	e4871340-b23a-4947-81a0-2099b6cc1147	0	Fajar Baskoro, S.Kom, MT
54	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
55	f20d607f-3176-4fec-b6ca-ba0c4a9300ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 8 Isi Tabel Target Setelah Sinkronisasi

5.3.2.2 Skenario Uji Coba 2

Skenario uji coba 2 merupakan skenario pengiriman data perubahan dari salah satu tabel di *source* menuju *target*. Kondisi ini merupakan kondisi dimana pengaturan sudah pernah dilakukan sinkronisasi sebelumnya. Tabel yang digunakan

adalah tabel ptk dengan jumlah perubahan data 3 baris yaitu 2 *input* dan 1 *update*. Skenario pengujian dijelaskan pada Tabel 5. 8. Ilustrasi perubahan isi tabel ditunjukkan pada Gambar 5. 9, Gambar 5. 10, Gambar 5. 11, dan Gambar 5. 12.

Tabel 5. 8 Skenario Uji Coba 2

Nomor	PS-02
Nama	Sinkronisasi data perubahan satu tabel
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data perubahan satu tabel dari <i>source</i> menuju <i>target</i> berfungsi atau tidak.
Kondisi awal	Pengaturan sudah pernah dilakukan sinkronisasi dan terjadi perubahan data di <i>source</i> .
Skenario	Administrator memilih tombol sinkronisasi pada pengaturan yang sudah pernah dilakukan sinkronisasi sebelumnya.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Berhasil

Properties	Statistics	Dependencies	Dependents
Property		Value	
Name		sink_data_p01_siakad	
OID		36643	
Owner		postgres	
Tablespace		pg_default	
ACL			
Of type			
Primary key		id_sink_data	
Rows (estimated)		0	
Fill factor			
Rows (counted)		55	
Inherits tables		No	

Properties	Statistics	Dependencies	Dependents
Property		Value	
Name		sink_data_p01_siakad	
OID		36643	
Owner		postgres	
Tablespace		pg_default	
ACL			
Of type			
Primary key		id_sink_data	
Rows (estimated)		0	
Fill factor			
Rows (counted)		58	
Inherits tables		No	

Properties	Statistics	Dependencies	Dependents
Property		Value	
Name		sinkbd_outbound	
OID		36592	
Owner		postgres	
Tablespace		pg_default	
ACL			
Of type			
Primary key		id_sink_data	
Rows (estimated)		0	
Fill factor			
Rows (counted)		55	
Inherits tables		No	

Properties	Statistics	Dependencies	Dependents
Property		Value	
Name		sinkbd_outbound	
OID		36592	
Owner		postgres	
Tablespace		pg_default	
ACL			
Of type			
Primary key		id_sink_data	
Rows (estimated)		0	
Fill factor			
Rows (counted)		3	
Inherits tables		No	

Properties	Statistics	Dependencies	Dependents
Property		Value	
Name		sinkbd_inbound	
OID		37771	
Owner		postgres	
Tablespace		pg_default	
ACL			
Of type			
Primary key		id_sink_data	
Rows (estimated)		0	
Fill factor			
Rows (counted)		55	
Inherits tables		No	

Properties	Statistics	Dependencies	Dependents
Property		Value	
Name		sinkbd_inbound	
OID		37771	
Owner		postgres	
Tablespace		pg_default	
ACL			
Of type			
Primary key		id_sink_data	
Rows (estimated)		0	
Fill factor			
Rows (counted)		3	
Inherits tables		No	

Gambar 5. 9 Perubahan Jumlah Baris Tabel Sinkronisasi

	id_sink_data [PK] serial	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
	50	public	ptk	I	"30104cff-c89e-4e22-8db7-b7a1
	51	public	ptk	I	"bee90028-5702-43f1-be88-415f
	52	public	ptk	I	"20ccf0ed-1147-4a6e-b8e3-51f1
	53	public	ptk	I	"b32393ac-e32a-4cfc-9b12-b96f
	54	public	ptk	I	"34ac113c-6d85-43e3-914d-faf1
	55	public	ptk	I	"9cc8aaaa-e528-4b4c-9700-222f
	56	public	ptk	I	(5e9bdc71-2bdc-40bd-a2ed-ff5f
	57	public	ptk	I	(1df1d6ce-bc7b-4a97-bde4-436f
	58	public	ptk	U	(1df1d6ce-bc7b-4a97-bde4-436f
*					

Gambar 5. 10 Isi Tabel sink_data Pada Source

	id_sink_data [PK] serial	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
	50	public	ptk	I	"30104cff-c89e-4e22-8db7-b7a1
	51	public	ptk	I	"bee90028-5702-43f1-be88-415f
	52	public	ptk	I	"20ccf0ed-1147-4a6e-b8e3-51f1
	53	public	ptk	I	"b32393ac-e32a-4cfc-9b12-b96f
	54	public	ptk	I	"34ac113c-6d85-43e3-914d-faf1
	55	public	ptk	I	"9cc8aaaa-e528-4b4c-9700-222f
	56	public	ptk	I	(5e9bdc71-2bdc-40bd-a2ed-ff5f
	57	public	ptk	I	(1df1d6ce-bc7b-4a97-bde4-436f
	58	public	ptk	U	(1df1d6ce-bc7b-4a97-bde4-436f
*					

Gambar 5. 11 Isi Tabel Target Sebelum Sinkronisasi

	id_sink_data [PK] serial	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
	50	public	ptk	I	"30104cff-c89e-4e22-8db7-b7a1
	51	public	ptk	I	"bee90028-5702-43f1-be88-415f
	52	public	ptk	I	"20ccf0ed-1147-4a6e-b8e3-51f1
	53	public	ptk	I	"b32393ac-e32a-4cfc-9b12-b96f
	54	public	ptk	I	"34ac113c-6d85-43e3-914d-faf1
	55	public	ptk	I	"9cc8aaaa-e528-4b4c-9700-222f
	56	public	ptk	I	(5e9bdc71-2bdc-40bd-a2ed-ff5f
	57	public	ptk	I	(1df1d6ce-bc7b-4a97-bde4-436f
	58	public	ptk	U	(1df1d6ce-bc7b-4a97-bde4-436f
*					

Gambar 5. 12 Isi Tabel Target Setelah Sinkronisasi

5.3.2.3 Skenario Uji Coba 3

Skenario uji coba 3 merupakan skenario pengiriman data awal lebih dari satu tabel di *source* menuju *target*. Kondisi ini merupakan kondisi dimana pengaturan belum pernah dilakukan sinkronisasi sebelumnya. Tabel yang digunakan adalah tabel ptk dengan jumlah data awal 55 baris dan tabel pd dengan jumlah data awal 207 baris sehingga total data yang dikirimkan 262 baris. Skenario pengujian dijelaskan pada Tabel 5. 9. Ilustrasi perubahan isi data ditunjukkan pada Gambar 5. 13, Gambar 5. 14, Gambar 5. 15, dan Gambar 5. 16.

