



# Pengimplementasian Fitur Data Profiling Pada Aplikasi Data Governance Berbasis Open Source Tools dengan Metode Iterative/Incremental

Ekky Chandra Wibowo\*, Tien Fabrianti Kusumasari, Ekky Novrizal Alam

Fakultas Rekayasa Industri, Sistem Informasi, Telkom University, Bandung  
Jl. Telekomunikasi. 1, Terusan Buahbatu - Bojongsoang, Telkom University, Sukapura, Kec. Dayeuhkolot, Kabupaten Bandung, Jawa Barat, Indonesia

Email: <sup>1,\*</sup>ekkychandrawibowo@student.telkomuniversity.ac.id, <sup>2</sup>tienkusumasari@telkomuniversity.ac.id, <sup>3</sup>ekkyNovrizal@telkomuniversity.ac.id

Email Penulis Korespondensi: ekkychandrawibowo@student.telkomuniversity.ac.id

Submitted: 29/09/2022; Accepted: 17/10/2022; Published: 31/10/2022

**Abstrak**—Data merupakan salah satu elemen yang dianggap penting pada suatu bisnis. Jika pengolahan dilakukan dengan baik dan sesuai kebutuhan maka data tersebut memiliki kualitas yang tinggi. Kualitas data yang tinggi dapat memberikan suatu dampak yang positif yaitu dapat menciptakan suatu informasi dan pengetahuan dalam mendukung pengambilan keputusan. Faktanya banyak organisasi yang melakukan penumpukan suatu data tanpa adanya suatu sistem dalam penyimpanan atau pengelolaan data yang baik dan menyebabkan suatu data menjadi besar sehingga sulit untuk dipahami dan menghasilkan informasi di dalamnya menjadi tidak dapat tersampaikan dengan baik yang membuat suatu keputusan tidak memberikan dampak positif yang maksimal. Data Quality Management (DQM) dinilai dapat membantu dalam menjaga suatu kualitas pada data dan memastikan tidak ada data yang hilang maupun rusak. Data yang berkualitas didapatkan dengan salah satu tahapan dalam DQM yaitu data profiling, yang melakukan proses pengecekan data yang tersedia dan tersimpan untuk menentukan suatu isi struktur dan kualitas data. Penelitian berfokus untuk implementasi fitur data profiling modul DQM pada aplikasi website Data Governance. Penelitian ini menghasilkan fitur data profiling yang dapat membantu orang-orang dalam melakukan pengolahan data dimana modul yang dihasilkan adalah value distribution, data completeness, clustering, value similarity dan data deduplication.

**Kata Kunci:** Data Governance; Data Quality; Data Quality Management; Profiling; Website

**Abstract**—Data is one element that is considered important in a business. If the processing is carried out correctly and as needed, the data has high quality. High data quality can have a positive impact, that is, it can create information and knowledge to support decision-making. In fact, many organizations accumulate data without a good data storage or management system and cause the data to become large so that it is difficult to understand and produce information in it that cannot be conveyed properly which makes a decision not have a maximum positive impact. Data Quality Management (DQM) is considered to be able to assist in maintaining quality in the data and ensure that no data is lost or damaged. Quality data is obtained by one of the stages in DQM, namely data profiling, which performs the process of checking the available and stored data to determine a content structure and data quality. The research focuses on implementing the DQM module data profiling feature on the Data Governance website application. This research produces data profiling features that can help people in data processing where the modules produced are value distribution, data completeness, clustering, value similarity and data deduplication.

**Keywords:** Data Governance; Data Quality; Data Quality Management; Profiling; Website

## 1. PENDAHULUAN

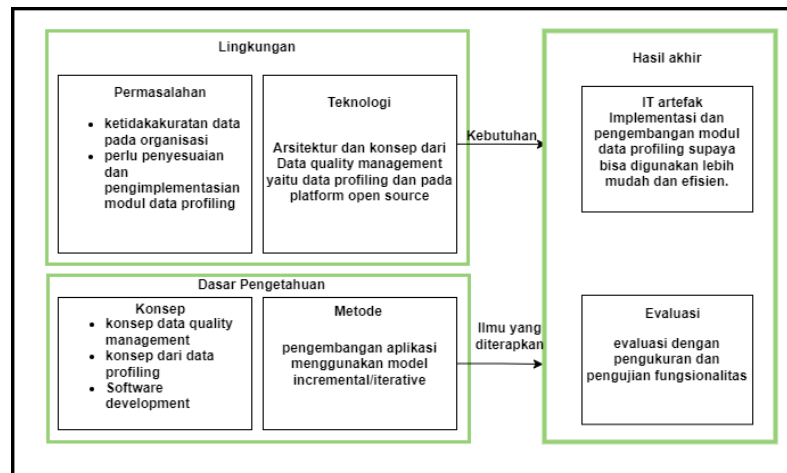
Data dinilai merupakan salah satu komponen terpenting yang ada dalam organisasi atau perusahaan sebagai suatu aset yang vital. Data maupun informasi dapat memberikan suatu pengetahuan yang menjelaskan mengenai pelanggan, produk dan layanan yang dimiliki [1]. Banyak organisasi memanfaatkan data dalam berinovasi dan membuat strategi bisnis yang baik dan efektif hal tersebut karena saat ini data dibutuhkan di seluruh lini area bisnis, proses, dan *system* yang berjalan. Kualitas data yang tinggi dikarenakan data tersebut diolah dengan baik dan sesuai dengan kebutuhan. Data yang memiliki kualitas yang baik dan tinggi dapat memberikan dampak positif dalam suatu informasi dan pengetahuan yang membantu suatu pengambilan keputusan dengan baik [2]. *Data quality* dapat diartikan sebuah ukuran yang ada pada data, data yang memiliki kualitas tinggi dinilai memiliki karakter lengkap, terbaru, aman dan dapat diakses kapan saja [1]. Data yang memiliki kualitas tinggi biasanya disesuaikan dengan tujuan dan strategi bisnis suatu organisasi, hal tersebut karena kebutuhan data setiap organisasi berbeda-beda [1]. Hal tersebut menjadi tantangan dalam menyiapkan, mengelola, dan memiliki data kualitas tinggi karena harus memahami kebutuhan data dari pengguna data. Banyak organisasi yang menumpuk data tanpa adanya suatu sistem penyimpanan dan pengelolaan yang baik yang menyebabkan ukuran data menjadi besar dan sulit untuk dipahami, sehingga informasi tidak dapat tersampaikan dengan baik dan membuat keputusan yang diambil tidak memberikan dampak positif.

