

**PENERAPAN API UNTUK SENTRALISASI DATA SISTEM
PEMBUKUAN**

Skripsi



oleh
HENRIKUS KAREL DWIPUTRA
72190327

**PROGRAM STUDI SISTEM INFORMASI FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA**

TAHUN 2023

**PENERAPAN API UNTUK SENTRALISASI DATA SISTEM
PEMBUKUAN**

Skripsi



Diajukan kepada Program Studi Sistem Informasi Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana
Sebagai Salah Satu Syarat dalam Memperoleh Gelar
Sarjana Komputer

Disusun oleh

HENRIKUS KAREL DWIPUTRA
72190327

PROGRAM STUDI SISTEM INFORMASI FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA

TAHUN 2023

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
SKRIPSI/TESIS/DISERTASI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika Universitas Kristen Duta Wacana, saya yang bertanda tangan di bawah ini:

Nama : Henrikus Karel Dwiputra
NIM : 72190327
Program studi : Sistem Informasi
Fakultas : Fakultas Teknologi Informasi
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Kristen Duta Wacana Hak Bebas Royalti Noneksklusif (*None-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul:


**“PENERAPAN API UNTUK SENTRALISASI DATA SISTEM
PEMBUKUAN”**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti/Noneksklusif ini Universitas Kristen Duta Wacana berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama kami sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Yogyakarta
Pada Tanggal : 14 Agustus 2023

Yang menyatakan


Henrikus Karel Dwiputra
NIM.72190327

HALAMAN PENGESAHAN

PENERAPAN WEB API UNTUK SENTRALISASI DATA SISTEM PEMBUKUAN

Oleh: HENRIKUS KAREL DWIPUTRA / 72190327

Dipertahankan di depan Dewan Penguji Skripsi
Program Studi Sistem Informasi Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana - Yogyakarta
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Komputer
pada tanggal
3 Juli 2023

Yogyakarta, 26 Juli 2023
Mengesahkan,

Dewan Penguji:

1. ARGO WIBOWO, ST., MT.
2. UMI PROBOYEKTI, S.Kom., MLIS.
3. BUDI SUSANTO, SKom., M.T.
4. LUSSY ERNAWATI, S.Kom, M.Acc



Dekan

(RESTYANDITO, S.Kom., MSIS., Ph.D)

Ketua Program Studi

(Drs. JONG JEK SIANG, M.Sc.)

HALAMAN PERSETUJUAN

Judul Skripsi : Penerapan Web API untuk Sentralisasi Data Sistem
Pembukuan
Nama Mahasiswa : HENRIKUS KAREL DWIPUTRA
N I M : 72190327
Matakuliah : Skripsi
Kode : SI4046
Semester : Genap
Tahun Akademik : 2022/2023

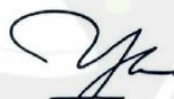
Telah diperiksa dan disetujui di Yogyakarta,
Pada tanggal 26 Juli 2023

Dosen Pembimbing I



ARGO WIBOWO, ST., MT.

Dosen Pembimbing II



UMI PROBOEKTI, S.Kom., MLIS.

DUTA WACANA

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

Penerapan Web API untuk Sentralisasi Data Sistem Pembukuan

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar kesarjanaan saya.

Yogyakarta, 26 Juli 2023




HENRIKUS KAREL DWIPUTRA

72190327

KATA PENGANTAR

Segala puji dan syukur kepada Tuhan yang maha kasih, karena atas segala rahmat, bimbingan, dan bantuan-Nya maka akhirnya Skripsi dengan judul PENERAPAN WEB UNTUK SENTRALISASI DATA SISTEM PEMBUKUAN

ini telah selesai disusun. Penulis memperoleh banyak bantuan dari kerja sama baik secara moral maupun spiritual dalam penulisan Skripsi ini, untuk itu tak lupa penulis ucapkan terima kasih yang sebesar-besarnya kepada:

1. Tuhan yang maha kasih,
2. Orang tua yang selama ini telah mendoakan penulis tanpa kenal lelah
3. Restyandito, S.Kom., MSIS., Ph.D. selaku Dekan FTI UKDW,
4. Drs. Jong Jek Siang, M.Sc. selaku Kaprodi Sistem Informasi,
5. Argo Wibowo, S.T., M.T. . selaku Dosen Pembimbing 1, yang telah memberikan ilmunya dan dengan penuh kesabaran membimbing penulis,
6. Umi Proboyekti, S.Kom., MLIS selaku Dosen Pembimbing 2, yang telah memberikan ilmunya dan dengan penuh kesabaran membimbing penulis,
7. Keluarga yang selalu memberikan motivasi.

Laporan skripsi ini tentunya tidak lepas dari segala kekurangan dan kelemahan, untuk itu segala kritikan dan saran yang bersifat membangun guna kesempurnaan skripsi ini sangat diharapkan. Semoga proposal/skripsi ini dapat bermanfaat bagi pembaca semua dan lebih khusus lagi bagi pengembangan ilmu komputer dan teknologi informasi.