Tabel 5. 9 Skenario Uji Coba 3

Nomor	PS-03
Nama	Sinkronisasi data awal beberapa tabel
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data awal lebih dari satu tabel dari <i>source</i> menuju <i>target</i> berfungsi atau tidak.
Kondisi awal	Pengaturan baru dibuat dan isi tabel belum sama.
Skenario	Administrator memilih tombol sinkronisasi.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Berhasil

Property	Value
Name	sink_data_ptk_pd_slakad
OID	36617
Owner	postgres
Tablespace	pg_default
ACL	
Of type	
Primary key	id_sink_data
Rows (estimated)	0
Fill factor	
Rows (counted)	0
Inherits tables	No

Property	Value
Name	sink_data_ptk_pd_slakad
OID	36617
Owner	postgres
Tablespace	pg_default
ACL	
Of type	
Primary key	id_sink_data
Rows (estimated)	0
Fill factor	
Rows (counted)	262
Inherits tables	No

Property	Value
Name	sinkbd_outbound
OID	36592
Owner	postgres
Tablespace	pg_default
ACL	
Of type	
Primary key	id_sink_data
Rows (estimated)	0
Fill factor	
Rows (counted)	0
Inherits tables	No

Property	Value
Name	sinkbd_outbound
OID	36592
Owner	postgres
Tablespace	pg_default
ACL	
Of type	
Primary key	id_sink_data
Rows (estimated)	0
Fill factor	
Rows (counted)	262
Inherits tables	No

Property	Value
Name	sinkbd_inbound
OID	37771
Owner	postgres
Tablespace	pg_default
ACL	
Of type	
Primary key	id_sink_data
Rows (estimated)	0
Fill factor	
Rows (counted)	0
Inherits tables	No

Property	Value
Name	sinkbd_inbound
OID	37771
Owner	postgres
Tablespace	pg_default
ACL	
Of type	
Primary key	id_sink_data
Rows (estimated)	0
Fill factor	
Rows (counted)	262
Inherits tables	No

Gambar 5. 13 Perubahan Jumlah Baris Tabel Sinkronisasi

The screenshot shows a window titled 'Edit Data - PC-Galih (10.151.31.33:5432) - sia-modul - sinkbd.sink_data_pik_pd_siakad'. The table structure is as follows:

	id_sink_data [PK] serial	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
254	254	public	pd	I	"5921b864-0a70-420d-b8ec-c60"
255	255	public	pd	I	"4b9ee700-7bd3-4203-90e9-390"
256	256	public	pd	I	"82e8729e-651f-4591-a740-437"
257	257	public	pd	I	"3ea50b57-e9c0-4135-b210-454"
258	258	public	pd	I	"e691025e-4327-4f06-b0b1-e70"
259	259	public	pd	I	"10bf536a-28f3-4a7e-870b-588"
260	260	public	pd	I	"ed8f7ed6-caa4-415e-b37c-be7"
261	261	public	pd	I	"040125a2-0306-44f5-8a57-eed"
262	262	public	pd	I	"153c410c-16d5-4741-be47-b4a"

Gambar 5. 14 Isi Tabel sink_data Pada Source

The top screenshot shows a window titled 'Edit Data - PC-Tommy (10.151.31.220:5432) - sia - pik'. The table structure is:

	id_pik [PK] uuid	nip_pik character varying(50)	nom_pik character varying(255)	a_pik_terbapak boolean	status_pik boolean
*					

The bottom screenshot shows a window titled 'Edit Data - PC-Tommy (10.151.31.220:5432) - sia - pd'. The table structure is:

	id_pd [PK] uuid	id_pik uuid	nama_pd character varying(50)	nom_pd character varying(255)	a_pd_terbapak boolean	angkatan_pd integer
*						

Gambar 5. 15 Isi Tabel Target Sebelum Sinkronisasi

The image displays two screenshots of a database management tool interface, likely MySQL Workbench, showing data synchronization between two tables. The top screenshot shows the 'ptk' table, and the bottom screenshot shows the 'pd' table.

Table 1: ptk

id_ptk [PK] uuid	nama_ptk character varying(50)	nama_ptk character varying(255)
47	c90b03b6-93ce-4371-945a-b5e7dfbba501	Sarwoeri, S.Kom, MT.
48	9f0d81e-90a9-44d0-9329-d8d493d795f4	Dwi Sunaryono, S.Kom, M.Kom
49	ca9de696-4edc-4d8c-97e5-a669cc07466e	Buruz Fajrin Ariyani, S.Kom, M.Sc
50	cfb99943-03ce-48f7-8f7e-d8ee6342aa33	Dr. Ropyssa Muslim Izzihadi, S.Kom, M
51	d003df1a-094f-4c02-8c66-262d60ab2990	Sulqis Analiah, S.Kom, M.Kom
52	de2990cc-beaf-42ee-fc0f-880490e2ea30	Dr. Chaetise Fatmahan, S.Kom, M.Kom
53	e4871340-b23e-4947-81e2-2099b6cc1167	Fajar Barokro, S.Kom, MT
54	ec3b9184-4a83-4542-9a29-fff788270789	Radityo Anggoro, S.Kom, M.Sc.
55	f20d607f-3176-4fec-b6ca-baceae93e0ee	Ahmad Nurul Basri, S.Kom, M.Sc.

Table 2: pd

id_pd [PK] uuid	nama_ptk uuid	nama_ptk character varying(50)	nama_ptk character varying(255)
199	f6f44194-9799-4427-9ac6-bdca19e636cc	9111100094	Dwi
200	f6f6ba45-745b-42f9-9e20-cd3aac2f8cee	9111100032	Sar
201	f7961f00-6048-4162-8938-e5e03790843f	9111100090	Buruz
202	f8b81404-7a63-4004-bd70-2314e87e94b1	9111100118	Dr.
203	f90ca00f-83fa-448e-b420-18911433e034	9111100010	Nur
204	fde7030a-4fcb-4c04-2d7d-6bc2d36cc254	9111100043	Dwi
205	fe6e6d8a-6a7e-49a8-b78c-a18512b48005	9111100004	Sulqis
206	fed5b91c-c530-42a9-ba01-060eab2052e	9111100027	Nur
207	fedc776e-7034-43e0-8104-1273aa8e73e	9111100168	Nur

Gambar 5. 16 Isi Tabel *Target* Setelah Sinkronisasi

5.3.2.4 Skenario Uji Coba 4

Skenario uji coba 4 merupakan skenario pengiriman data perubahan lebih dari satu tabel di *source* menuju *target*. Kondisi ini merupakan kondisi dimana pengaturan sudah pernah dilakukan sinkronisasi sebelumnya. Tabel yang digunakan adalah tabel ptk dengan jumlah perubahan data 3 baris yaitu 2 *insert* dan 1 *update* dan tabel pd dengan jumlah perubahan data 2 baris yaitu 2 *insert* sehingga total data yang dikirimkan 5 baris perubahan. Skenario pengujian dijelaskan pada Tabel 5. 10. Ilustrasi perubahan isi data ditunjukkan pada Gambar 5. 17, Gambar 5. 18, Gambar 5. 19, dan Gambar 5. 20.

Tabel 5. 10 Skenario Uji Coba 4

Nomor	PS-04
Nama	Sinkronisasi data perubahan beberapa tabel
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data perubahan lebih dari satu tabel dari <i>source</i> menuju <i>target</i> berfungsi atau tidak.
Kondisi awal	Pengaturan sudah pernah dilakukan sinkronisasi dan terjadi perubahan data di <i>source</i> .
Skenario	Administrator memilih tombol sinkronisasi pada pengaturan yang sudah pernah dilakukan sinkronisasi sebelumnya.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Berhasil

The diagram illustrates the change in the number of rows in three tables during synchronization. Each table's properties are shown in two states: before and after synchronization, with the 'Rows (counted)' value circled in red. Blue arrows indicate the transition between states.