Data survei yang dilakukan oleh *Experian Data Quality* menunjukkan lebih dari 90% organisasi masih melakukan kesalahan data kontak penyebab dari hal tersebut karena masih digunakan cara manual dalam melakukan pengecekan data setelah data tersebut dikumpulkan [3]. Jika data memiliki kualitas yang tidak layak maka dapat membuat tingkat produktivitas dan keberhasilan suatu bisnis menurun. Pembangunan infrastruktur dan sistem pengolahan data yang terstruktur dan terintegrasi dapat mempengaruhi kualitas data yang tinggi. Untuk

mendapatkan data berkualitas maka perlu memastikan data tersebut memiliki kualitas tinggi, *Data Quality Management* (DQM) dinilai dapat membantu menjaga suatu kualitas data yang memastikan data tidak hilang maupun rusak dengan adanya aturan, proses, dan sistem [1]. Proses dalam DQM untuk memastikan data memiliki kualitas yang tinggi yaitu sesuai standar karakteristik akurat, lengkap, konsisten, terintegrasi, terbaru, unik, dan valid [1]. Tahapan pada DQM di antaranya *data profiling*, *data cleansing*, *data quality assessment*, dan *data quality monitoring* [4].

DQM digunakan untuk menyelesaikan suatu masalah data yang umumnya disebabkan oleh perbedaan suatu format pada data yang bersumber dari berbagai sumber sehingga memiliki ukuran data yang semakin besar maka diperlukan penyesuaian dan pembersihan terhadap data tersebut. Hal tersebut dapat membuat data menjadi dapat digunakan dan data memiliki nilai dalam membantu mengambil suatu keputusan [5]. Dalam tahapan DQM untuk mendapatkan data yang berkualitas yaitu dengan *data profiling*. *Data profiling* merupakan suatu kegiatan yang di dalamnya terdapat proses pengecekan suatu data yang tersedia dan tersimpan dengan tujuan untuk menentukan isi struktur dan kualitas suatu data [2]. *Data profiling* dinilai dapat membantu suatu data agar sesuai dengan kebutuhan data dari organisasi, sehingga informasi yang diberikan menjadi maksimal dan membantu dalam pengambilan keputusan [5]. Selanjutnya proses *data cleansing* pada tahapan ini melakukan proses pembersihan suatu data yang membuat data menjadi sesuai dengan kebutuhan dari organisasi seperti pengecekan salah ketik, tanda baca, dan kata hubung sehingga data menjadi beraturan dan data dapat dilakukan pengolahan yang menghasilkan suatu informasi yang dapat membantu pengambilan keputusan [2]. *Data Cleansing* sendiri dapat diartikan sebagai pembersihan data agar mendapatkan yang sesuai dengan format dan kualitasnya baik [6]. Terdapat 5 tahapan dalam *data cleansing* yaitu *data analysis*, *Definition of transformation workflow and mapping rule*, *Verification*, *Transformation*, dan *Backflow of cleaned data* [7]. Pengecekan data atau *data profiling* perlu dilakukan terlebih dahulu untuk melakukan pengecekan persebaran data, pengecekan data yang kosong, pengecekan data yang sama. Dalam penelitian ini akan diimplementasikan yaitu *Value Distribution*, *Data Completeness*, *Clustering*, *Data Similarity* dan *Data Deduplication*.

Penelitian ini akan mengimplementasikan platform open-source yang tujuannya agar aplikasi dapat digunakan secara gratis oleh siapa saja yang memerlukan sehingga permasalahan suatu kualitas data dapat terselesaikan dengan mudah dan gratis Penelitian ini memiliki fokus dalam melakukan suatu implementasi fitur pada *data profiling* yang ada pada modul *data quality management* pada aplikasi *website Data Governance* yang sudah dirancang oleh peneliti sebelumnya. Penelitian ini juga melakukan penyesuaian suatu fitur agar dapat digunakan dengan baik pada platform Linux.