Yogyakarta, 14 Agustus 2023

Henrikus Karel Dwiputra

DAFTAR ISI

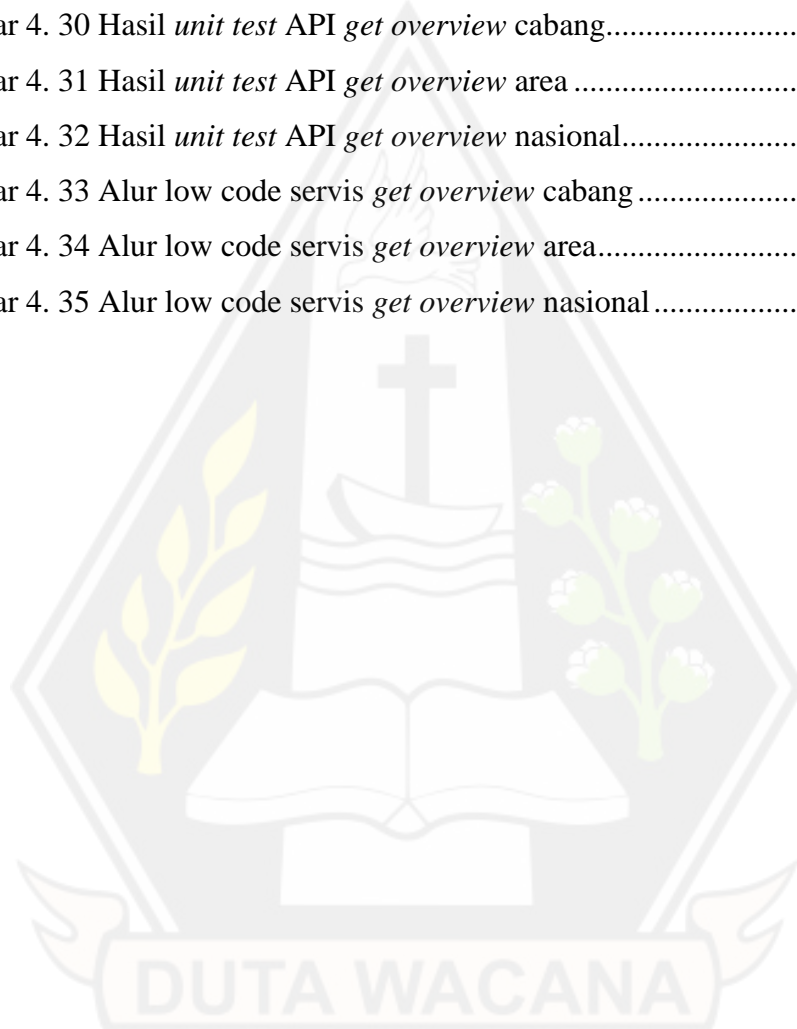
SAMPUL.....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PERSETUJUAN.....	iii
PERNYATAAN KEASLIAN SKRIPSI.....	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL.....	x
DAFTAR <i>CODE</i>	xii
DAFTAR SKENARIO	xiii
ABSTRAK.....	xiv
ABSTRACT.....	xv
BAB 1	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Spesifikasi Sistem	3
1.5. Tujuan Penelitian	4
1.6. Metodologi Penelitian	4
1.7. Sistematika Penulisan	4
BAB 2	6
2.1. Tinjauan Pustaka	6
2.2. Service Oriented Architecture(SOA)	7
2.3. Web Service	8

2.4.	<i>Representational State Transfer (REST)</i>	10
2.5.	<i>Application Programming Interface (API)</i>	10
2.6.	PL/SQL	11
2.7.	Personal Extreme Programing (PXP)	12
BAB 3	15
3.1.	<i>Requirements</i>	15
3.2.	<i>Planning</i>	17
3.3.	<i>Iteration Initialization</i>	18
3.4.	<i>Design</i>	18
BAB 4	49
4.1.	<i>Implementation</i>	49
4.2.	<i>Software Testing</i>	98
4.3.	Spesifikasi API	114
4.4.	Kelebihan dan Kekurang API	114
BAB 5	115
5.1.	Kesimpulan	115
5.2.	Saran	115
DAFTAR PUSTAKA	116
LAMPIRAN A	119
LAMPIRAN B	190