Table Name	Property	Value (Before)	Value (After)
sink_data_ptk_pd_siakad	Rows (counted)	262	267
	Rows (estimated)	0	0
sinkbd_outbound	Rows (counted)	262	5
	Rows (estimated)	0	0
sinkbd_inbound	Rows (counted)	262	5
	Rows (estimated)	0	0

Gambar 5. 17 Perubahan Jumlah Baris Tabel Sinkronisasi

id	social	table_name	event	raw_data
259	publisc	pd	2	"1207316a-2823-8a7e-d708-500f"
260	publisc	pd	2	"e6077a03-0aa9-815e-8370-0e7f"
261	publisc	pd	2	"e40123a2-0308-4473-9a17-8ed"
262	publisc	pd	2	"1133e410e-1685-4741-8ea17-84a"
263	publisc	pd	2	"895e8113-3d4f-431e-fc1a-b07f"
264	publisc	pd	2	"8a2f0c03-19a3-8a0c-85a3-a38f"
265	publisc	pta	2	"34883c20-71a3-817e-8a98-3a8f"
266	publisc	pta	2	"e7eab102-8280-8103-8290-848f"
267	publisc	pta	2	"e7eab102-8280-8103-8290-848f"

Gambar 5. 18 Isi Tabel sink_data Pada Source

id	mpg_mtk	nm_mtk
47	c30b03e6-810e-817e-941e-85e7d3b8e111	Sarwatri, S.Kom, M.
48	c8f081e-51e9-4482-8219-88465d47104	Dei Suharyono, S.Kom, M.Kom
49	ca8e498-8e08-8882-87e5-8480c07040e	Muzli Fajris Arsyani, S.Kom, M.Sc
50	c0d18e83-c10e-887f-817e-838e81a2aa27	Dr. Ropyana Nuzila Iyiliani, S.Kom, M
51	8903a71e-8947-8e02-8e86-f42407ab2190	Rizqie Amaliah, S.Kom, M.Kom
52	8e1590e8-8ea8-42ae-818f-885407e2ea30	Dr. Charissa Patricia, S.Kom, M.Kom
53	4e171340-823e-8967-81a0-8098640cc187	Fajar Sarocco, S.Kom, M.
54	8c28e014-8483-4282-8a29-82f78270709	Radriyo Anggoro, S.Kom, M.Sc.
55	f2a8877f-3174-87ec-840e-8a0e8ab300ee	Ahmad Rizki Sastru, S.Kom, M.Sc.

Gambar 5. 19 Isi Tabel Target Sebelum Sinkronisasi

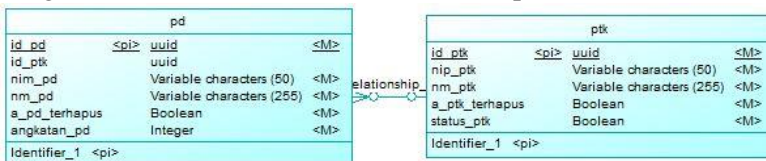
The image shows two screenshots of a data editor window titled 'Edit Data - PC-Tommy (10.151.33.220:5432) - via - ptk'. The top screenshot shows a table with columns 'id_pnk [PK] uuid', 'nip_pnk character varying(50)', and 'nm_pnk character varying(255)'. The bottom screenshot shows the same table with a different set of data rows.

id_pnk [PK] uuid	nip_pnk character varying(50)	nm_pnk character varying(255)
c30b9306-830e-4175-941e-85e7d2f8e011		Sarwadi, S.Rm, ST
c81c831e-51e9-4423-8029-8b4b82947124		Dr. Suharyono, S.Rm, N.Rm
ca98e410-6e1e-42a2-97e5-a8f9c007e40e		Muzli Fauziz Ariyani, S.Rm, N.St
c7b910e3-030e-4872-8d7e-83e8342aa333		Dr. Royana Nurliq Iyuliani, S.Rm, N
80236f1e-094f-4e02-8e90-f42480ab2990		Roliga Amaliah, S.Rm, N.Rm
de198cc1-ba6f-42ee-81df-8b540e0ea310		Dr. Charissa Fariqah, S.Rm, N.Rm
e4f71340-023e-4947-81a0-209864cc1187		Fajar Basoro, S.Rm, ST
ec2e8c1e-4e05-4262-8e99-22f78d27e789		Radityo Anggoro, S.Rm, N.St.
f2e88072-3176-426c-840a-9a1e8a930bee		Ahmad Nurul Basri, S.Rm, N.St.

Gambar 5. 20 Isi Tabel *Target* Setelah Sinkronisasi

5.3.2.5 Skenario Uji Coba 5

Skenario uji coba 5 merupakan skenario pengiriman data dari tabel di *source* menuju tabel pada *target* yang memiliki relasi dengan tabel lain. Relasi tabel dicontohkan pada Gambar 5. 21.



Gambar 5. 21 Relasi Tabel Pengujian pd dan ptk

Pada Gambar 5. 21, tabel pd memiliki relasi *one to many* dengan tabel ptk. Sehingga seharusnya jika data *foreign key* tidak

ada di tabel ptk, maka data tidak dapat dimasukkan. Uji coba akan dilakukan dengan sinkronisasi tabel pd yang memiliki 55 baris data. Skenario pengujian dijelaskan pada Tabel 5. 11.

Tabel 5. 11 Skenario Uji Coba 5

Nomor	PS-05
Nama	Sinkronisasi tabel yang memiliki relasi
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data tabel dari <i>source</i> menuju <i>target</i> yang memiliki relasi berfungsi atau tidak.
Kondisi awal	Pengaturan baru dibuat dan isi tabel di <i>target</i> masih kosong.
Skenario	Administrator memilih tombol sinkronisasi pada halaman pengaturan.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Gagal