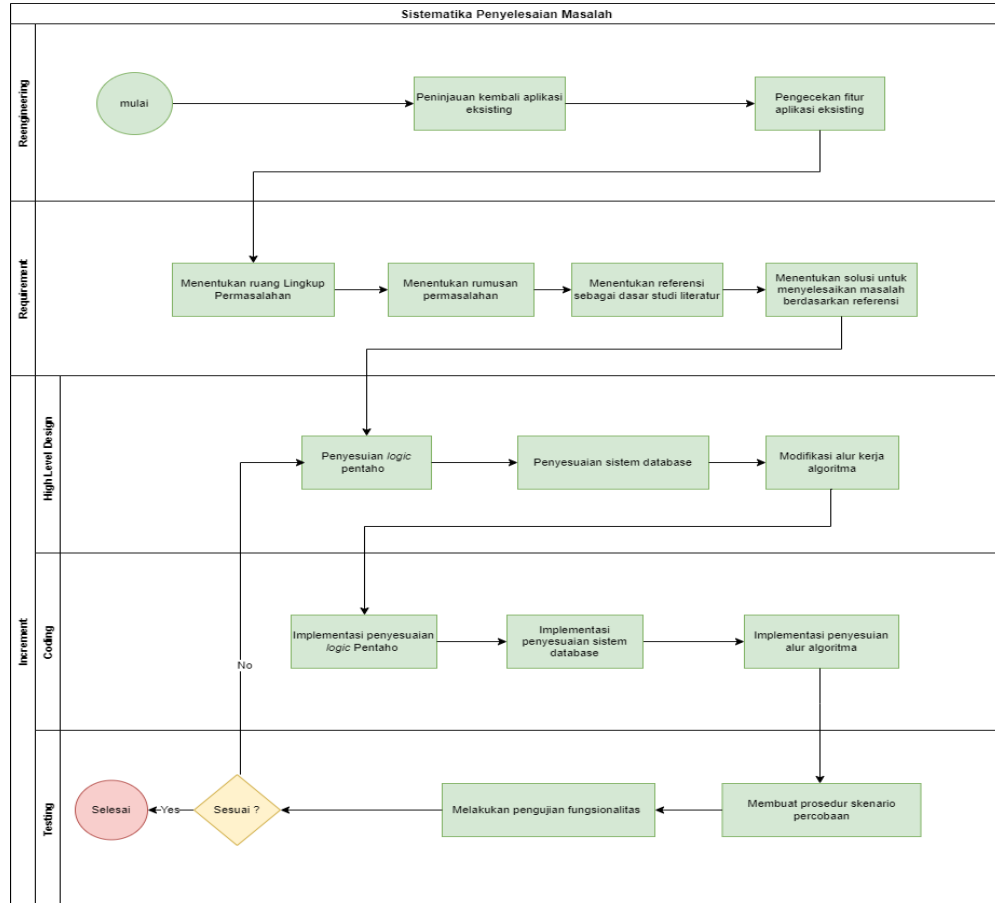
**Gambar 1.** Model Konseptual

Pada gambar 1 menjelaskan mengenai model konseptual yang digunakan yaitu *Desain Science Research* (DSR). DSR bertujuan untuk menghasilkan pengetahuan mengenai bagaimana hal-hal dapat dibangun maupun diatur untuk mencapai tujuan yang diinginkan [8]. Terdapat masalah ketidaksesuaian yang terdapat dalam gambar 1 pada organisasi hal tersebut karena organisasi memiliki banyak data yang menumpuk dan tidak memiliki sistem yang baik dalam manajemen dan penyimpanan data sehingga tidak dapat digunakan dengan mudah dan fleksibel pada modul *data profiling*. Permasalahan tersebut merupakan dasar dari penelitian ini dengan dilakukan pemanfaatan teknologi maupun konsep *Data Quality Management* di mana terdapat 2 modul dalam aplikasi eksisting yang dinilai belum maksimal pada pengimplementasiannya. *Framework Laravel* dan *Pentaho Data Integration* digunakan pada penelitian ini sebagai kerangka kerja yang memiliki output adalah modul *data profiling* yang fleksibel penggunaannya. Penelitian ini menggunakan model *Software Development Incremental/Iterative* karena memiliki kelebihan pada waktu pengembangan yang lebih cepat serta manajemen risiko yang baik. Pada tahap evaluasi penggunaan metode *Blackbox testing* berupa pengujian terhadap aplikasi secara fungsional digunakan pada penelitian ini.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Dalam melakukan penelitian ini peneliti melakukan tahapan ataupun langkah-langkah yang akan dilakukan untuk menyelesaikan permasalahan yang ada agar permasalahan dapat diselesaikan dengan mudah dan efisien.



**Gambar 2.** Tahapan Penelitian

Pada Gambar 2 menjelaskan mengenai tahapan penelitian dengan model yang digunakan yaitu *Incremental/Iterative* yang menggabungkan *model waterfall* yang dilakukan secara *increment* [7].

**Tabel 1.** Teknik dan Output Tahapan Penelitian

No	Tahapan	Teknik	Output
1	<i>Reengineering</i>	<i>Code review</i>	Analisis aplikasi <i>eksisting</i> dan fitur
2	<i>Requirement</i>	<i>Requirements Development</i>	Spesifikasi <i>requirement</i> aplikasi
3	<i>High Level Design</i>	<i>Detailed Design</i>	Usulan modifikasi <i>logic pentaho</i> , <i>Component diagram</i> , arsitektur diagram dan <i>data flow diagram</i>
4	<i>Coding</i>	<i>Coding and debugging</i>	Aplikasi hasil implementasi
5	<i>Testing</i>	<i>Fungsional test</i>	Hasil pengujian fungsional fitur aplikasi

Tabel 1 menjelaskan mengenai teknik serta output dari tahapan pada penelitian ini. Berikut merupakan penjelasan mengenai tahapan pada penelitian ini.

#### 1. Tahapan Reengineering

Tahap ini melakukan pengambilan data yang dilakukan proses pengambilan data dari peneliti sebelumnya. Dimulai dengan menjalankan aplikasi eksisting untuk melakukan improvisasi dalam membuat aplikasi eksisting berjalan efisien serta fleksibel. Code review merupakan teknik yang digunakan, code yang dibuat peneliti sebelumnya dianalisis untuk pengembangan fitur di dalamnya.

#### 2. Tahapan Requirement

Tahap yang menjelaskan apa yang seharusnya dilakukan oleh perangkat lunak, dikenal sebagai requirements development, requirements analysis, analysis, requirements definition, software requirements, specification,

functional spec, dan spec [9]. Proses persiapan dilakukan yaitu analisis untuk mengetahui persyaratan dan informasi mengenai dokumentasi program [10].

