

PERANCANGAN MIDDLEWARE DALAM PELAYANAN PUBLIK

Bernard Hasibuan
Lisa Ratnasari
Soecahyadi

UNIVERSITAS SAHID PRESS
Jl. Prof. Dr Supomo, SH No. 84 Tebet, Jakarta

PERANCANGAN MIDDLEWARE DALAM PELAYANAN PUBLIK

**Bernard Hasibuan
Lisa Ratnasari
Soecahyadi**

SAHID UNIVERSITY PRESS
Jalan Prof. Dr. Supomo SH No 84, Tebet, Jakarta

PERANCANGAN MIDDLEWARE DALAM PELAYANAN PUBLIK

Hak cipta pada penulis, dilarang keras mengutip, menjiplak, memphoto copy baik sebagian atau keseluruhan dari isi buku ini tanpa mendapat ijin tertulis dari pengarang dan penerbit

Penulis : Bernard Hasibuan
Lisa Ratnasari
Soecahyadi

Editor : Ferry Ferdana Putra

Page make up : Dode Subchan

Desain cover : Dode Subchan

Dicetak oleh : Sahid University Press

Cetakan Pertama, Juni 2020

ISBN : 978-602-53736-2-6

@HAK CIPTA DILINDUNGI OLEH UNDANG-UNDANG

Sanksi pelanggaran pasal 44 :

Undang-undang Nomor 7 tahun 1987 tentang

Perubahan atas Undang-undang Nomor 6 tahun 1982

Tentang Hak Cipta

1. Barang siapa dengan sengaja dan tanpa hak mengumumkan atau memperbanyak suatu ciptaan atau memberi izin untuk itu, dipidana dengan tindakan penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp. 100.000.000,- (seratus juta)
2. Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran Hak Cipta sebagaimana dimaksud dalam ayat (1), dipidana dengan pidana paling lama 5 (lima) tahun dan/atau denda paling banyak Rp. 50.000.000,- (lima puluh juta rupiah).

KATA PENGANTAR

Pelayanan publik saat ini penuh tantangan dan dinamika pada masyarakat dalam era milenial. Tantangan yang terbesar adalah bagaimana menyediakan informasi yang cepat, lengkap, akurat dan terkini. Segala perubahan dan perkembangan data atau informasi harus dapat dirangkum dalam suatu media yang dapat diakses oleh masyarakat dengan baik. Hal lain yang juga menjadi tantangan adalah sharring data dan koordinasi kerja antar satuan kerja dalam suatu pelayanan publik. Penyediaan data dan informasi yang berkualitas dapat mendukung pengambilan keputusan yang baik dalam pelayanan pada masyarakat.

Dalam rangka penyediaan informasi yang berkualitas dan terkini dibutuhkan suatu super sistem yang dapat mengakomodir berbagai sistem pada satuan kerja. Super sistem yang dikenal Middleware akan berkerja diatas berbagai sistem pada satuan kerja, dan menyediakan data atau informasi hasil dari kolaborasi berbagai sumber informasi.

Penulis

DAFTAR ISI

Kata Pengantar	iv
Daftar Isi	v
Daftar Tabel	vi
Daftar Gambar	viii

BAB KERANGKA KONSEPTUAL SISTEM

1

1.1. Karakter Sistem	1
1.2. Konsep Dasar Informasi	3
1.3. Siklus Informasi	3
1.4. Sistem Informasi	4
1.5. Komponen Sistem Informasi	5

BAB BASIS DARA

2

2.1. Istilah-istilah Database	7
2.2. Struktur basis Data	12
2.3. Perbedaan OLAP dan OLTP	14
2.4. Entity Relationship Model	18

BAB MIDDLEWARE

3

3.1. Konsep Middleware	23
3.2. Tujuan Penggunaan Middleware	24
3.3. Manfaat Menggunakan Middleware	24
3.4. Contoh Layanan Middleware	25
3.5. Tujuan Umum Middleware	25
3.6. Lingkungan Komputasi	26

BAB ARSITEKTUR APLIKASI

4

4.1. Konsep arsitektur Aplikasi	29
4.2. Bahasa Pemrograman	31
4.3. Topologi	35
4.4. Virtualisasi	36
4.5. Perangkat Lunak	44

BAB DED MIDDLEWARE

5

5.1. Struktur dan Alur Proses Bisnis	53
5.2. Arsitektur Aplikasi	58

DAFTAR PUSTAKA

DAFTAR TABEL

Tabel 2.1. Simbol-simbol diagram ER	19
Tabel 1.2. Komponen use case diagram	22
Tabel 2.2. Komponen activity diagram	23
Tabel 3.1. Parameter e-planning	69
Tabel 5.2. Parameter e-budgeting	59
Tabel 5.3. Parameter e-project planning	70
Tabel 5.4. Parameter e-procurement	71
Tabel 5.5. Parameter e-delivery	72
Tabel 5.6. Parameter e-monev	72
Tabel 5.7. Parameter e-performance	73

