

PEMANFAATAN BASIS DATA ORACLE PADA SISTEM INFORMASI WORK ORDER PADA PT XYZ DI JAKARTA TIMUR

Hendro Purwoko
Program Studi Informatika
Universitas Indraprasta PGRI
hendroprwk08@gmail.com

Abstrak— Kemampuan basis data Oracle dapat bekerja pada banyak *platform* merupakan salah satu fasilitas yang dimanfaatkan untuk menyimpan data dengan membuat *subroutine* sehingga dalam melakukan eksekusi data menjadi cepat. Kemampuan tersebut dimanfaatkan untuk membangun basis data pada Sistem Informasi *Work Order* Pada PT XYZ di Jakarta Timur yang bergerak pada bidang Jasa Perbaikan Alat Kesehatan di Rumah Sakit, dengan menerapkan metode *Rapid Application Development* agar setiap tahapan dapat langsung memproses *feedback*. Hasil dari pemanfaatan Oracle sebagai Basis Data mampu mendukung kinerja PT XYZ dalam melayani konsumen dengan ratusan transaksi perhari secara cepat dan mudah diakses oleh aplikasi lain yang memiliki beragam bahasa pemrograman.

Keywords— Oracle, basis data, rapid application development, work order

I. PENDAHULUAN

Oracle merupakan salah satu aplikasi basis data relasional yang dapat digunakan sebagai media penyimpan data di berbagai *platform*. Teknologi *client-server* pada Oracle merupakan salah satu *Database Management System (DBMS)* yang lengkap [1], salah satu kemampuan yang dapat dimanfaatkan oleh pengembang aplikasi dan *database engineer* adalah pembuatan *query* atau perintah untuk menampilkan data dari database dengan mengambil dari tabel-tabel yang berelasi [2] dan melakukan operasi-operasi lainnya dengan kode yang mudah dipahami.

Kemampuan basis data Oracle tersebut dimanfaatkan untuk membangun basis data pada Sistem Informasi *Work Order* Pada PT XYZ di Jakarta Timur yang bergerak pada bidang Jasa Perbaikan Alat Kesehatan di Rumah Sakit seluruh Indonesia yang saat ini proses operasi kerja pada perusahaan masih menggunakan aplikasi perkantoran yang tak efisien dan menimbulkan banyak permasalahan seperti: pemberian harga pada *customer* yang tak dapat terekam dengan baik, lamanya waktu untuk membuat laporan kerja bulanan atau tahunan serta staf juga kesulitan untuk melihat histori perubahan harga jasa dan alat yang digunakan.

Pembangunan basis data ini menerapkan metode *Rapid Application Development* agar proses masing-masing tahapan berkesinambungan dan dapat langsung memproses *feedback*. Hasil dari pemanfaatan Oracle sebagai Basis Data mampu mendukung kinerja PT XYZ dalam melayani konsumen dengan ratusan transaksi perhari secara cepat dan mudah diakses oleh aplikasi lain yang memiliki beragam bahasa pemrograman dengan menyajikan *sub routine* yang dapat mengoptimalkan waktu eksekusi data.

II. TINJAUAN PUSTAKA

A. Oracle

Oracle adalah *database* relasional yang terdiri dari kumpulan data dalam suatu sistem manajemen basis data *RDBMS (Relational Data Base Management System)* yang *multi-platform*. Basis data Oracle ini pertama kali dikembangkan oleh Larry Ellison, Bob Miner dan Ed Oates lewat perusahaan konsultasinya bernama *Software Development Laboratories (SDL)* pada tahun 1977. Pada tahun 1983, perusahaan ini berubah nama menjadi Oracle Corporation sampai sekarang. Server Oracle berisi Oracle Instance dan Oracle Database, dimana Oracle Instance berisi struktur memory yang disebut dengan *system-global-area (SGA)* dan *background-process* yang dipergunakan oleh *server* Oracle untuk mengatur *database* [1].