Uji coba dengan kondisi seperti ini gagal. Pesan yang ditampilkan server seperti pada Gambar 5. 22. Pesan *error* tersebut menjelaskan bahwa data yang sudah valid tidak dapat dimasukkan ke dalam tabel. Penjelasan akan dibahas pada bagian evaluasi pengujian.

```

Console [X] Markers [X] Progress [X]
Tomcat v8.0 Server at localhost [Apache Tomcat/8.0.45] bin\java.exe (Jun 19, 2015, 2:10:34 PM)
at java.lang.Thread.run(Unknown Source)
org.postgresql.util.PSQLException: ERROR: insert or update on table "pd" violates foreign key constraint "fk_pd_wali_ptk"
details: key (id_gtk) (2b873dc-fb2f-4b3-8e6d-d0b88d0c198) is not present in table "ptk".
at org.postgresql.core.v3.QueryExecutorImpl.receiveErrorResponse(QueryExecutorImpl.java:2278)
at org.postgresql.core.v3.QueryExecutorImpl.processResults(QueryExecutorImpl.java:1994)
at org.postgresql.core.v3.QueryExecutorImpl.execute(QueryExecutorImpl.java:255)
at org.postgresql.jdbc2.AbstractJdbc2Statement.execute(AbstractJdbc2Statement.java:578)
at org.postgresql.jdbc2.AbstractJdbc2Statement.execute(Unknown Source)
at org.postgresql.jdbc2.AbstractJdbc2Statement.executeUpdate(AbstractJdbc2Statement.java:486)
at com.siakad.sinkbd.service.SgService.dbConnectInsert(SgService.java:137)
at com.siakad.sinkbd.service.SgService.insertValidData(SgService.java:633)
at com.siakad.sinkbd.HomeController.doSync(HomeController.java:211)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethod)
at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
at java.lang.reflect.Method.invoke(Unknown Source)
at org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(InvocableHandlerMethod.java:221)
at org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.java:137)
at org.springframework.web.servlet.mvc.method.annotation.ServletInvocableHandlerMethod.invokeAndHandle(ServletInvocableHandlerMethod.java:118)
at org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.invokeHandlerMethod(RequestMappingHandlerAdapter.java:776)
at org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handleInternal(RequestMappingHandlerAdapter.java:765)
at org.springframework.web.servlet.mvc.method.AbstractHandlerMethodAdapter.handle(AbstractHandlerMethodAdapter.java:85)
at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:959)
at org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:893)
at org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.java:968)
at org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:857)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:618)

```

Gambar 5. 22 Pesan Error Memasukkan Data Valid

5.3.2.6 Skenario Uji Coba 6

Skenario uji coba 6 merupakan skenario pengiriman data dua arah yaitu pengiriman data dilakukan 2 tahapan yaitu dari *source* menuju *target* kemudian dari *target* menuju *source*. Kondisi ini memungkinkan merupakan kondisi kedua tabel sudah terisi dan setelah dilakukan sinkronisasi memiliki data yang sama. Skenario pengujian dijelaskan pada Tabel 5. 12. Ilustrasi perubahan isi data ditunjukkan pada Gambar 5. 23, Gambar 5. 24, Gambar 5. 25, dan Gambar 5. 26.

Tabel 5. 12 Skenario Uji Coba 6

Nomor	PS-06
Nama	Sinkronisasi tabel dua arah
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data tabel dari <i>source</i> menuju <i>target</i> dan dari <i>target</i> menuju <i>source</i> berfungsi atau tidak.
Kondisi awal	Tabel di <i>source</i> dan tabel di <i>target</i> sudah berisi data yang berbeda.
Skenario	Administrator memilih tombol sinkronisasi pada halaman pengaturan.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Berhasil

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
47	244a6fc1-2f02-4904-bf95-cd79dac00fd0	0	Shintami Chusnul H., S.Kom
48	20cc0fed-1147-4a6e-b8e3-51f182def6f9	197906262005012002	Umi Laili Yuhana, S.Kom, M.Sc.
49	1e7181c8-a5d4-4136-a962-856050ec2c49	0	MM. Irfan Subakti, S.Kom, M.Sc.Eng
50	19e32459-9b82-4090-b9f9-c6fc3f62474d	0	Arya Yudhi Wijaya, S.Kom, M.Kom
51	186b90a6-6d36-41c4-aae7-e6812f996bbe	0	Ir. FX. Arunanto, M.Sc.
52	182f875b-1b7f-4fda-b021-45236ab03119	0	Dr. Tohari Ahmad, S.Kom, MIT
53	17145f03-459f-4537-8f5d-4199b084b93f	0	Abdul Munif, S.Kom, M.Sc.
54	1478ad44-01be-4dd7-b910-17f47695d101	0	Erina Letivina Anggraini, S.Kom
55	12b8e09d-070d-4925-8657-705ee2f6bb0c	51000	Prof.Ir. Handayani Tjandrasa, M.Sc, Ph.
*			

Gambar 5. 23 Isi Tabel ptk *Source* Sebelum Sinkronisasi

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
53	ca9de696-4ede-4ddc-97e5-a669cc57466e	0	Nurul Fajrin Ariyani, S.Kom, M.Sc
54	cd5c4c3a-4aae-402f-929d-a3216444d066	197906262005015004	Dummy Insert ptk 4
55	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.K
56	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
57	d95349e4-4c73-47ce-bdcf-732cb5b391fd	197906262005015006	Dummy Insert ptk 6
58	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
59	e4871340-b23a-4947-81a0-2099b6ccc1167	0	Fajar Baskoro, S.Kom, MT
60	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
61	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 24 Isi Tabel ptk *Source* Setelah Sinkronisasi

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)	a_ptk_terhapus boolean	status_pt boolean
1	0e3d47ba-ea8b-4fca-b613-3143959e064b	197906262005015004	Dummy Insert ptk 4	FALSE	TRUE
2	1527ee92-7ec2-4df7-8423-45764c66c405	197906262005015003	Dummy Insert ptk 3	FALSE	TRUE
3	3d72ce92-77da-4462-bcc4-0111ee637cf2	197906262005015006	Dummy Insert ptk 6	FALSE	TRUE
4	54629c3e-decb-423b-a790-8017685e63c7	197906262005015002	Dummy Insert ptk 2	FALSE	TRUE
5	7cae1428-c6d7-43aa-a771-12477e692b8e	197906262005015001	Dummy Insert ptk 1	FALSE	TRUE
6	e47faddf-7f3e-46cf-a317-d815d856d5d8	197906262005015005	Dummy Insert ptk 5	FALSE	TRUE
*					

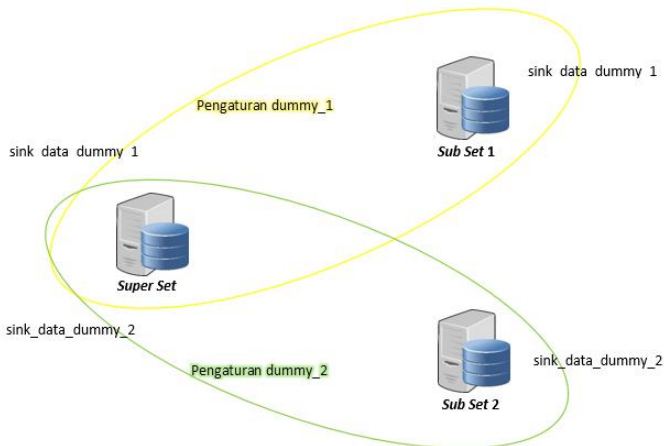
Gambar 5. 25 Isi Tabel ptk *Target* Sebelum Sinkronisasi

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
53	ca9de696-4ede-4ddc-97e5-a669cc57466e	0	Nurul Fajrin Ariyani, S.Kom, M.Sc
54	cd5c4c3a-4aae-402f-929d-a3216444d066	197906262005015004	Dummy Insert ptk 4
55	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M
56	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
57	d95349e4-4c73-47ce-bdcf-732cb5b391fd	197906262005015006	Dummy Insert ptk 6
58	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
59	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
60	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
61	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 26 Isi Tabel ptk *Target* Setelah Sinkronisasi

5.3.2.7 Skenario Uji Coba 7

Skenario uji coba 7 merupakan skenario pengiriman data pada pengaturan khusus seperti pada Gambar 5. 27.