DAFTAR GAMBAR

Gambar 1.1. Siklus informasi (Jogiyanto, 2005)	4
Gambar 1.1. Ilustrasi <i>Database</i>	7
Gambar 2.2. Database	8
Gambar 2.3. Struktur Database	12
Gambar 2.4. OLTP vs OLAP	13
Gambar 3.1. Struktur Middleware	24
Gambar 4.1. contoh arsitektur dari revive	30
Gambar 4.2. Arsitektur Aplikasi	31
Gambar 4.3. Konsep client server	36
Gambar 4.4. Layer Sebelum ada Virtualisasi	37
Gambar 4.5. <i>Traditional Infrastructure</i>	38
Gambar 4.6. <i>Virtualization Infrastructure Architecture</i>	39
Gambar 4.7. Hosted Architecture	40
Gambar 4.8. Bare-Metal (Hypervisor) Architecture	40
Gambar 4.9. Virtual Infrascructure	41
Gambar 4.10. Virtual Infrastructure	41
Gambar 4.11. Virtualization Model	43
Gambar 4.12. Detail Native Virtualization	43
Gambar 4.13. <i>Operating System Layer</i>	45
<i>Gambar 4.14. Logo FreeBSD</i>	45
Gambar 4.15. Apache Logo	51

Gambar 5.1. Model GRMS Provinsi Aceh	57
Gambar 5.2. alur GRMS	60
Gambar 5.3. Diagram Alur Sistem E-Planning	61
Gambar 5.4. Diagram Alur Sistem E-Budgeting	62
Gambar 5.5. Diagram Alur Sistem E-Project Planning	64
Gambar 5.6. Diagram Alur Sistem E-Procurement	66
Gambar 5.7. Diagram Alur Sistem E-Delivery	67
Gambar 5.8. Diagram Alur Sistem E-Monev	68
Gambar 5.9. Diagram Alur Sistem E-Monev	68
Gambar 5.10. Relasi Usulan	74
Gambar 5.11. Usulan Topologi & Sistem Keamanan GRMS	75

BAB 1

KONSEPTUAL SISTEM

1.1. KARAKTER SISTEM

Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu (Jogiyanto, 2005).

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen (*components*), batasan sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*) dan sasaran (*objective*) atau tujuan (*goal*) (Jogiyanto, 2005).

1. Komponen sistem (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batasan sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antar sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan (*Environment*)

Bentuk apapun, yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar sistem tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan, jika tidak akan mengganggu kelangsungan sistem tersebut.

4. Penghubung Sistem (*Interface*)

Media menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau interface. Interface memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian dapat terjadi suatu integritas sistem yang membentuk satu kesatuan

5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan dan sinyal. Contohnya, di dalam suatu unit sistem komputer, "program" adalah maintenance input yang digunakan untuk mengoperasikan komputer dan "data" adalah signal input untuk diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Hasil energi yang diolah data dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan.

7. Pengolah sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Contohnya sistem akuntansi, sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran sistem (*Objective*)

Suatu sistem memiliki tinjauan dan sasaran yang pasti dan bersifat *deterministic*. Apabila suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang direncanakan.

1.2. KONSEP DASAR INFORMASI

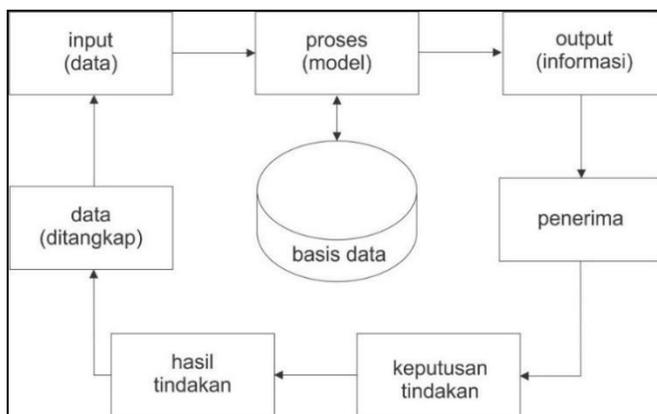
Informasi adalah segala sesuatu keterangan yang bermanfaat untuk para pengambil keputusan atau manajer dalam rangka mencapai tujuan organisasi yang sudah ditetapkan sebelumnya (Gaol, 2008).

Informasi merupakan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya (Jepersen, 2014).

Menurut (Supriyanto, 2007), informasi adalah data yang telah diolah menjadi bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang.