3. Tahapan High Level Design

Tahapan yang bertujuan untuk melakukan evaluasi dan menemukan suatu solusi dari hasil analisis yang dilakukan pada tahap sebelumnya. Design sendiri merupakan salah satu tahapan di mana solusi dari pengembangan ditemukan design dinilai baik apabila dapat membantu proses implementasi tahap coding dengan mudah dan terstruktur [9].

4. Tahapan Coding

Tahapan ini menjelaskan proses implementasi dari hasil analisis dari kebutuhan maupun design atau suatu rencana untuk mendapatkan solusi suatu permasalahan pada tahap sebelumnya [10]. Solusi ditulis dalam code untuk menjalankan program yang dapat menyelesaikan masalah dan implementasi usulan suatu penyesuaian logic pada aplikasi open-souce Pentaho yang digunakan sebagai algoritma dalam pengolahan data aplikasi eksisting.

5. Tahapan Testing

Tahap ini menjelaskan mengenai aktivitas dalam melakukan test terhadap program yang telah dibuat pada tahapan sebelumnya. Pada tahap ini metode Blackbox testing digunakan dalam melakukan testing yang meliputi pengujian secara fungsionalitas secara keseluruhan [11].

**2.2 Pengumpulan Data**

Penelitian ini pengumpulan data dilakukan dengan menggunakan metode *software reengineering* karena dilakukan proses pengembangan aplikasi *eksisting* yang sudah ada di mana dilakukan proses perbaikan, peningkatan kinerja dan menyesuaikan arsitektur apabila dibutuhkan [12]. Data didapatkan dari penelitian sebelumnya Dewanti [13] dan Rahmani [14]. Didapatkan juga data berupa *source code* aplikasi *eksisting* yang akan dikembangkan [15].

**2.3 Pengolahan Data**

Pengolahan data dilakukan setelah pengumpulan data selesai, data yang dikumpulkan dilakukan analisis untuk menemukan permasalahan yang ada. Kemudian hasil analisis dijadikan rancangan dalam design penyelesaian masalah. Studi literatur digunakan sebagai acuan dalam mencari solusi yang sesuai dengan permasalahan. Tahap pengembangan menggunakan metode *iterative incremental*. Pengembangan ini menggunakan bahasa pemrograman PHP dan *framework Laravel* dalam mengembangkan website. Penggunaan *database* menggunakan RDBMS MySQL sebagai *database* dan menggunakan *Pentaho Data Integration* sebagai kerangka integrasi data yang sesuai dengan penelitian sebelumnya. Dilakukan evaluasi dengan metode *Blackbox testing*, yang merupakan pengujian perangkat lunak secara fungsionalitas yang sesuai dengan deskripsi dan tujuan sistem dibuat atau dikembangkan [16]. Penelitian ini akan melakukan serangkaian skenario dalam melakukan pengujian terhadap hasil implementasi untuk memastikan hasil implementasi tersebut telah sesuai dengan kebutuhan mampu menyelesaikan masalah yang ada.

**3. HASIL DAN PEMBAHASAN**

**3.1 Reengineering Stage**

**Tabel 2.** Reengineering Stage

<i>Criteria</i>	<i>Description</i>
<i>Input</i>	-
<i>Proses</i>	<i>Running</i> aplikasi dari <i>source code</i> dan <i>logic</i> yang telah dibuat pada aplikasi sebelumnya
<i>Output</i>	Aplikasi eksisting

Berdasarkan tabel 2 pada tahap awal yang dilakukan adalah melakukan *re-running* aplikasi yang telah dibuat dalam aplikasi sebelumnya di mana hal tersebut dilakukan untuk menjadi dasar dalam melakukan analisis dalam tahap ini. Berdasarkan pada analisis yang dilakukan dari peninjauan dari .ktr Pentaho yang telah dibuat sebelumnya ditemukan permasalahan di mana .ktr tidak bisa digunakan atau error ketika diintegrasikan hal tersebut membuat fitur pada aplikasi masih menggunakan .ktr yang belum di tingkatkan performanya sehingga aplikasi belum memiliki performa terbaik.

**Tabel 3.** Analisis Modul profiling Eksisting

<b>Modul</b>	<b>Deskripsi</b>
<i>Single value distribution</i>	Modul ini berfungsi untuk mengidentifikasi nilai text pada suatu kolom.
<i>Column data completeness</i>	Modul ini berfungsi untuk melakukan pengecekan terhadap kelengkapan data.

	Modul	Deskripsi
Multi Column	<i>clustering</i>	Modul ini berfungsi untuk melakukan pengelompokan data.
	<i>value similarity</i>	Modul ini berfungsi untuk mengidentifikasi nilai yang sama pada beberapa kolom yang berbeda.
	<i>data deduplication</i>	Modul ini berfungsi untuk melakukan identifikasi terhadap data yang terduplikat

Berdasarkan tabel 3 menjelaskan mengenai fungsi dari modul *profiling* yang telah dibuat pada penelitian sebelumnya. Dalam penelitian ini .ktr yang telah dibuat dilakukan penyesuaian untuk dapat diimplementasikan dan diintegrasikan pada aplikasi eksisting *Data Governance* pada modul *profiling*.