Gambar 5. 27 Pengaturan Khusus Uji Coba 7

Pada skenario ini terdapat 2 pengaturan dan 3 basis data. Salah satu basis data disebut *Super Set* yaitu basis data yang tersambung dengan 2 pengaturan sinkronisasi. Basis data lainnya yang masing-masing memiliki 1 pengaturan disebut *Sub Set*. Kedua pengaturan yang ada merupakan pengaturan 2 arah.

Skenario pengujian secara bertahap dijelaskan pada Tabel 5. 13, Tabel 5. 14, dan Tabel 5. 15.

Tabel 5. 13 Skenario Uji Coba 7 Bagian 1

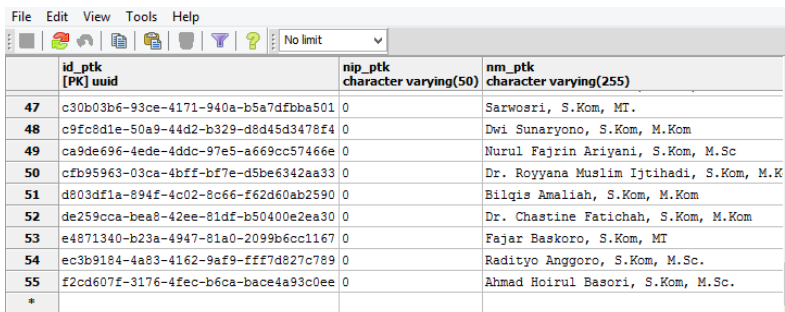
Nomor	PS-07
Nama	Sinkronisasi pengaturan khusus dua arah
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data tabel dari basis data yang memiliki lebih dari satu pengaturan berfungsi atau tidak.
Kondisi awal	Pengaturan dummy_1 dan dummy_2 sudah dibuat dan belum dilakukan sinkronisasi.
Skenario	Administrator memilih tombol sinkronisasi pada pengaturan dummy_1.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>sub set</i> 1 sama dengan isi tabel di <i>super set</i> .
Hasil pengujian	Berhasil

Tabel 5. 14 Skenario Uji Coba 7 Bagian 2

Nomor	PS-08
Nama	Sinkronisasi pengaturan khusus dua arah
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data tabel dari basis data yang memiliki lebih dari satu pengaturan berfungsi atau tidak.
Kondisi awal	Pengaturan dummy_1 dan dummy_2 sudah dibuat dan dummy_1 sudah dilakukan sinkronisasi.
Skenario	Administrator memilih tombol sinkronisasi pada pengaturan dummy_2.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>sub set</i> 2 sama dengan isi tabel di <i>super set</i> .
Hasil pengujian	Berhasil

Tabel 5. 15 Skenario Uji Coba 7 Bagian 3

Nomor	PS-09
Nama	Sinkronisasi pengaturan khusus dua arah
Use Case	UC-W02
Tujuan	Memeriksa fungsi sinkronisasi data tabel dari basis data yang memiliki lebih dari satu pengaturan berfungsi atau tidak.
Kondisi awal	Pengaturan dummy_1 dan dummy_2 sudah dibuat dan dummy_2 sudah dilakukan sinkronisasi.
Skenario	Administrator memilih tombol sinkronisasi pada pengaturan dummy_1 kedua kalinya.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>sub set 2</i> sama dengan isi tabel di <i>sub set 1</i> .
Hasil pengujian	Berhasil



	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
47	c30b03b6-93ce-4171-940a-b5a7dfbba501	0	Sarwosri, S.Kom, MT.
48	c9fc8d1e-50a9-44d2-b329-d8d45d3478f4	0	Dwi Sunaryono, S.Kom, M.Kom
49	ca9de696-4ede-4ddc-97e5-a669cc57466e	0	Nurul Fajrin Ariyani, S.Kom, M.Sc
50	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.K
51	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
52	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
53	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
54	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
55	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 28 Isi Tabel ptk Sub Set 1 Sebelum Sinkronisasi

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
52	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.K
53	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
54	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
55	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
56	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
57	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
58	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
59	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
60	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 29 Isi Tabel ptk *Sub Set 1* Setelah Sinkronisasi dummy_1

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
52	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.K
53	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
54	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
55	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
56	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
57	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
58	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
59	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
60	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 30 Isi Tabel ptk *Sub Set 1* Setelah Sinkronisasi dummy_2

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
56	0c2e202e-e03d-42e0-bd1b-d06930e1a7d0	197906262005015001	Superset Insert ptk 1
57	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.K
58	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
59	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
60	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
61	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
62	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
63	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
64	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
65	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 31 Isi Tabel ptk *Sub Set 1* Setelah Sinkronisasi dummy_1
Kedua Kalinya

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)	a_ptk terhapus boolean	status_boolea
1	56b6bc98-9e1b-47cc-a931-3cd68d907593	197906262005015005	Superset Insert ptk 5	FALSE	TRUE
2	cc2e828e-e03d-42e0-8bfb-bb695ce1a7d0	197906262005015001	Superset Insert ptk 1	FALSE	TRUE
3	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3	FALSE	TRUE
4	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2	FALSE	TRUE
5	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4	FALSE	TRUE
*					

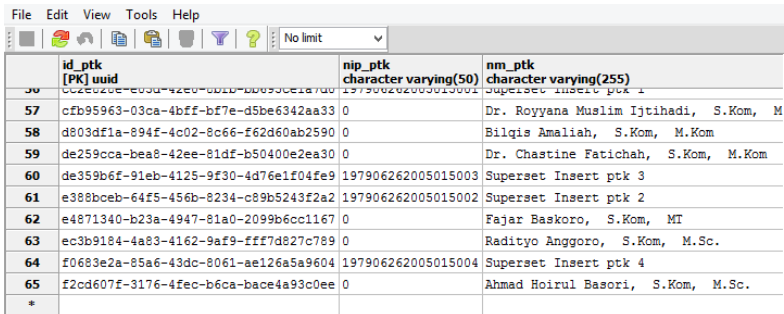
Gambar 5. 32 Isi Tabel ptk *Super Set* Sebelum Sinkronisasi

	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
52	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.
53	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
54	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Faticah, S.Kom, M.Kom
55	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
56	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
57	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
58	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
59	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
60	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 33 Isi Tabel ptk *Super Set* Setelah Sinkronisasi dummy_1

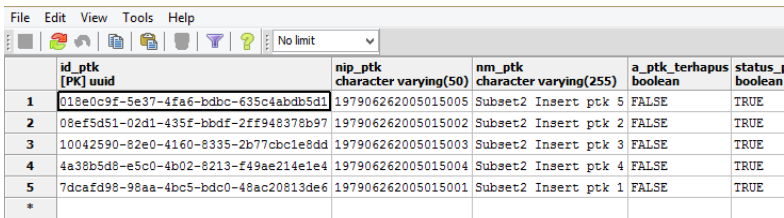
	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
56	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M.
58	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
59	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Faticah, S.Kom, M.Kom
60	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
61	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
62	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
63	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
64	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
65	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 34 Isi Tabel ptk *Super Set* Setelah Sinkronisasi dummy_2