DAFTAR GAMBAR

Gambar 3. 1 <i>Flowchart</i> alur prosedur <i>login</i>	23
Gambar 3. 2 <i>Flowchart</i> alur prosedur <i>check token</i>	24
Gambar 3. 3 <i>Flowchart</i> alur prosedur <i>logout</i>	25
Gambar 3. 4 <i>Flowchart</i> alur prosedur <i>create</i> transaksi.....	33
Gambar 3. 5 <i>Flowchart</i> alur <i>update</i> cabang transaksi.....	34
Gambar 3. 6 <i>Flowchart</i> alur prosedur <i>delete</i> transaksi	35
Gambar 3. 7 <i>Flowchart</i> alur prosedur <i>authorize</i> transaksi.....	41
Gambar 3. 8 <i>Flowchart</i> alur prosedur <i>reanalyze</i> transaksi	43
Gambar 4. 1 Hasil implementasi servis <i>login</i>	53
Gambar 4. 2 Hasil implementasi servis <i>logout</i>	54
Gambar 4. 3 Hasil <i>unit test</i> API <i>login</i>	54
Gambar 4. 4 Hasil <i>unit test</i> API <i>logout</i>	55
Gambar 4. 5 Alur <i>low code</i> servis <i>login</i>	65
Gambar 4. 6 Alur <i>low code</i> servis <i>logout</i>	65
Gambar 4. 7 Hasil implmentasi servis <i>get all</i> Transaksi.....	68
Gambar 4. 8 Hasil implementasi servis <i>create</i> Transaksi	69
Gambar 4. 9 Hasil implementasi servis <i>update</i> Transaksi	69
Gambar 4. 10 Hasil implementasi servis <i>delete</i> Transaksi	70
Gambar 4. 11 Hasil <i>unit test</i> API <i>create</i> Transaksi.....	70
Gambar 4. 12 Hasil <i>unit test</i> API <i>get all</i> Transaksi	71
Gambar 4. 13 Hasil <i>unit test</i> API <i>update</i> Transaksi.....	71
Gambar 4. 14 Hasil <i>unit test</i> API <i>delete</i> Transaksi	72
Gambar 4. 15 Alur <i>low code</i> servis <i>create</i> transaksi.....	78
Gambar 4. 16 Alur <i>low code</i> servis <i>get all</i> transaksi.....	78
Gambar 4. 17 Alur <i>low code</i> servis <i>update</i> transaksi.....	79
Gambar 4. 18 Alur <i>low code</i> servis <i>delete</i> transaksi	79
Gambar 4. 19 Hasil implementasi servis <i>get by status</i> transaksi.....	81
Gambar 4. 20 Hasil implementasi servis <i>authorize</i> transaksi	82
Gambar 4. 21 Hasil implementasi servis <i>reanalyze</i> transaksi.....	82
Gambar 4. 22 Hasil <i>unit test</i> API Get Transaksi by Status and Cabang Transaksi83	

Gambar 4. 23 Hasil <i>unit test</i> API <i>authorize</i> Transaksi	83
Gambar 4. 24 Hasil <i>unit test</i> API <i>reanalyze</i> Transaksi	84
Gambar 4. 25 Alur <i>low code</i> servis <i>get by status</i> transaksi.....	90
Gambar 4. 26 Alur <i>low code</i> servis <i>authorize</i> transaksi.....	90
Gambar 4. 27 Hasil implementasi servis <i>get overview</i> cabang.....	92
Gambar 4. 28 Hasil implementasi servis <i>get overview</i> area.....	93
Gambar 4. 29 Hasil implementasi servis <i>get overview</i> nasional.....	93
Gambar 4. 30 Hasil <i>unit test</i> API <i>get overview</i> cabang.....	94
Gambar 4. 31 Hasil <i>unit test</i> API <i>get overview</i> area	94
Gambar 4. 32 Hasil <i>unit test</i> API <i>get overview</i> nasional.....	95
Gambar 4. 33 Alur <i>low code</i> servis <i>get overview</i> cabang	97
Gambar 4. 34 Alur <i>low code</i> servis <i>get overview</i> area.....	97
Gambar 4. 35 Alur <i>low code</i> servis <i>get overview</i> nasional	97



DAFTAR TABEL

Tabel 3. 1 Tabel <i>User Story</i>	16
Tabel 3. 2 Tabel <i>task</i>	16
Tabel 3. 3 Hasil Perencanaan Iterasi	17
Tabel 3. 4 Entitas Area.....	19
Tabel 3. 5 Entitas Jabatan.....	19
Tabel 3. 6 Entitas Level	20
Tabel 3. 7 Entitas Pengguna.....	20
Tabel 3. 8 Entitas Login	21
Tabel 3. 9 Analisis dan perancangan API <i>Login</i>	26
Tabel 3. 10 Rancangan dan analisis API <i>Logout</i>	27
Tabel 3. 11 Data Transaksi Departemen Remedial	28
Tabel 3. 12 Tabel Atribut Data Transaksi	28
Tabel 3. 13 Entitas Transaksi	30
Tabel 3. 14 Entitas Tipe transaksi	32
Tabel 3. 15 Rancangan dan analisis API <i>Create Transaction</i>	35
Tabel 3. 16 Rancangan dan analisis API <i>Read Transaction All</i>	37
Tabel 3. 17 Rancangan dan analisis API <i>Update Transaction</i>	39
Tabel 3. 18 Rancangan dan analisis API <i>Delete Transaction</i>	40
Tabel 3. 19 Rancangan dan analisis API <i>Authorize Transaction</i>	43
Tabel 3. 20 Rancangan dan analisis API <i>Reanalyze Transaction</i>	44
Tabel 3. 21 Entitas Overview Cabang.....	45
Tabel 3. 22 Entitas Overview Area	45
Tabel 3. 23 Rancangan dan analisis API <i>Read Overview 1</i>	46
Tabel 3. 24 Rancangan dan analisis API <i>Read Overview 2</i>	47
Tabel 3. 25 Rancangan dan analisis API <i>Read Overview 3</i>	48
Tabel 4. 1 Hasil implementasi tabel Area.....	49
Tabel 4. 2 Hasil implementasi tabel Jabatan.....	50
Tabel 4. 3 Hasil implementasi tabel Level.....	50
Tabel 4. 4 Hasil implementasi dari tabel pengguna	51
Tabel 4. 5 Hasil implementasi tabel <i>Login</i>	51