### 3.2 Requirement Stage

**Tabel 4.** Requirement Stage

Criteria	Description
Input	Aplikasi Eksisting
Proses	Analisis aplikasi eksisting untuk menemukan permasalahan dari aplikasi eksisting dengan tujuan melakukan penyesuaian dan pengembangan.
Output	Analisis aplikasi eksisting

Berdasarkan tabel 4 menjelaskan mengenai yang harus dilakukan pada tahapan ini untuk mendapatkan hasil analisis dari aplikasi eksisting yang telah dilakukan *running* dalam penelitian pada tahapan sebelumnya. Analisis aplikasi eksisting dilakukan analisis pada modul *profiling* yang telah dibuat sebelumnya. Terdapat permasalahan yang ditemukan yaitu terdapat error yang disebabkan tidak terhubungnya file *.ktr logic pentaho* dengan *interface* aplikasi sebagai platform untuk menampilkan hasil dari proses yang dijalankan serta belum terhubungnya sistem untuk menjalankan aplikasi *pentaho* menggunakan *Carte* menggunakan web server yang disediakan oleh *pentaho* dan fungsi yang sudah dibuat penghubungnya oleh Salmawati [17]. Hal tersebut membuat proses pengolahan data masih membutuhkan waktu yang lebih lama.

**Tabel 5.** Fitur Modul DQM pada Aplikasi Eksisting

	Modul	Eksisting	Penambahan	Modifikasi
Profiling	Single Column	<i>value distribution</i>	✓	✓
		<i>data completeness</i>	✓	✓
		<i>Show null</i>	✓	
		<i>clustering</i>	✓	✓
		<i>value similarity</i>	✓	✓
Cleansing	Multi Column	<i>data deduplication</i>	✓	✓
		<i>Format</i>	✓	
		<i>Null Modul</i>	✓	
		<i>Modul (MDM)</i>	✓	
		<i>Deduplication</i>	✓	
		<i>Clustering</i>	✓	

Berdasarkan tabel 5 menjelaskan perlu adanya penyesuaian pada algoritma dari aplikasi eksisting agar dapat mengimplementasikan file .ktr yang telah dibuat dengan menggunakan sistem *Carte.bat* untuk membuat performa aplikasi eksisting meningkat.

### 3.3 Perancangan

#### 3.3.1 High Level Design Stage

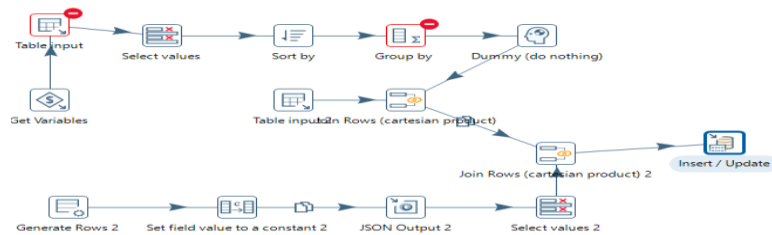
**Tabel 6.** High Level Design Stage

Kriteria	Deskripsi
Input	Analisis aplikasi eksisting
Proses	1. Mengusulkan penyesuaian <i>logic pentaho</i> 2. Mengusulkan penggantian <i>job trigger</i> pan,bat menjadi <i>carte</i> 3. Penyesuaian tabel di dalam database
Output	Rancangan penyesuaian aplikasi Iterasi 1
Tujuan	Membuat usulan rancangan penyesuaian <i>logic</i>

Kriteria	Deskripsi
Proses	<ol style="list-style-type: none"> <li>1. Penyesuaian <i>logic data distribution</i></li> <li>2. Penyesuaian <i>logic data completeness</i></li> <li>3. Penyesuaian <i>logic clustering</i></li> <li>4. Penyesuaian <i>logic value similarity</i></li> <li>5. Penyesuaian <i>logic data deduplication</i></li> </ol>
Output	Usulan penyesuaian <i>logic</i> pentaho Iterasi 2
Tujuan	Membuat usulan rancangan penggantian <i>job trigger</i>
Proses	<ol style="list-style-type: none"> <li>1. Membuat rancangan arsitektur penggantian <i>job trigger</i></li> <li>2. Membuat rancangan penggantian <i>component</i> pada aplikasi</li> </ol>
Output	<ol style="list-style-type: none"> <li>1. Arsitektur diagram</li> <li>2. <i>Component</i> diagram</li> </ol> Iterasi 3
Tujuan	Membuat rancangan aliran data
Proses	Membuat <i>Data Flow Diagram</i>
Output	<i>Data Flow Diagram</i>

Berdasarkan tabel 6 menjelaskan mengenai proses yang akan dilakukan pada tahap perancangan dengan metode *Iterative/Incremental* pada tahapan *high level design*.

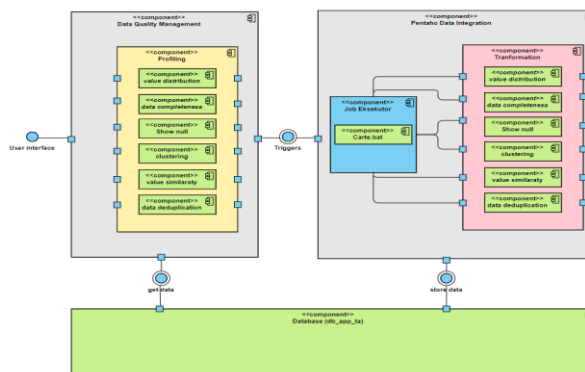
### 3.3.2 Usulan Penyesuaian Logic Pentaho



**Gambar 3.** Logic Pentaho pada aplikasi eksisting modul value distribution

Berdasarkan hasil analisis yang dilakukan ditemukan 2 faktor penyebab aplikasi eksisting error yaitu belum ada pengaturan koneksi *database*, variabel dan parameter data yang akan diolah dengan file *.ktr*. gambar 3 menjelaskan mengenai *Logic Pentaho* pada aplikasi eksisting modul *value distribution*, berdasarkan hasil analisis pada fitur profiling dan permasalahan di atas dapat diusulkan untuk dilakukannya penyesuaian dengan membuat koneksi database di dalam pentaho dan kemudian dibuatkan *default value* dari variabel dan parameter yang membuat *ktr*.