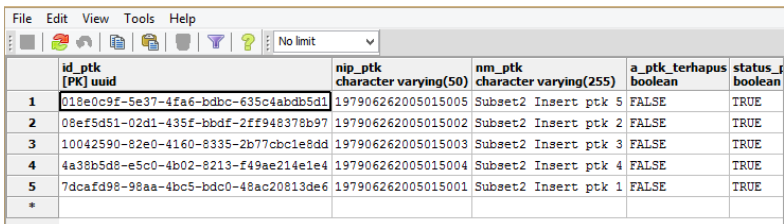
	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
56	cc2e202e-e030-42e0-b21b-bb9930e1ad	197906262005015001	Superset insert ptk 1
57	cfb95963-03ca-4bf7-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom, M
58	d803dfla-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
59	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Faticah, S.Kom, M.Kom
60	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
61	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
62	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
63	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
64	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
65	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 35 Isi Tabel ptk *Super Set* Setelah Sinkronisasi dummy_1 Kedua Kalinya



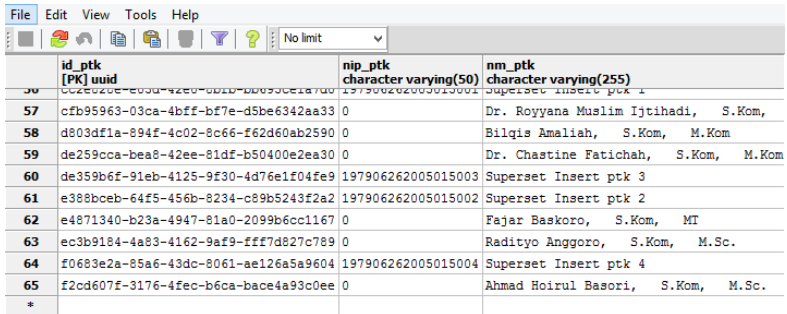
	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)	a_ptk_terhapus boolean	status boolean
1	018e0c9f-5e37-4fa6-bdbc-635c4abdb5d1	197906262005015005	Subset2 Insert ptk 5	FALSE	TRUE
2	08ef5d51-02d1-435f-bbdf-2ff948378b97	197906262005015002	Subset2 Insert ptk 2	FALSE	TRUE
3	10042590-82e0-4160-8335-2b77cbc1e8dd	197906262005015003	Subset2 Insert ptk 3	FALSE	TRUE
4	4a38b5d8-e5c0-4b02-8213-f49ae214e1e4	197906262005015004	Subset2 Insert ptk 4	FALSE	TRUE
5	7dcadf98-98aa-4bc5-bdc0-48ac20813de6	197906262005015001	Subset2 Insert ptk 1	FALSE	TRUE
*					

Gambar 5. 36 Isi Tabel ptk *Sub Set 2* Sebelum Sinkronisasi



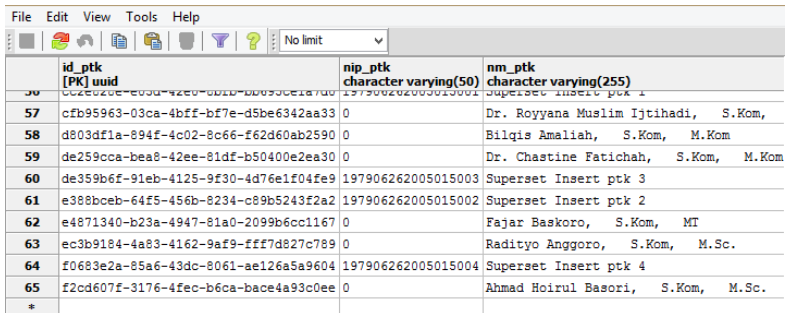
	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)	a_ptk_terhapus boolean	status boolean
1	018e0c9f-5e37-4fa6-bdbc-635c4abdb5d1	197906262005015005	Subset2 Insert ptk 5	FALSE	TRUE
2	08ef5d51-02d1-435f-bbdf-2ff948378b97	197906262005015002	Subset2 Insert ptk 2	FALSE	TRUE
3	10042590-82e0-4160-8335-2b77cbc1e8dd	197906262005015003	Subset2 Insert ptk 3	FALSE	TRUE
4	4a38b5d8-e5c0-4b02-8213-f49ae214e1e4	197906262005015004	Subset2 Insert ptk 4	FALSE	TRUE
5	7dcadf98-98aa-4bc5-bdc0-48ac20813de6	197906262005015001	Subset2 Insert ptk 1	FALSE	TRUE
*					

Gambar 5. 37 Isi Tabel ptk *Sub Set 2* Setelah Sinkronisasi dummy_1



	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
56	c628e2de-e03d-42eb-b01b-bb93cc1a7d0	197906262005015001	Superset Insert ptk 1
57	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom,
58	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
59	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
60	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
61	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
62	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
63	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
64	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
65	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 38 Isi Tabel ptk *Sub Set 2* Setelah Sinkronisasi dummy_2



	id_ptk [PK] uuid	nip_ptk character varying(50)	nm_ptk character varying(255)
56	c628e2de-e03d-42eb-b01b-bb93cc1a7d0	197906262005015001	Superset Insert ptk 1
57	cfb95963-03ca-4bff-bf7e-d5be6342aa33	0	Dr. Royyana Muslim Ijtihadi, S.Kom,
58	d803df1a-894f-4c02-8c66-f62d60ab2590	0	Bilqis Amaliah, S.Kom, M.Kom
59	de259cca-bea8-42ee-81df-b50400e2ea30	0	Dr. Chastine Fatichah, S.Kom, M.Kom
60	de359b6f-91eb-4125-9f30-4d76e1f04fe9	197906262005015003	Superset Insert ptk 3
61	e388bceb-64f5-456b-8234-c89b5243f2a2	197906262005015002	Superset Insert ptk 2
62	e4871340-b23a-4947-81a0-2099b6cc1167	0	Fajar Baskoro, S.Kom, MT
63	ec3b9184-4a83-4162-9af9-fff7d827c789	0	Radityo Anggoro, S.Kom, M.Sc.
64	f0683e2a-85a6-43dc-8061-ae126a5a9604	197906262005015004	Superset Insert ptk 4
65	f2cd607f-3176-4fec-b6ca-bace4a93c0ee	0	Ahmad Hoirul Basori, S.Kom, M.Sc.
*			

Gambar 5. 39 Isi Tabel ptk *Sub Set 2* Setelah Sinkronisasi dummy_1 Kedua Kalinya

Gambar 5. 28 hingga Gambar 5. 39 merupakan ilustrasi dari skenario uji coba 7. Pada gambar-gambar tersebut menunjukkan tahapan-tahapan perubahan data yang terjadi.

5.3.2.8 Skenario Uji Coba 8

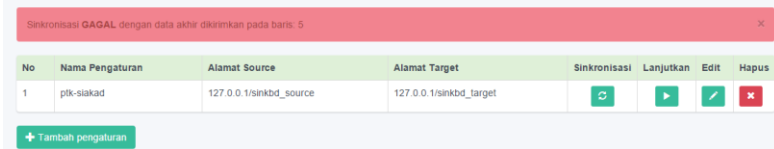
Skenario uji coba 8 merupakan skenario melanjutkan sinkronisasi yang terputus. Skenario ini untuk mengantisipasi ketika pengiriman data ternyata tidak valid akibat sambungan yang terputus. Uji coba dilakukan dengan mengubah status valid pada kode program untuk menciptakan suasana sambungan terputus. Kemudian isi tabel *target* dihapus setelah baris pada status valid yang sudah ditetapkan. Status valid diubah menjadi 5

sehingga pengiriman data valid sampai baris ke 5 saja. Skenario pengujian dijelaskan pada Tabel 5. 16. Ilustrasi perubahan isi tabel diperlihatkan pada Gambar 5. 41, Gambar 5. 42, dan Gambar 5. 43.