Tabel 4. 6 Hasil implementasi tabel transaksi.....	66
Tabel 4. 7 Hasil implementasi tabel tipe transaksi.....	66
Tabel 4. 8 Hasil implementasi tabel <i>overview</i> cabang	91
Tabel 4. 9 Hasil implementasi tabel <i>overview</i> area.....	92
Tabel 4. 10 Hasil pengujian fungsionalitas pada iterasi-01	98
Tabel 4. 11 Hasil pengujian fungsionalitas pada iterasi-02	100
Tabel 4. 12 Hasil pengujian fungsionalitas pada iterasi-03	106
Tabel 4. 13 Hasil pengujian fungsionalitas pada iterasi-03	111



DAFTAR CODE

Code 4. 1 Hasil implementasi dari prosedur <i>login</i>	52
Code 4. 2 Hasil implementasi prosedur <i>check token</i>	52
Code 4. 3 Hasil implementasi prosedur <i>logout</i>	53
Code 4. 4 Dokumentasi <i>source code</i> pembuatan tabel Pengguna.....	55
Code 4. 5 Dokumentasi <i>source code</i> pembuatan tabel Area.....	56
Code 4. 6 Dokumentasi <i>source code</i> pembuatan tabel Jabatan	57
Code 4. 7 Dokumentasi <i>source code</i> pembuatan tabel Level	58
Code 4. 8 Dokumentasi <i>source code</i> pembuatan tabel <i>Login</i>	58
Code 4. 9 Dokumentasi <i>source code</i> pembuatan prosedur <i>Login</i>	59
Code 4. 10 Dokumentasi <i>source code</i> pembuatan prosedur <i>Check Token</i>	62
Code 4. 11Dokumentasi <i>source code</i> pembuatan prosedur <i>Logout</i>	64
Code 4. 12 Hasil implementasi prosedur <i>create</i> transaksi	67
Code 4. 13 Hasil implementasi prosedur <i>update</i> transaksi	67
Code 4. 14 Hasil implementasi prosedur <i>delete</i> transaksi.....	68
Code 4. 15 Dokumentasi <i>source code</i> pembuatan tabel Transaksi	72
Code 4. 16 Dokumentasi <i>source code</i> pembuatan prosedur <i>create</i> Transaksi.....	73
Code 4. 17 Dokumentasi <i>source code</i> pembuatan prosedur <i>update</i> Transaksi.....	76
Code 4. 18 Dokumentasi <i>source code</i> pembuatan prosedur <i>delete</i> Transaksi	77
Code 4. 19 Hasil implementasi prosedur <i>authorize</i> transaksi	80
Code 4. 20 Hasil implementasi prosedur <i>reanalyze</i> transaksi.....	81
Code 4. 21 Dokumentasi <i>source code</i> pembuatan prosedur <i>authorize</i> Transaksi. 84	
Code 4. 22 Dokumentasi <i>source code</i> pembuatan prosedur <i>reanalyze</i> Transaksi 87	
Code 4. 23 Dokumentasi <i>source code</i> pembuatan tabel <i>overview</i> cabang	95
Code 4. 24 Dokumentasi <i>source code</i> pembuatan tabel <i>overview</i> area.....	96

DAFTAR SKENARIO

Skenario 3. 1 Skenario pengguna..... 15



ABSTRAK

PENERAPAN WEB UNTUK SENTRALISASI DATA SISTEM PEMBUKUAN

Oleh

HENRIKUS KAREL DWIPUTRA

72190327

Usaha finansial merupakan usaha yang berfokus pada bidang keuangan. Salah satu jenis usaha finansial yaitu penyedia jasa peminjaman dana berupa uang atau barang. Departemen Remedial merupakan departemen yang ada pada perusahaan yang bergerak di bidang keuangan dan memiliki tugas untuk menandai, mengevaluasi, dan melakukan pemulihan terhadap pinjaman-pinjaman yang telah dikeluarkan perusahaan. Dalam melakukan tugasnya Departemen Remedial sudah memiliki sistem yang bernama Harbag, namun sistem ini dinilai tidak efektif karena hanya berisi HTML dan tidak memiliki database dan arsitektur. Sehingga sistem tersebut hanya dapat melakukan input dan output saja. Oleh karena itu Departemen Remedial harus melakukan beberapa kali input data.