### 3.3.3 Usulan Penyesuaian Job Trigger Aplikasi Eksisting



**Gambar 4.** Component Diagram

Berdasarkan gambar 4 diusulkan adanya penyesuaian dan perubahan dalam mengganti eksekutor dari *Pentaho* dengan *Carte* server yang merupakan eksekutor transformasi berbasis web.

### 3.3.4 Rancangan Database

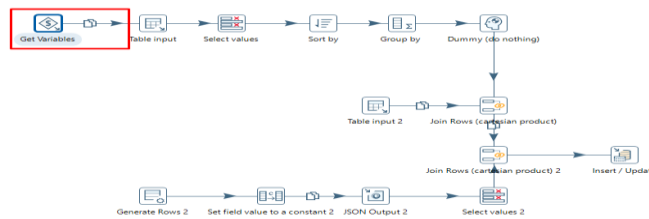
Pada aplikasi eksisting terdapat banyak tabel di dalam database *db\_app\_ta* yang memiliki fungsi dan tujuan sesuai kebutuhannya. Pada tabel 7 menjelaskan mengenai beberapa tabel database yang ada pada penelitian ini.

**Tabel 7.** Daftar Tabel Database

No	Nama tabel	Deskripsi
1	val_distribution_table_result	Menyimpan hasil dari <i>run fitur value distribution</i>
2	datacomp_result	Menyimpan hasil dari <i>run data Completeness</i>
3	shownull_result	Menyimpan hasil dari <i>run show null</i>
4	clustering_result	Menyimpan hasil dari <i>run clustering</i>
5	val_similarity_table_result	Menyimpan hasil dari <i>run value similarity</i>
6	data_deduplication_table_result	Menyimpan hasil dari <i>run data deduplication</i>

### 3.4 Implementasi

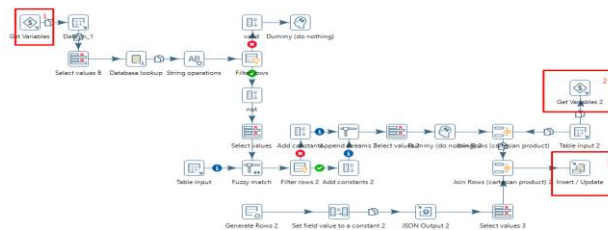
#### 3.4.1 Implementasi Logic Pentaho Modul Value Distribution



**Gambar 5.** Penyesuaian Logic Value Distribution

Pada tahap ini melakukan penyesuaian dalam *logic value distribution* yang perlu dilakukan suatu penyesuaian dengan tujuan agar logic tersebut dapat diintegrasikan dengan sebuah platform yang memiliki tampilan web. Pada gambar 5 sendiri dapat dilihat adanya penyesuaian pada “Get Variabel” yang berguna dalam mengambil nilai dari suatu variabel yang akan dilakukan pengecekan suatu *value distribution* sesuai dengan suatu inputan user pada form aplikasi eksisting. Dalam aplikasi eksisting sebelumnya variabel tidak dapat dilakukan inisiasi variabel yang menyebabkan langkah “Get Variabel” tidak dapat mengambil nilai dari inputan user yang digunakan sehingga logic menjadi error. Oleh karena itu dilakukan penyesuaian pada langkah “Get Variabel” menjadi tidak error dan dapat digunakan dalam implementasi pengolahan data pada modul *value distribution*.

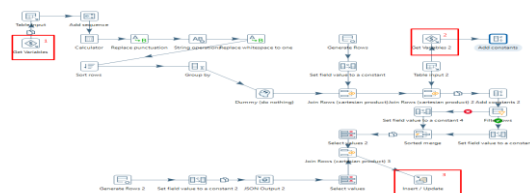
#### 3.4.2 Implementasi Logic Pentaho Modul Data Completeness



**Gambar 1.** Penyesuaian Logic Data Completeness

Dalam modul *data completeness* dilakukan penyesuaian agar membuat ktr *logic* dapat digunakan ketika di lakukan integrasi pada aplikasi web. Penyesuaian yang dilakukan terdiri dari 3 hal yaitu penyesuaian pada langkah “Get variable 1”, “Get Variable 2” dan “insert/update” yang dapat dilihat pada Gambar 6. “Get Variable 1” dilakukan penyesuaian untuk dapat mengambil nilai pada inputan yang di inputkan user dalam aplikasi eksisting. Pada langkah “Get Variable 2” penyesuaian dilakukan inisiasi variable yang digunakan untuk menerima inputan user sama pada langkah “Get Variable 1”. Pada langkah “Insert/Update” digunakan untuk penyimpanan data hasil proses dalam penggunaan modul pada aplikasi eksisting. Dilakukan penyesuaian pada *connection* sesuai dengan koneksi database dan *update fields* untuk pengaturan terhadap kolom apa saja yang akan di *update* atau insert setelah melakukan *running* modul.

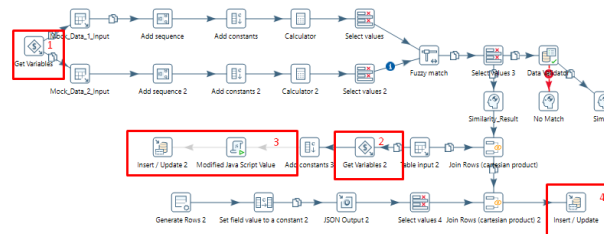
#### 3.4.3 Implementasi Logic Pentaho Modul Clustering



**Gambar 2.** Penyesuaian Logic Clustering

Implementasi *logic clustering* perlu dilakukan penyesuaian pada step di ktr *logic clustering* dengan tujuan untuk dapat diintegrasikan pada aplikasi web. Dalam Gambar 7 menjelaskan penyesuaian langkah yang dilakukan yaitu “Get Variable”, “Get Variable 2”, “Insert/Update”. Langkah “Get Variable” disesuaikan agar membuat ktr *logic clustering* dapat digunakan karena memiliki fungsi untuk menerima suatu inputan dari user pada aplikasi dan dilakukan eksekusi pada pentaho. Begitu pun pada langkah “Get Variable 2” dilakukan penyesuaian agar dapat menginisiasi variable agar logic dapat menerima inputan dari user yang ada pada aplikasi. Penyesuaian selanjutnya pada langkah “Insert/Update” dilakukan penyesuaian koneksi database sesuai dengan yang ada untuk menyimpan hasil dari running *logic clustering*. Selanjutnya mengisi tabel update fields yang memiliki fungsi untuk menentukan kolo apa saja yang ada di dalam database yang akan di *insert/update*.