Tabel 5. 16 Skenario Uji Coba 8

Nomor	PS-10
Nama	Melanjutkan sinkronisasi yang terputus
Use Case	UC-W03
Tujuan	Memeriksa fungsi melanjutkan sinkronisasi data yang terputus berfungsi atau tidak.
Kondisi awal	Sinkronisasi mengeluarkan peringatan gagal sinkronisasi.
Skenario	Administrator memilih tombol lanjutkan sinkronisasi.
Masukan	Aktifitas sinkronisasi.
Keluaran yang diharapkan	Isi tabel yang dikirimkan dari <i>source</i> sama dengan isi tabel yang diterima <i>target</i> .
Hasil pengujian	Berhasil

Pengaturan Sinkronisasi Basis Data



The screenshot shows a web interface for database synchronization. At the top, a red error message states: "Sinkronisasi GAGAL dengan data akhir dikirimkan pada baris: 5". Below this is a table with columns: No, Nama Pengaturan, Alamat Source, Alamat Target, Sinkronisasi, Lanjutkan, Edit, and Hapus. The table contains one row with the following data: No: 1, Nama Pengaturan: plik-siakad, Alamat Source: 127.0.0.1/sinkbd_source, Alamat Target: 127.0.0.1/sinkbd_target. The 'Sinkronisasi' column has a refresh icon, 'Lanjutkan' has a play button, 'Edit' has a pencil icon, and 'Hapus' has a red X icon. At the bottom left, there is a green button labeled "+ Tambah pengaturan".

No	Nama Pengaturan	Alamat Source	Alamat Target	Sinkronisasi	Lanjutkan	Edit	Hapus
1	plik-siakad	127.0.0.1/sinkbd_source	127.0.0.1/sinkbd_target				

+ Tambah pengaturan

Gambar 5. 40 Tampilan Pesan Sinkronisasi Gagal

Gambar 5. 40 merupakan kondisi setelah sinkronisasi dilakukan namun terjadi kegagalan. Tombol untuk melanjutkan sinkronisasi yang pada awalnya tidak aktif sekarang menjadi aktif.

	id_sink_data [PK] serial	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
47	47	public	ptk	I	"92da596d-d72f-48c9-bd38-343c"
48	48	public	ptk	I	"2a745e9a-0da9-4bd4-ad61-003c"
49	49	public	ptk	I	"1478ad44-01be-4dd7-b910-17fa"
50	50	public	ptk	I	"30104cff-c89e-4e22-8db7-b7a1"
51	51	public	ptk	I	"bae98028-5702-43f1-be88-415c"
52	52	public	ptk	I	"20ccf0ed-1147-4a6e-b8e3-51fa"
53	53	public	ptk	I	"b32393ac-e32a-4cfc-9b12-b96f"
54	54	public	ptk	I	"34ac113c-6d85-43e3-914d-faf8"
55	55	public	ptk	I	"9cc8aeaa-e528-4b4c-9700-222f"
*					

Gambar 5. 41 Isi Tabel sinkbd_outbound *Source*

	id_sink_data [PK] integer	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
1	1	public	ptk	I	"933bf4a9-a099-464a-958e-191e46"
2	2	public	ptk	I	"12b8e09d-070d-4925-8657-705ee2"
3	3	public	ptk	I	"60f82a13-9cd8-49ca-80ac-b6a049"
4	4	public	ptk	I	"b1bb7292-493b-48cd-8ff0-38b646"
5	5	public	ptk	I	"4ca62f82-7b1b-4b7f-954d-01ddb4"
*					

Gambar 5. 42 Isi Tabel sinkbd_inbound *Target* Setelah Sinkronisasi Gagal

	id_sink_data [PK] integer	schema_name character varying(50)	table_name character varying(50)	event character varying(1)	row_data text
47	47	public	ptk	I	"92da596d-d72f-48c9-bd38-343c"
48	48	public	ptk	I	"2a745e9a-0da9-4bd4-ad61-003c"
49	49	public	ptk	I	"1478ad44-01be-4dd7-b910-17fa"
50	50	public	ptk	I	"30104cff-c89e-4e22-8db7-b7a1"
51	51	public	ptk	I	"bae98028-5702-43f1-be88-415c"
52	52	public	ptk	I	"20ccf0ed-1147-4a6e-b8e3-51fa"
53	53	public	ptk	I	"b32393ac-e32a-4cfc-9b12-b96f"
54	54	public	ptk	I	"34ac113c-6d85-43e3-914d-faf8"
55	55	public	ptk	I	"9cc8aeaa-e528-4b4c-9700-222f"
*					

Gambar 5. 43 Isi Tabel sinkbd_inbound *Target* Setelah Melanjutkan Sinkronisasi

5.4 Evaluasi Pengujian

Berdasarkan pengujian fungsionalitas yang telah dilakukan, pengujian memberikan hasil yang sesuai dengan skenario yang

telah direncanakan. Evaluasi pengujian pada masing-masing fungsionalitas dijelaskan sebagai berikut.

1. Pengujian membuat pengaturan baru sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PW-01 yang memberikan informasi bahwa proses membuat pengaturan baru berjalan dengan benar.
2. Pengujian mengubah pengaturan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PW-02 yang memberikan informasi bahwa proses mengubah pengaturan berjalan dengan benar.
3. Pengujian menghapus pengaturan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PW-03 yang memberikan informasi bahwa proses menghapus pengaturan berjalan dengan benar.
4. Pengujian sinkronisasi dengan skenario uji coba 1 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-01 yang memberikan informasi bahwa data yang dikirimkan sudah benar.
5. Pengujian sinkronisasi dengan skenario uji coba 2 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-02 yang memberikan informasi bahwa data yang dikirimkan sudah benar.
6. Pengujian sinkronisasi dengan skenario uji coba 3 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-03 yang memberikan informasi bahwa data yang dikirimkan sudah benar.
7. Pengujian sinkronisasi dengan skenario uji coba 4 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-04 yang memberikan informasi bahwa data yang dikirimkan sudah benar.
8. Pengujian sinkronisasi dengan skenario uji coba 5 tidak sesuai dengan yang diharapkan. Kondisi ini

diperlihatkan pada pengujian PS-05 yang memberikan informasi bahwa data tidak bisa dimasukkan. Pada Gambar 5. 12 pesan *error* yang muncul disebabkan tabel pd memiliki *constraint* *fk_pd_wali_ptk* sebagai *foreign key* untuk tabel ptk. Karena tabel ptk masih kosong, maka data valid tidak bisa dimasukkan karena *foreign key* tidak ada di tabel ptk. Untuk menanggulangi kesalahan ini dapat dilakukan dengan membuat pengaturan sesuai urutan tertentu. Misalkan pada kasus ini tabel ptk harus disinkronisasikan terlebih dahulu, baru kemudian tabel pd disinkronisasikan agar data dapat dimasukkan karena *foreign key* tersedia di tabel ptk.