Penelitian ini menawarkan solusi berupa pembuatan API yang berguna untuk pembuatan sistem dalam membantu pekerjaan Departemen Remedial. Sistem ini terintegrasi dengan web service yang memungkinkan data dapat tersentralisasi. Karena adanya data yang tersentralisasi, Departemen Remedial tidak perlu memasukkan data berulang kali karena data dapat diakses antar cabang. Penelitian ini dikembangkan dengan metode Personal Extreme Programming (PXP) dan memanfaatkan PL/SQL dalam Oracle dan SOA dalam SoftwareAG.

Kata-kata kunci : *web service, sentralisasi data, Personal Extreme Programming, PL/SQL, SOA*

ABSTRACT

WEB API APPLICATION FOR SYSTEM DATA CENTRALIZATION BOOKEEPING

By

HENRIKUS KAREL DWIPUTRA

72190327

Financial business is a business that focuses on the financial sector. One type of financial business is a provider of lending services in the form of money or goods. The Remedial Department is a department in a company engaged in the financial sector and has the task of marking, evaluating and recovering loans that have been issued by the company. In carrying out its duties the Remedial Department already has a system called Harbag, but this system is considered ineffective because it only contains HTML and does not have a database and architecture. So that the system can only do input and output only. Therefore, the Remedial Department has to input data several times.

This research offers a solution in the form of creating an API that is useful for building systems to assist the work of the Remedial Department. This system is integrated with a web service that allows data to be centralized. Due to centralized data, the Remedial Department does not need to enter data repeatedly because data can be accessed between branches. This research was developed using the Personal Extreme Programing (PXP) method and utilizing PL/SQL in Oracle and SOA in SoftwareAG..

Keywords: web service, sentralisasi data, Personal Extreme Programing, PL/SQL, SOA

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Usaha finansial merupakan usaha yang berfokus pada bidang keuangan. Salah satu jenis usaha finansial yaitu penyedia jasa peminjaman dana berupa uang atau barang. Sistem kerja dari penyedia jasa peminjaman tersebut, perusahaan akan meminjamkan dana berupa uang kepada pelanggan yang nantinya pelanggan tersebut harus melakukan pembayaran secara berskala hingga dana pengembalian yang telah diputuskan oleh perusahaan selesai terbayarkan. Ada juga perusahaan penyedia jasa peminjaman uang yang membutuhkan jaminan berupa barang atau surat dari pelanggan. Ketika nominal pengembalian terbayarkan maka barang atau surat tersebut akan dikembalikan kepada pelanggan. Dalam menjalankan bisnis finansial tersebut ada pelanggan-pelanggan yang bermasalah dalam pembayaran seperti jatuh tempo pembayaran tidak ditepati dan tidak ada pembayaran menyebabkan perusahaan penyedia jasa peminjaman keuangan mengalami kerugian dan berpeluang menyebabkan kebangkrutan.

Perusahaan melakukan pencegahan dengan cara membuat departemen-departemen yang dapat membantu masalah kerugian tersebut. Salah satunya yaitu departemen Remedial. Departemen Remedial bertugas untuk menandai, mengevaluasi, dan melakukan pemulihan terhadap pinjaman-pinjaman yang telah dikeluarkan perusahaan. Transaksi atau pinjaman yang mengalami kendala tersebut ditandai sebagai transaksi yang merugikan. Transaksi yang ditandai dengan kerugian, dievaluasi agar dari kerugian tersebut perusahaan masih mendapatkan keuntungan. Upaya mendapatkan penghasilan disaat transaksi rugi ini disebut pemulihan.

Departemen Remedial membutuhkan sebuah sistem yang dapat membantu kebutuhan pekerjaan tersebut. Departemen Remedial memiliki sistem bernama Harbag yang beberapa tahun belakang sudah tidak digunakan lagi, karena

keterbatasan fitur dalam sistem. Sistem Harbag belum memiliki arsitektur dan *database*, sehingga sistem hanya dapat membantu pekerjaan pemasukan data transaksi saja. Departemen Remedial harus melakukan perulangan pemasukan data antar cabang, karena pelanggan yang berpindah-pindah kota. Oleh karena itu departemen Remedial mengalami kesulitan karena data antar cabang tidak menjadi satu. Departemen Remedial perlu-sistem baru yang memiliki fitur yang lebih dari pada sistem harbag. Sistem baru ini diharapkan dapat membantu departemen Remedial untuk kebutuhan kinerja dan mengatasi kendala-kendala yang ada pada sistem Harbag.