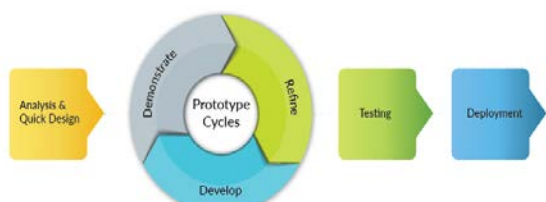
B. Work Order

Work Order merupakan tugas atau pekerjaan bagi pelanggan, yang dapat dijadwalkan atau ditugaskan kepada seseorang. Perintah semacam itu mungkin berasal dari permintaan pelanggan atau dibuat secara internal di dalam organisasi [5] *Work Order* juga dapat dibuat sebagai tindak lanjut dari Inspeksi atau Audit. *Work Order* dapat berisi satu atau beberapa hal berikut: Instruksi, Perkiraan biaya, Formulir, Tanggal dan waktu untuk melaksanakan perintah kerja, Informasi tentang lokasi dan entitas untuk melaksanakan perintah kerja dan orang yang diberi tugas kerja ditugaskan. Pemberian beban kerja harus sesuai dan memenuhi agar hak-haknya [7].

III. METODE PENELITIAN

Metode yang digunakan dalam pembangunan basis data adalah *Rapid Application Development* yang lebih cepat dan efisien dibandingkan metode

tradisional atau dikenal dengan Metode *Water Fall* yang memiliki alur kerja berurutan.



Gbr 1. Metode *Rapid Application Development*

Metode *Rapid Application Development* memiliki alur yang berkesinambungan saat membangun prototipe, setiap *feedback* yang masuk dapat langsung dieksekusi hingga sistem yang dibuat berfungsi dengan baik.

TABEL 1
PENERAPAN METODE
RAPID APPLICATION DEVELOPMENT

Tahap	Pelaksanaan
<i>Analysis & Quick Design</i>	Menganalisa kebutuhan tabel, kolom, tipe data juga <i>User Management</i> yang akan diterapkan pada sistem. Untuk mempercepat proses eksekusi maka perlu dilakukan pembuatan <i>View, Procedure dan Trigger</i> agar memudahkan perancang sistem.
<i>Develop</i>	Menyatakan hasil dari tahap sebelumnya menjadi basis data yang berelasi.
<i>Refine</i>	Melakukan koreksi dan pemeriksaan lebih lanjut agar basis data siap digunakan.
<i>Demonstrate</i>	Menjalankan seluruh <i>Query</i> yang telah dibuat untuk memastikan tak ada kesalahan kode
<i>Testing</i>	Menerapkan <i>Query</i> basis data dengan aplikasi yang sedang dibangun dan melihat tingkat keefektifitasan.
<i>Deployment</i>	Penyebaran basis data dengan melakukan instalasi pada <i>server</i> PT XYZ.

IV. HASIL DAN PEMBAHASAN

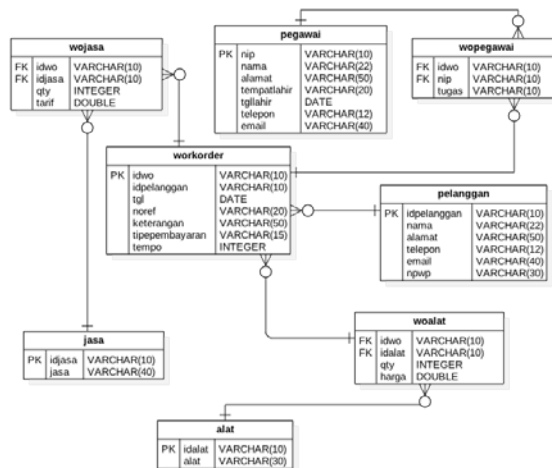
Tahap awal dalam menggunakan basis data Oracle adalah masuk melalui akun *System* dan membuat *tablespace* dengan menambahkan fasilitas penambahan ukuran secara otomatis.

```
create tablespace dbworkorder datafile
'dbworkorder' size 10M autoextend on;
```