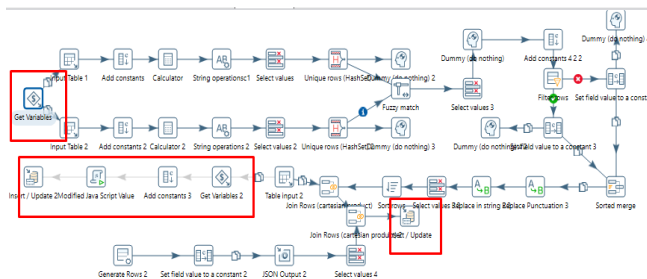
### 3.4.4 Implementasi Logic Pentaho Modul Value similarity



Gambar 3. Penyesuaian Logic Value Similarity

Penyesuaian langkah yang dilakukan pada modul ini bertujuan agar ktr dapat diintegrasikan dengan aplikasi web. Berdasarkan Gambar 8 penyesuaian meliputi “Get Variable”, “Get variable 2”, “Disable step”, dan “insert/update”. Langkah pertama penyesuaian pada “Get variable” hal ini dilakukan dengan tujuan agar *file logic* dapat menerima input dari user, terdapat inisiasi variable yang sesuai dengan inputan user pada aplikasi begitupun dengan “Get variable 2”. Penyesuaian selanjutnya adalah membuat disable pada langkah karena tidak perlu dilakukan hal tersebut karena data yang dimasukkan merupakan data yang sama diambil pada proses sebelumnya agar tidak terjadi *double data*. Penyesuaian terakhir yaitu koneksi database dan table di mana nantinya hasil *running* dari modul disimpan dan dieksekusi pada langkah “insert/update”.

### 3.4.5 Implementasi Logic Pentaho Modul data deduplication



Gambar 4. Penyesuaian Logic Data Deduplication

Berdasarkan Gambar 9 dilakukan penyesuaian “Get Variable”, “Desable step”, “Insert/Update” dengan tujuan agar ktr *logic clustering* dapat diintegrasikan pada aplikasi web. Penyesuaian “Get variable” untuk membuat *file logic* yang dapat menerima inputan user pada aplikasi. Penyesuaian berikutnya dengan menonaktifkan beberapa langkah yaitu “Get Variable 2”, “Add constanta 3”, “modified java script value”, “ insert/update 2” dilakukan seperti pada modul sebelumnya karena jika aktif akan membuat data menjadi berulang. Penyesuaian terakhir adalah penyesuaian *database* dan tabel yang akan menyimpan hasil dari *running* modul dengan mengganti koneksi *database* sesuai dengan penyesuaian sebelumnya.

### 3.5 Pengujian

Pengujian dilakukan sesuai metode penyelesaian masalah di mana dengan iterasi sesuai modul yang telah di implementasikan sebelumnya dan akan di uji mengenai fungsinya sesuai dengan metode yang digunakan. Pengujian yang dilakukan dengan menggunakan fungsional testing atau *blackbox testing* yang berfokus pada modul yang dilakukan penyesuaian dapat diakses dan digunakan dengan baik. Pengujian menggunakan skenario pengetesan yang dilakukan dengan cara *Increment* menggunakan *blackbox testing* yang dijelaskan pada tab 8 sebagai berikut.

**Tabel 8.** Testing Stage

No	Skenario	Hasil yang diharapkan	Hasil yang terjadi	Hasil Sesuai? (Y/N)
1	Menjalankan modul <i>value distribution</i>	Proses <i>value distribution</i> berhasil di <i>running</i>	<i>Value Distribution</i> berhasil di <i>running</i> dan menampilkan data	Y
2	Menjalankan modul <i>Data Completeness</i>	Proses <i>Data Completeness</i> berhasil di <i>running</i>	<i>Data Completeness</i> berhasil di <i>running</i> dan menampilkan data	Y
3	Menjalankan modul <i>Clustering</i>	Proses <i>Clustering</i> berhasil di <i>Running</i>	<i>Clustering</i> berhasil di <i>running</i> dan menampilkan data	Y
4	Menjalan modul <i>value similarity</i>	Proses <i>value similarity</i> berhasil di <i>running</i>	<i>Value similarity</i> berhasil di <i>running</i> dan menampilkan data	Y
5	Menjalankan modul <i>data deduplication</i>	Proses <i>Data Deduplication</i> berhasil di <i>running</i>	<i>Data Deduplication</i> berhasil di <i>running</i> dan menampilkan data	Y

#### 4. KESIMPULAN

Berdasarkan hasil dari penelitian dapat ditarik kesimpulan yaitu pengimplementasian *data profiling* yang terdapat dalam fitur *Data Quality Management (DQM)* pada aplikasi *Data Governance* memerlukan penyesuaian dalam logic pentaho agar dapat digunakan tanpa adanya error serta penyesuaian job trigger yang digunakan sebagai eksekutor pada aplikasi pentaho dari Pan.bat menjadi Carte dengan tujuan agar eksekusi menjadi lebih baik. Selanjutnya penyesuaian perlu dilakukan dengan tujuan agar fitur *Data Quality Management (DQM)* dapat dijalankan juga dalam platform Linux dengan penyesuaian *syntax* yang berfungsi untuk mengubah format penulisan suatu file agar perintah tersebut dapat dijalankan pada terminal platform Linux.