Penelitian ini membangun sistem baru yang memiliki arsitektur dan database untuk menggantikan sistem Harbag. Fungsi database dan arsitektur ini memungkinkan pengguna untuk memiliki sebuah database yang tersentralisasi dengan tujuan untuk mencegah terjadinya pemasukan data ganda antar cabang. Sistem menggunakan arsitektur RESTful Web API yang memungkinkan pertukaran data antar sistem menjadi lebih aman karena pengguna tidak langsung mengakses database tetapi mengakses API yang terhubung dengan database. Sistem baru yang dibuat merupakan sistem internal departemen Remedial yang bernama *Recovery, Recognition, Registration (RRR)*.

1.2. Rumusan Masalah

Departemen Remedial mengalami kendala:

1. Sistem lama menghasilkan keluaran berupa file Excel karena tidak didukung basis data. Hal ini menyebabkan perulangan input data saat terjadi kesalahan.

Kendala-kendala di atas menyebabkan departemen Remedial membutuhkan sistem baru untuk peningkatan kinerja.

1.3. Batasan Masalah

Batasan masalah dalam penelitian ini adalah:

1. Penelitian meliputi pembuatan sistem dari sisi *backend* pada perusahaan Bina Rancang Wasisa Berjalan yang digunakan hanya untuk keperluan departemen Remedial.
2. Sistem yang dibangun merupakan aplikasi berbasis *website application*.
3. Struktur *database* dari sistem yang dibangun diambil sebagian dari perusahaan, selebihnya struktur dirancang melalui diskusi dengan pengguna atau yang berkepentingan.
4. Sistem memiliki 3 level pengguna terdiri dari level 1 dengan status tertinggi hingga level 3 dengan status terendah. Masing-masing level pengguna memiliki hak akses sistem yang berbeda beda.
5. Penelitian dikatakan selesai apabila sistem telah berhasil melakukan tugas-tugas sesuai dengan kebutuhan pengguna.

1.4. Spesifikasi Sistem

Spesifikasi dari sistem yang dibangun memiliki kemampuan:

1. Memfasilitasi semua pengguna mengakses halaman dashboard yang berisi data statistik dari transaksi yang telah mencapai di tahap *registration* dari setiap cabang, provinsi, dan nasional
2. Mengelola data perusahaan yang berhubungan dengan departemen Remedial oleh *Head Office*
3. Mendukung *Coordinator* cabang untuk mengakses halaman *Recognition*
4. Mendukung *Officer* cabang untuk mengakses halaman *Registration*
5. Mempermudah proses pemindahan transaksi dari *Recovery* ke *Recognition* untuk setiap pengguna yang dapat mengakses halaman *Recovery*
6. Menyediakan proses pemindahan transaksi dari *Recognition* ke *Registration* untuk setiap pengguna yang dapat mengakses halaman *Recognition*
7. Memfasilitasi setiap pengguna untuk melakukan simpan data transaksi di setiap halaman, *Recovery*, *Recognition*, *Registration*.

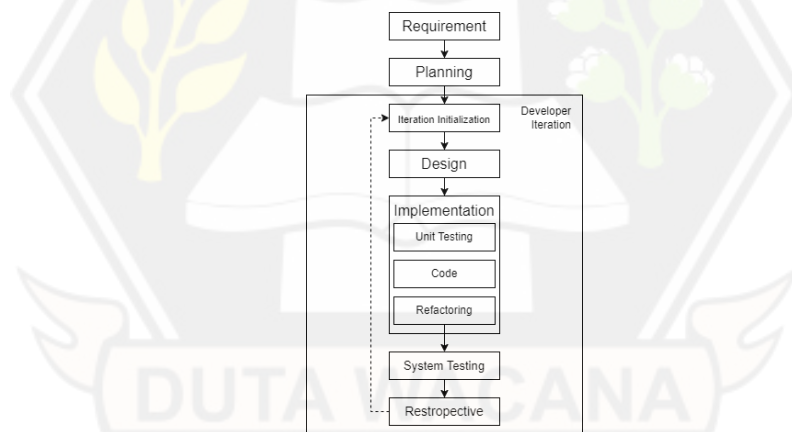
1.5. Tujuan Penelitian

Tujuan dari penelitian ini menghasilkan API dan basisdata untuk mendukung kinerja Departemen Remedial dalam menandai, mengevaluasi, dan melakukan pemulihan terhadap pinjaman-pinjaman yang terhambat atau mengalami kerugian.