1.3. SIKLUS INFORMASI

Sebuah informasi dapat dijelaskan melalui sebuah siklus, yaitu siklus informasi seperti pada gambar 1.1.



Gambar 1.1. Siklus informasi (Jogiyanto, 2005)

Data yang diolah menjadi suatu model informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan, dan melakukan tindakan. Yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus (Jogiyanto, 2005).

1.4. SISTEM INFORMASI

Sistem informasi adalah suatu sistem yang menerima *input* data dan instruksi, mengolah data sesuai dengan instruksi dan mengeluarkan hasilnya (Davis, 1991). Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi (Alter, 1992). Sistem informasi adalah sebuah rangkaian prosedur formal di mana data dikelompokkan, diproses menjadi informasi, dan didistribusikan kepada pemakai (Hall, 2001)

Menurut (Kertahadi, 2007) sistem informasi adalah alat untuk menyajikan informasi sedemikian rupa sehingga bermanfaat bagi penerimanya. Tujuannya adalah untuk memberikan informasi dalam perencanaan, memulai, pengorganisasian, operasional sebuah perusahaan yang melayani sinergi organisasi dalam proses mengendalikan pengambilan keputusan.

1.5. KOMPONEN SISTEM INFORMASI

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan atau *building block* (Jogiyanto, 2005). Blok bangunan dibagi menjadi:

1. Blok Masukan (*Input Block*)

Input mewakili data yang masuk ke dalam sistem informasi. *Input* di sini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model (*Model Block*)

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*Technology Block*)

Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Terdiri dari tiga bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*) dan perangkat keras.

5. Blok Basis Data (*Database Block*)

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut DBMS (*Database Management Systems*).

6. Blok Kendali (*Controls Block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah

ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

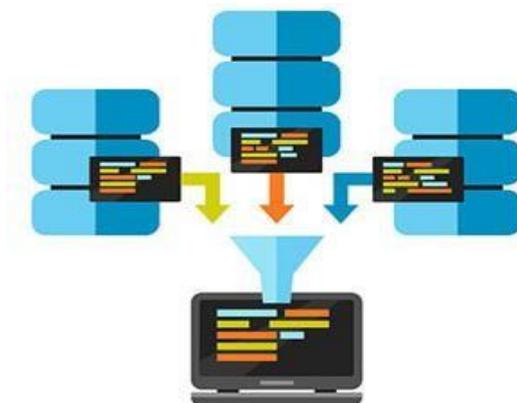
BAB

2

BASIS DATA

2.1. ISTILAH-ISTILAH DATABASE

Dalam bahasa Indonesia, pengertian *database* diartikan sebagai basis data. Yang menunjukan kepada sumber atau tempat dikumpulkannya banyak data yang berbeda. *Database* atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Data sendiri merupakan fakta mengenai obyek, orang dan lain-lain. Data dinyatakan dengan nilai (angka, deretan, karakter atau symbol).



Gambar 1.1. Ilustrasi *Database*

Database menjadi penting karena dapat digunakan untuk menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data dan juga untuk memperbaharui yang kompleks. Sistem manajemen basis data diperlukan dalam proses menginput atau memasukkan data dan mengambil data ke dan dari media storage data. *Database* merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai.



Gambar 2.2. Database

Beberapa hal yang termaksud unsur-unsur dari basis data (database) adalah sebagai berikut:

- Entitas

Entitas adalah orang, tempat, kejadian atau konsep yang informasinya direkam. Pada bidang kesehatan Entity adalah Pasien, Dokter, Kamar.

- Field

Setiap entity mempunyai atribut atau sebutan untuk mewakili suatu entity. Seorang siswa dapat dilihat dari atributnya misalnya, NIM, Nama_siswa, Alamat.

- Record

Record adalah kumpulan isi elemen data (atribut) yang saling berhubungan menginformasikan tentang suatu entity secara lengkap. Contoh Kumpulan atribut NIP, Nama, dan alamat berisikan "01001245566", Sanusi, Jl. Hati suci No 2 Kupang.

- Data Value

Merupakan data aktual atau informasi yang disimpan di tiap data elemen. Isi atribut disebut nilai data.

Jenis Database

Database terdiri dari data yang akan digunakan atau diperuntukkan terhadap banyak user, dari masing-masing user akan menggunakan data tersebut sesuai dengan tugas dan fungsinya. Sebuah sistem yang berisi database disebut Sistem Manajemen Database (DBMS). *Database* juga memiliki berbagai jenis diantaranya adalah :

1. Operational Database

Database ini menyimpan data rinci yang diperlukan untuk mendukung operasi dari seluruh organisasi. Mereka juga disebut subject-area databases (SADB), transaksi database dan produksi database. Contoh: database pelanggan, database pribadi, database inventaris, akuntansi database.