Kemudian membuat pengguna basis data sekaligus memberikan hak akses kepada pengguna yang berlaku sebagai administrator.

```
create user admn identified by orcl
default tablespace dbworkorder;
grant create session, connect, resource,
dba to admn;
```

Proses selanjutnya adalah login menggunakan akun yang telah dibuat dilanjutkan dengan pembuatan tabel yang berelasi melalui *primary key* dan *foreign key*, pada sistem *work order* ini menggunakan beberapa table, yaitu: Pegawai, Jasa, Alat, WorkOrder, WOJasa, WOAlat dan WOpegawai yang dibuat sesuai dengan struktur yang tergambar pada ERModel dibawah ini.



Gbr 2. *Entity Relationship Model*

Ada beberapa kode pada basis data yang umum dilakukan untuk optimalisasi yaitu: *view, procedure, function* dan *trigger* namun tak semuanya harus diterapkan hal ini melihat dari kebutuhan dalam mengembangkan aplikasi.

Procedure merupakan *sub routine* yang berisi *statement* sehingga semua daya hanya berjalan pada sisi *server* saja karena telah disimpan dalam *database schema* dan tak perlu melakukan *parsing* yang memakan waktu lagi saat menjalankan *query*. Pada sistem ini ada beberapa *procedure* untuk melakukan *input data work order* yang akan mendapatkan beban paling besar diantara semua tabel:

```
1. Procedure input table work order
create or replace procedure
pworkorder
(
    p_idwo in varchar,
    p_idpelanggan in varchar,
    p_tgl in date,
    p_noref in varchar,
    p_keterangan in varchar,
    p_tipepembayaran in varchar,
    p_tempo in number
)
as
begin
    insert into workorder values
    (p_idwo, p_idpelanggan, p_tgl,
    p_noref, p_keterangan,
    p_tipepembayaran, p_tempo);
end;
```

2. Procedure input table *wojasa*

```
create or replace procedure pwojasa
(
  p_idwo in varchar,
  p_idjasa in varchar,
  p_qty in number,
  p_tarif in number
)
as
begin
  insert into wojasa values (p_idwo,
  p_idjasa, p_qty, p_tarif);
end;
```

3. Procedure input table *woalat*

```
create or replace procedure pwoalat
(
  p_idwo in varchar,
  p_idalat in varchar,
  p_qty in number,
  p_harga in number
)
as
begin
  insert into woalat values (p_idwo,
  p_idalat, p_qty, p_harga);
end;
```

4. Procedure input table *wopegawai*

```
create or replace procedure
pwopegawai
(
  p_idwo in varchar,
  p_nip in varchar,
  p_job in varchar
)
as
begin
  insert into wopegawai values
(p_idwo, p_nip, p_job);
end;
```

Jenis *Sub routine* lain yaitu *view* untuk meringkas kode ketika menampilkan data yang relevan dengan menggunakan *subquery* dan atau *join* jika tabel memiliki relasi dengan tabel lain. Adapun *view* yang dibutuhkan adalah sebagai berikut:

1. Menampilkan order berdasarkan pelanggan pertahun dan perbulan dengan *query* sebagai berikut:

```
create or replace view
vorderpelanggan as
select
extract (year from tgl) as tahun,
extract (month from tgl) as bulan,
idwo,
idpelanggan,
(select nama from pelanggan where
pelanggan.idpelanggan =
workorder.idpelanggan ) as nama,
```

```
(select case when sum(qty * tarif) is
null then 0 else sum(qty * tarif) end
from wojasa where wojasa.idwo =
workorder.idwo) as jasa,
(select case when sum(qty * harga) is
null then 0 else sum(qty * harga) end
from woalat where woalat.idwo =
workorder.idwo) as alat,
(select case when sum(qty * tarif) is
null then 0 else sum(qty * tarif) end
from wojasa where wojasa.idwo =
workorder.idwo) +
(select case when sum(qty * harga) is
null then 0 else sum(qty * harga) end
from woalat where woalat.idwo =
workorder.idwo) as Jumlah
from workorder
```