1.6. Metodologi Penelitian

Penelitian ini menggunakan pendekatan metode *Personal Extreme Programming* (XP). Tahapan dalam metode ini adalah sebagai berikut:

1. *Requirements*
2. *Planning*
3. *Iteration Initialization*
4. *Design*
5. *Implementation*
6. *System Testing*
7. *Retrospective*



Gambar 1. 1 Alur metodologi *Personal Extreme Programming* (XP)

1.7. Sistematika Penulisan

Laporan skripsi disusun menggunakan sistematika yang terdiri dari 5 bab: Bab 1 berisi tentang pendahuluan yang terdiri dari latar belakang, perumusan masalah, batasan masalah, spesifikasi sistem, tujuan penelitian, metodologi penelitian dan

sistematika penulisan dari penelitian. Bab 2 berisi tentang tinjauan pustaka dan landasan teori yang mendukung kegiatan dan penulisan penelitian. Bab 3 berisi analisis-analisis dari penelitian mencakupi analisis data, rancangan sistem, rancangan hasil sistem, dan rancangan pengujian.

Bab 4 berisi hasil dari pengerjaan penelitian ini mencakup gambar dan table dari hasil penelitian. Bab 5 berisi penutup yang terdiri dari kesimpulan dan saran atas penelitian yang dikerjakan guna menjawab sistem yang dibangun berhasil mengatasi hal-hal yang ada pada rumusan masalah.



BAB 5

KESIMPULAN DAN SARAN

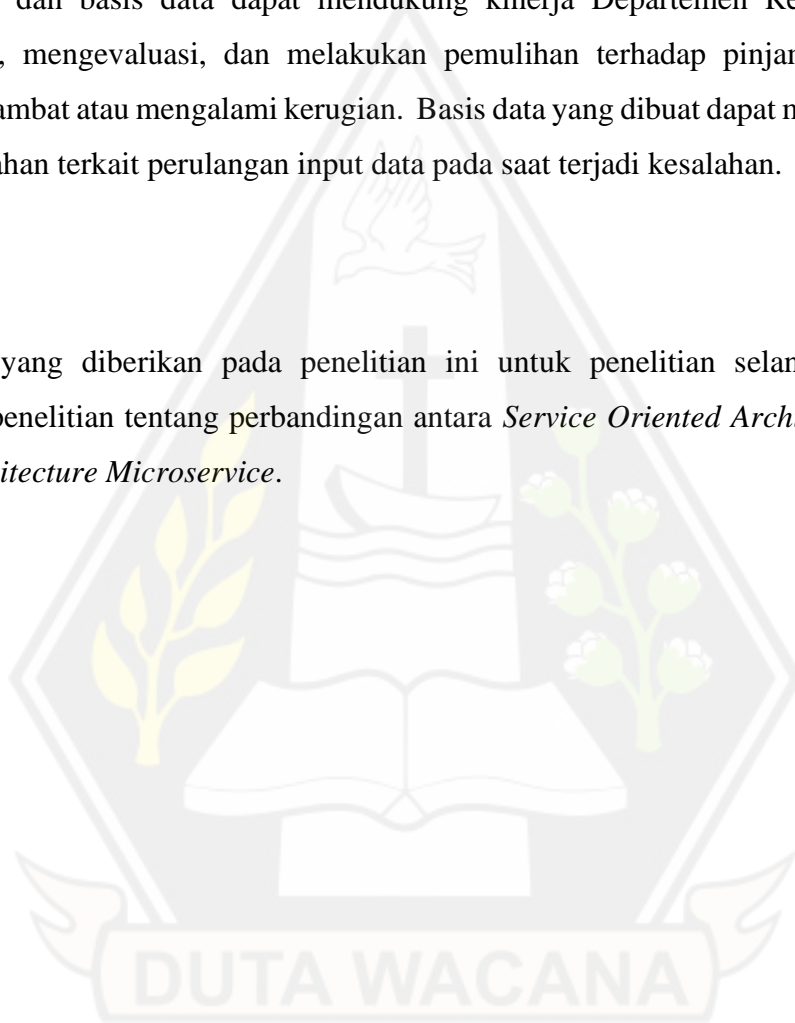
5.1. Kesimpulan

Hasil dari penelitian ini memiliki beberapa kesimpulan sebagai berikut:

1. Web API dan basis data dapat mendukung kinerja Departemen Remedial untuk menandai, mengevaluasi, dan melakukan pemulihan terhadap pinjaman-pinjaman yang terhambat atau mengalami kerugian. Basis data yang dibuat dapat menyelesaikan permasalahan terkait perulangan input data pada saat terjadi kesalahan.

5.2. Saran

Saran yang diberikan pada penelitian ini untuk penelitian selanjutnya untuk melakukan penelitian tentang perbandingan antara *Service Oriented Architectur* (SOA) dengan *Architecture Microservice*.