#### REFERENCES

- [1] Dama International, *DAMA-DMBOK2 (Data Management Body of knowledge)*, 2 ed. Basking Ridge, NJ 07920 USA: Technics Publication, 2017.
- [2] T. F. Kusumasari dan Fitria, “Data profiling for data quality improvement with OpenRefine,” *2016 Int. Conf. Inf. Technol. Syst. Innov. ICITSI 2016 - Proc.*, no. June, 2017, doi: 10.1109/ICITSI.2016.7858197.
- [3] Shelly Kramer, “The High Costs of Dirty Data - V3B: Marketing and Social Media Agency,” Mar 25, 2015. <https://v3b.com/2015/03/the-high-costs-of-dirty-data/#ixzz3Y2t602Ex> (diakses Nov 08, 2021).
- [4] M. R. Effendy, T. F. Kusumasari, dan M. A. Hasibuan, “Analysis and Design of Data Quality Monitoring Application using Open Source Tools: A Case Study at a Government Agency,” no. Iccetim 2019, hal. 214–219, 2020, doi: 10.5220/0009867402140219.
- [5] A. N. Laksono, T. F. Kusumasari, dan M. A. Hasibuan, “Implementation of Data Quality Management Application Architecture,” *SCITEPRESS – Sci. Technol. Publ.*, no. Iccetim 2019, hal. 268–274, 2020, doi: 10.5220/0009868302680274.
- [6] H. A. Sulisty, T. F. Kusumasari, dan E. N. Alam, “Implementation of Data Cleansing Pattern Module for Data Quality Management Application using Open Source Tools,” *2020 3rd Int. Conf. Comput. Informatics Eng. IC2IE 2020*, hal. 7–12, 2020, doi: 10.1109/IC2IE50715.2020.9274628.
- [7] F. Ridzuan dan W. M. N. Wan Zainon, “A review on data cleansing methods for big data,” *Procedia Comput. Sci.*, vol. 161, hal. 731–738, 2019, doi: 10.1016/j.procs.2019.11.177.
- [8] J. vom Brocke, A. Hevner, dan A. Maedche, “Introduction to Design Science Research,” <https://www.researchgate.net/publication/345430098> *Introd. to Des. Sci. Res. Chapter · Sept. 2020 DOI 10.1007/978-3-030-46781-4\_1 CITATIONS*, no. November, hal. 1–13, 2020, doi: 10.1007/978-3-030-46781-4\_1.
- [9] A. M. Kuhn, *Code Complete*, vol. 47, no. 4. 2005.
- [10] A. Alshamrani dan A. Bahattab, “A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model,” *IJCSI Int. J. Comput. Sci. Issues*, vol. 12, no. 1, hal. 106–111, 2015, [Daring]. Tersedia pada: [https://www.academia.edu/10793943/A\\_Comparison\\_Between\\_Three\\_SDLC\\_Models\\_Waterfall\\_Model\\_Spiral\\_Model\\_and\\_Incremental\\_Iterative\\_Model](https://www.academia.edu/10793943/A_Comparison_Between_Three_SDLC_Models_Waterfall_Model_Spiral_Model_and_Incremental_Iterative_Model).
- [11] S. Supriyono, “Software Testing with the approach of Blackbox Testing on the Academic Information System,” *Int. J. Inf. Syst. Technol.*, vol. 3, no. 2, hal. 227–233, 2020.
- [12] A. Stojkov, L. Kazi, dan M. Bla, “Software Reengineering with Object-Oriented n-Tier Architecture : Case of Desktop-to-Web Transformation,” *Int. Conf. Inf. Technol. Dev. Educ. – ITRO 2020*, 2020.
- [13] S. K. Dewanti, T. F. Kusumasari, dan E. N. Alam, “ANALISIS DAN KLASIFIKASI PROSES DATA PROFILING PADA DATA QUALITY MANAGEMENT MENGGUNAKAN PENTAHO DATA INTEGRATION ANALYSIS AND CLASSIFICATION OF DATA PROFILING PROCESSES IN DATA QUALITY,” 2021.



- [14] D. A. Rahmani, T. F. Kusumasari, dan E. N. Alam, “Addition of Process Decomposition in Open Source Tools-Based Cleansing Data Modules,” *ICODSA*, hal. 129–134, 2021, doi: 10.1109/icodsa53588.2021.9617555.
- [15] R. A. Nugroho, T. F. Kusumasari, dan E. N. Alam, “INTEGRASI PAKET MODUL DATA CLEANSING PADA APLIKASI DATA QUALITY MANAGEMENT MENGGUNAKAN OPEN SOURCE TOOLS INTEGRATION OF DATA CLEANSING MODULE PACKAGES IN DATA QUALITY,” 2021.
- [16] S. Nidhra, “Black Box and White Box Testing Techniques - A Literature Review,” *Int. J. Embed. Syst. Appl.*, vol. 2, no. 2, hal. 29–50, 2012, doi: 10.5121/ijesa.2012.2204.
- [17] K. F. Salmawati dan U. Telkom, *ANALISIS PERFORMANSI MODUL PROFILING PADA APLIKASI DATA QUALITY MANAGEMENT BERBASIS OPEN SOURCE TUGAS AKHIR Tugas akhir disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana dari Universitas Telkom ANALISIS PERFORMANSI MODUL PROFILING PADA APLI*, vol. 1202160041. 2020.