Contoh hasil keluaran dari kode diatas adalah:

TABEL 2
HASIL KELUARAN DATA ORDER
BERDASARKAN PELANGGAN PERTAHUN DAN PERBULAN

TAHUN	BULAN	IDWO	IDPELANGGAN	NAMA	JASA	ALAT	JUMLAH
2018	2	WO001	P002	Klinik Permata Karta	8415000	0	8415000
2018	2	WO002	P003	Puskesmas Indarto Winangun	0	7600000	7600000
2018	2	WO003	P002	Klinik Permata Karta	2080000	0	2080000
2018	3	WO004	P001	RS Budi Warman	8830000	440690000	449520000
2018	4	WO005	P005	RSUD Mawardi	2940000	0	2940000
2018	4	WO006	P005	RSUD Mawardi	1600000	489400000	491000000
2018	4	WO007	P002	Klinik Permata Karta	0	20300000	20300000

2. Merekap Total Order pertahun dan perbulan

```
create or replace view
vordertahunbulan as
select tahun, bulan, sum(jasa) as
jasa, sum(alat) as alat, sum(jumlah)
as jumlah from vorderpelanggan group
by tahun, bulan
```

Hasil dari kode pembuatan *view* diatas adalah:

TABEL 3
HASIL KELUARAN REKAP ORDER
PELANGGAN PERTAHUN DAN PERBULAN

TAHUN	BULAN	JASA	ALAT	JUMLAH
2018	4	4540000	509700000	514240000
2018	2	10495000	7600000	18095000
2018	3	8830000	440690000	449520000

Dari tabel diatas terlihat bahwa total jasa, alat serta jumlah ditampilkan dengan jelas sehingga dapat mempermudah kinerja pemrogram.

3. Menampilkan historikal tarif jasa yang pernah ditawarkan kepada pelanggan berdasarkan id, hal ini dibuat mengingat penetapan tarif dapat berubah sesuai negosiasi dan lokasi pelanggan

```
create or replace view vtarifjasa as
select wojasa.idjasa, jasa.jasa,
wojasa.tarif
from wojasa inner join jasa
```

```
on wojasa.idjasa = jasa.idjasa
order by idjasa asc
```

Hasil olah *query* diatas terbentuk menjadi tabel dibawah ini:

TABEL 4
HASIL KELUARAN
HISTORIKAL PEMBERIAN TARIF KEPADA PELANGGAN

IDJASA	JASA	TARIF
J001	Pb. Patient Monitor	400000
J002	Penggantian komponen	450000
J002	Penggantian komponen	200000
J003	Pb. Spirometer	1230000
J004	Pb. Threadmill	825000
J004	Pb. Threadmill	900000

- Menampilkan historikal harga alat yang ditawarkan kepada pelanggan berdasarkan id memiliki fungsi untuk melihat historikal pemberian harga yang diberikan pada pelanggan

```
create or replace view vhargaalat as
select woalat.idalat, alat.alat,
woalat.harga
from woalat inner join alat
on woalat.idalat = alat.idalat
order by idalat asc
```

Hasil keluaran kode diatas adalah:

TABEL 5
HASIL KELUARAN HISTORIKAL
PEMBERIAN HARGA ALAT KEPADA PELANGGAN

IDALAT	ALAT	HARGA
A001	Selimut	150000
A001	Selimut	150000
A002	Trolley	150000
A004	Otoskop	3200000
A004	Otoskop	3700000
A005	Stetoskop	2500000
A007	Alat USG	12600000
A007	Alat USG	12800000

Untuk mempermudah pemrogram dalam menampilkan data yang spesifik dan dapat di selipkan pada sebuah *query* perlu dibuatkan *function* yang berguna untuk mendapatkan total order berdasarkan idwo, *query* tersebut adalah:

```
create or replace function
gettinvoice(id in varchar) return number
IS
    valtotal number;
BEGIN
    select jumlah into valtotal
    from vorderpelanggan
    where idwo = id;

    return valtotal;
end;
```