DAFTAR PUSTAKA

- Abbas, C. J. (2016). REKAYASA ARSITEKTUR SISTEM INFORMASI PELAYANAN PERIJINAN SATU ATAP DI KABUPATEN KUNINGAN DENGAN MENGGUNAKAN METODA SOA (SERVICE ORIENTED ARCHITECTURE). *JEJARING: Jurnal Teknologi dan Manajemen Informatika*, 15-21.
- Arianto, M. A., Munir, S., & Khotimah, K. (2016). ANALISIS DAN PERANCANGAN REPRESENTATIONAL STATE TRANSFER (REST) WEB SERVICE SISTEM INFORMASI AKADEMIK STT TERPADU NURUL FIKRI MENGGUNAKAN YII FRAMEWORK. *Jurnal Teknologi Terpadu*.
- Asiz, N. (2022). *Basis Data Lanjutan dengan PLSQL*. Bandung: WIDINA BHAKTI PERSADA BANDUNG.
- Baharuddin, Wakkang, H., & Irianto, B. (2022). IMPLEMENTASI WEB SERVICE DENGAN METODE REST API UNTUK INTEGRASI DATA COVID 19 DI SULAWESI SELATAN. *JURNAL SINTAKS LOGIKA*, 236-241.
- Barus, A. C., Harunguan, J., & Manulu, E. (2021). PENGUJIAN API WEBSITE UNTUK PERBAIKAN PERFORMANSI APLIKASI DITENUN. *Journal of Applied Technology and Informatics*, 14-21.
- Choirudin, R., & Adil, A. (2019). IMPLEMENTASI REST API WEB SERVICE DALAM MEMBANGUN APLIKASI MULTI PLATFORM UNTUK USAHA JASA. *Jurnal MATRIK*, 284-293.
- Filiana, A., Rini, M. N., Prabawati, A. G., & Samat, R. A. (2022). PENGEMBANGAN REST API UNTUK INFORMASI PASAR TRADISIONAL DI KOTA

YOGYAKARTA DENGAN METODE INCREMENTAL. *SINTECH JOURNAL*, 10-23.

Hadji, S., Taufik, M., & Mulyono, S. (2019). IMPLEMENTASI METODE SCRUM PADA PENGEMBANGAN APLIKASI DELIVERY ORDER BERBASIS WEBSITE (STUDI KASUS PADA RUMAH MAKAN LOMBOK IDJO SEMARANG). *KONFERENSI ILMIAH MAHASISWA UNISSULA (KIMU) 2*, 32-43.

Hantana, J. S. (2013). Pendekatan Service Oriented Architecture (SOA) Pada Pelaksanaan E-Government di Kementerian Hukum dan HAM RI. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, 254-260.

Junaedi, I. (2020). PENGEMBANGAN APLIKASI ANALISIS VENDOR PERFORMANCE DI PT. SUZUKI INDOMOBIL. *JISAMAR (Journal of Information System, Applied, Management, Accounting and Research)*, 37-55.

Kurniawan, Y. K., Oslan, Y., & Kristanto, H. (2013). IMPLEMENTASI REST - API UNTUK PORTAL AKADEMIK UKDWBERBASIS ANDROID. *Jurnal EKSISJurnal EKSIS*, 29-40.

Lutfiani, N., Harahap, E. P., Aini, Q., Ahmad, A. D., & Rahardja, U. (2020). Inovasi Manajemen Proyek I-Learning Menggunakan Metode Agile Scrumban. *InfoTekJar : Jurnal Nasional Informatika dan .*

Muri, M. F., Utomo, H. S., & Sayyidati, R. (2019). Search Engine Get Application Programming Interface. *Jurnal Sains dan Informatika*, 87-97.

Nurgaida, S., & Satrinia, D. (2021). Automated Functional Testing pada API menggunakan Keyword Driven Framework. *Journal of Informatics and Communications Technology (JICT)*, 1-14.

- Permatasari, D. I., Santoso, B., Ningtias, N., R, M. H., Atika, R., Widad, N., . . . R, A. A. (2019). PENGUKURAN THROUGHPUT LOAD TESTING MENGGUNAKAN TESTCASE SAMPLING GORILLA TESTING. *Seminar Nasional Sistem Informasi*, 2008-2014.
- Pranata, B. A., Hijriani, A., & Junaidi, A. (2018). PERANCANGAN APPLICATION PROGRAMMING INTERFACE (API) BERBASIS WEB MENGGUNAKAN GAYA ARSITEKTUR REPRESENTATIONAL STATE TRANSFER (REST) UNTUK PENGEMBANGAN SISTEM INFORMASI ADMINISTRASI PASIEN KLINIK PERAWATAN KULIT. *Jurnal Komputasi*, 33-42.
- Pratama, S., Ibrahim, S., & Reybaharsyah, M. A. (2022). Penggunaan Metode Scrum Dalam Membentuk Sistem Informasi Penyimpanan Gudang Berbasis Web. *INFORMATIKA DAN TEKNOLOGI (INTECH)*, 27-35.
- Rizal, R., & Rahmatulloh, A. (2019). RESTful Web Service untuk Integrasi Sistem Akademik dan Perpustakaan Universitas Perjuangan. *Jurnal Ilmiah Informatika*, 54-59.
- Yanti, S. N., & Rihyanti, E. (2021). Penerapan Rest API untuk Sistem Informasi Film Secara Daring. *Jurnal Informatika Universitas Pamulang*, 195-201.