9. Pengujian sinkronisasi dengan skenario uji coba 6 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-06 yang memberikan informasi bahwa data yang dikirimkan sudah benar.
10. Pengujian sinkronisasi dengan skenario uji coba 7 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-07, PS-08, dan PS-09 yang memberikan informasi bahwa data yang dikirimkan sudah benar. Pada pengaturan khusus ini dilakukan 3 kali tahapan sinkronisasi untuk membuat tiga tabel pada masing-masing basis data memiliki isi yang sama.
11. Pengujian sinkronisasi dengan skenario uji coba 8 sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PS-10 yang memberikan informasi bahwa data yang dikirimkan sudah benar. Data berhasil dikirimkan berdasarkan akhir baris data dinyatakan valid oleh Sinkbd.

DAFTAR PUSTAKA

- [1] Kusrini, Strategi Perancangan Dan Pengelolaan Basis Data, Yogyakarta: Andi, 2007.
- [2] A. Kadir, Konsep & Tuntunan Praktis Basis Data, Yogyakarta: Andi, 2003.
- [3] A. Istiqomah and Abdurrachman, "Sinkronisasi Data," *Praktikum Basis Data Terdistribusi*, pp. 1-2, 2011.
- [4] E. Long, "How SymetricDS Works," SymetricDS, 15 September 2012. [Online]. Available: <http://www.symmetricds.org/docs/how-to/how-symmetricds-works>. [Accessed 24 December 2014].
- [5] P. Mahendra, "Rancang Bangun Sistem Informasi Laboratorium Teknik Elektro Terpadu Universitas Lampung Berbasis Web," 2014. [Online]. Available: <http://digilib.unila.ac.id/2838>. [Accessed 24 December 2014].
- [6] P. Patel, Java Database Programming with JDBC, The Coriolis Group, 1996, p. 499.
- [7] M. Hall, "Accessing Databases with JDBC," in *Accessing Databases with JDBC*, Prentice Hall and Sun Microsystems Press, p. 499.
- [8] S. E. f. B. I. Systems, "Definition Of Application Landscape," Sebis, 21 January 2009 . [Online]. Available: <http://www.matthes.in.tum.de/wikis/system-cartography/application-landscape>. [Accessed 24 December 2014].
- [9] Z. Hakim, "Pengertian Basis Data RDBMS," 6 February 2013. [Online]. Available: <http://www.zainalhakim.web.id/pengertian-basis-data-rdbms.html>. [Accessed 24 December 2014].
- [10] D. Alfiani, "Basis Data Relasional," 2004. [Online].

Available:

<http://diah.staff.gunadarma.ac.id/Downloads/files/4121/BASIS+DATA+RELASIONAL.doc>. [Accessed 24 December 2014].

- [11] R. A, "Pengujian Perangkat Lunak Dengan Menggunakan Metodewhite Box Dan Black Box," *Stimik Himsya Journal*," vol. VIII, no. 1, pp. 3-6, 2012.
- [12] Oracle, "NetBeans IDE Features," NetBeans, [Online]. Available:
<https://netbeans.org/features/ide/database.html>. [Accessed 24 December 2014].
- [13] Tutorialspoint, "Data Integrity Testing," Tutorialspoint, [Online]. Available:
http://www.tutorialspoint.com/software_testing_dictionary/data_and_database_integrity_testing.htm. [Accessed 24 December 2014].

BAB VI

KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan penelitian lebih lanjut.

6.1 Kesimpulan

Dalam proses pengerjaan tugas akhir dari tahap pendahuluan, kajian pustaka, analisis, perancangan, implementasi dan pengujian kerangka kerja Sinkbd diperoleh kesimpulan sebagai berikut.

1. Kerangka kerja Sinkbd dibangun dengan menggunakan kerangka kerja *Spring Web Application* dan bahasa pemrograman Java. Kerangka kerja ini dapat melakukan pertukaran data dari basis data sumber ke basis data tujuan dengan membuat pengaturan terlebih dahulu.
2. Kerangka kerja Sinkbd dapat melakukan sinkronisasi basis data tanpa mengubah struktur basis data sumber dan tujuan dengan menggunakan metode *Change Data Capture* digabungkan dengan pengembangan metode *Routing and Batching*.
3. Kerangka kerja Sinkbd dibangun dengan server terpisah dari kedua basis data yang disinkronisasikan. Sinkbd memiliki antarmuka tersendiri untuk membuat pengaturan.
4. Kerangka kerja Sinkbd dapat melakukan pemeriksaan pada setiap baris data yang dikirimkan dari basis data sumber ke basis data tujuan sehingga setiap kesalahan data dapat terdeteksi.

6.2 Saran

Berikut ini merupakan beberapa saran mengenai pengembangan lebih lanjut kerangka kerja Sinkbd berdasarkan hasil rancangan, implementasi dan uji coba yang telah dilakukan.

1. Antarmuka kerangka kerja Sinkbd masih sangat sederhana sehingga diperlukan pengembangan lebih lanjut.
2. Kerangka kerja ini perlu dilakukan ujicoba pada basis data yang memiliki jumlah data yang sangat besar untuk menguji kebenaran data.
3. Kerangka kerja ini perlu dilakukan ujicoba pada koneksi publik karena sejauh ini ujicoba dilakukan pada koneksi lokal yang dimiliki ITS.
4. Kerangka kerja ini masih memiliki kekurangan pada pencatatan hasil sinkronisasi. Dibutuhkan pengembangan untuk membagi *file log* menjadi 2 untuk mencatat sinkronisasi berhasil dan sinkronisasi gagal.

BIODATA PENULIS



Tommy Nurwantoro, lahir di Lumajang, pada tanggal 18 Juni 1993. Penulis menempuh pendidikan mulai dari SD Negeri 2 Martapura Sumatera Selatan (1999-2005), SMP Negeri 2 Martapura Sumatera Selatan (2005-2006), SMP Negeri 1 Sukodono Lumajang Jawa Timur (2006-2008) SMA Negeri 2 Lumajang Jawa Timur (2008-2011) dan S1 Teknik Informatika ITS (2011-2015).

Selama masa kuliah, penulis aktif dalam organisasi Himpunan Mahasiswa Teknik Computer (HMTTC). Diantaranya adalah menjadi staff departemen hubungan luar himpunan mahasiswa teknik computer ITS 2012-2013, koordinator OC (*Organize Comitte*) kaderisasi 2012, koordinator SC (*Streering Comitte*) kaderisasi 2013, dan staff ahli departemen pengembangan profesi himpunan mahasiswa teknik computer ITS 2013-2014. Penulis juga aktif dalam kegiatan kepanitiaan Schematics. Diantaranya penulis pernah menjadi wakil ketua bagian keamanan dan perizinan Schematics 2012 dan staff NST (National Seminar of Technology) Schematics 2013. Penulis juga aktif di bidang kepanitiaan fakultas. Selama berkuliah penulis juga sering terlibat di dalam *project* pembuatan web.

Selama kuliah di teknik informatika ITS, penulis mengambil rumpun mata kuliah Rekayasa Perangkat Lunak (RPL). Penulis pernah menjadi asisten dosen mata kuliah pemrograman terstruktur. Komunikasi dengan penulis dapat melalui email: **kidyagami@gmail.com**.