TABEL 6
HASIL KELUARAN DARI FUNCTION GETTINVOICE

GETTINVOICE('WO002')
10000000

Pada tabel *workorder* terdapat relasi ke beberapa tabel yaitu: *wojasa*, *wopegawai* dan *woalat*. Hal ini dapat memberatkan jika ketika proses penghapusan data jika dilakukan satu persatu karenanya sebelum tabel tersebut dihapus maka akan dilakukan penghapusan data *wojasa*, *wo* aat dan *wopegawai* dahulu. Proses ini dapat dilakukan dengan menambahkan *subroutine* berupa *trigger* seperti yang tertera pada kode dibawah ini:

```
create or replace trigger trDelWO
before delete on workorder
for each row
begin
    delete from wojasa where idwo =
:old.idwo;
    delete from woalat where idwo =
:old.idwo;
    delete from wopegawai where idwo =
:old.idwo;
end;
```

Dari *query* diatas menghasilkan *banyaksub routine* yang siap digunakan oleh pengembang aplikasi dengan menggunakan berbagai bahasa pemrograman untuk berbagai *platform* baik berupa sistem informasi berbasis *desktop*, *mobile application* atau *web*.

V. KESIMPULAN

Pemanfaatan basis data Oracle mampu mendukung kinerja PT XYZ dalam membangun sistem informasi *work order* dengan membuat *subroutine* yang terekam pada *database schema* untuk mempercepat operasi *database*. *Subroutine* yang telah dibuat dapat diakses oleh aplikasi lain dengan menggunakan berbagai bahasa pemrograman dari berbagai *platform*. Metode *Rapid Application Development* yang digunakan pun mampu mempercepat proses pembangunan basis data sehingga dapat menyesuaikan kebutuhan staf PT XYZ.

REFERENSI

- Enjang Yusup Ali, Agil Assagaf. Penerapan Data Warehouse Menggunakan Teknologi Oracle Sebagai Solusi Big Data Pada Database Client server. JURNAL DIGIT, Vol. 4 No.1 Mei 2014, pp.12-21
- Junus Sinuraya. 2017. Metode Pencarian Data Menggunakan Query Hash Join Dan Query Nested Join. Jurnal Teknovasi Volume 04, Nomor 01, 2017, 42-50
- Dyah Paramita, Firdaus, Mira Afrina. 2012. Penerapan Data Mart Penjualan Hypermarket XYZ Menggunakan Metode From Enterprise Models To Dimensional Models. Jurnal Sistem Informasi (JSI), VOL. 4, NO. 2, Oktober 2012,
- Hendro Purwoko, Harry Dhika, Sutan Muhammad Arif. 2017. Rancang Bangun Sistem Informasi Work Order Menggunakan Unified Modeling Language (UML) Pada PT XYZ. SEMNAS RISTEK 2017

- [5] J. I. Maanari, R. Sengkey, ST., MT. , Ir. H. F. Wowor, M.Kom., Y. D. Y. Rindengan, ST., MM., Msc. 2013. Perancangan Basis Data Perusahaan Distribusi dengan Menggunakan Oracle. e-journal Teknik Elektro dan Komputer (2013)
- [6] Work Order. 2018. Retrieved Mei 20, 2018, From https://en.wikipedia.org/wiki/Work_order
- [7] Sutabri, T. 2012. Analisis Sistem Informasi. Jakarta: Andi.
- [8] Hartono, B. 2013. Sistem Informasi Manajemen Berbasis Komputer. Jakarta: Rineka Cipta.
- [9] Syamsul. 2012. Rancang Bangun Sistem Pendukung Keputusan Pembagian Beban Kerja Dosen (BKD) Berbasis Logika Fuzzy. Jurnal Litek, 9(2), 124–133.