2. Database Relational (Relational Database)

Database relasional merupakan jenis database yang terpopuler. Database relasional memungkinkan pengguna untuk mengakses, update dan mencari informasi berdasarkan hubungan data yang

disimpan dalam tabel yang berbeda. Database relasional juga dapat menjalankan query yang melibatkan beberapa database.

3. Distributed Database

Merupakan suatu kumpulan data-data yang saling berhubungan antara satu dengan yang lainnya tersebar secara fisik dengan jaringan komputer sehingga dapat digunakan bersama-sama. Misalnya database ini dapat di akses dan digunakan oleh kelompok kerja dan departemen kantor tertentu saja atau cabang-cabang pabrik dan lokasi kerja lainnya.

4. Analytical Database

Yaitu database yang menyimpan data-data dan juga informasi yang diambil dari operasional yang telah dipilih dan dari eksternal database. Terdiri dari data maupun informasi yang dirangkum yang paling di perlukan oleh suatu organisasi dan oleh end-user. Orang-orang sering juga menyebutnya informasi database atau manajemen database.

5. End-User Database

Database ini terdiri dari berbagai file data yang dikembangkan oleh end-user di workstation mereka. Contoh dari ini adalah koleksi dokumen dalam spreadsheet, word processing dan bahkan download file.

6. External Database

Database ini menyediakan akses ke eksternal, data milik pribadi online – tersedia untuk biaya kepada pengguna akhir dan organisasi dari layanan komersial. Akses ke kekayaan informasi dari database

eksternal yang tersedia untuk biaya dari layanan online komersial dan dengan atau tanpa biaya dari banyak sumber di Internet.

7. Hypermedia Databases on The Web

Yaitu suatu kumpulan dari halaman multimedia yang saling semuanya terhubung satu sama lain pada sebuah situs web. Terdiri dari halaman home page dan halaman lainnya yang dapat berisi multimedia atau campuran beberapa media seperti teks, gambar, video dan lain-lain.

8. External Database

Merupakan database yang menyediakan akses ke luar, seperti mengakses data online milik pribadi. Untuk menyediakan akses dari database eksternal biasanya memerlukan biaya dari layanan online atau bisa juga tanpa biaya dari banyak sumber yang ada di internet.

9. Navigational Database

Dalam navigasi database, queries menemukan benda terutama dengan mengikuti referensi dari objek lain.

10. In-memory Database

Database di memori terutama bergantung pada memori utama untuk penyimpanan data komputer. Ini berbeda dengan sistem manajemen database yang menggunakan disk berbasis mekanisme penyimpanan. Database memori utama lebih cepat daripada dioptimalkan disk database sejak Optimasi algoritma internal menjadi lebih sederhana dan lebih sedikit CPU mengeksekusi instruksi.

Mengakses data dalam menyediakan memori lebih cepat dan lebih dapat diprediksi kinerja dari disk. Dalam aplikasi di mana waktu respon sangat penting, seperti peralatan jaringan telekomunikasi yang mengoperasikan sistem darurat, database memori utama yang sering digunakan.

11. Document-Oriented Database

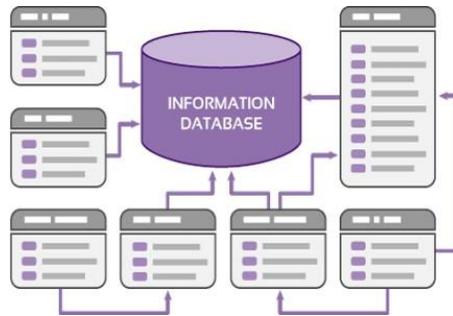
Document-oriented databases merupakan program komputer yang dirancang untuk aplikasi berorientasi dokumen. Sistem ini bisa diimplementasikan sebagai lapisan di atas sebuah database relasional atau objek database. Sebagai lawan dari database relasional, dokumen berbasis database tidak menyimpan data dalam tabel dengan ukuran seragam kolom untuk setiap record. Sebaliknya, mereka menyimpan setiap catatan sebagai dokumen yang memiliki karakteristik tertentu. Sejumlah bidang panjang apapun dapat ditambahkan ke dokumen. Bidang yang dapat juga berisi beberapa bagian data.

12. Real-time Database

Sistem database ini dirancang untuk memberikan solusi pada pekerjaan yang dapat mengalami perubahan secara terus menerus, berbeda dengan database lain yang tidak terpengaruh oleh waktu. Contohnya pada pasar saham yang selalu mengalami perubahan secara cepat dan juga dinamis. Real-Time Database sangat diperlukan untuk catatan medis, analisis, akuntansi dan lain-lain yang dimana pekerjaan mengalami perubahan secara cepat dan dinamis seperti pasar saham tadi.

2.2. STRUKTUR BASIS DATA

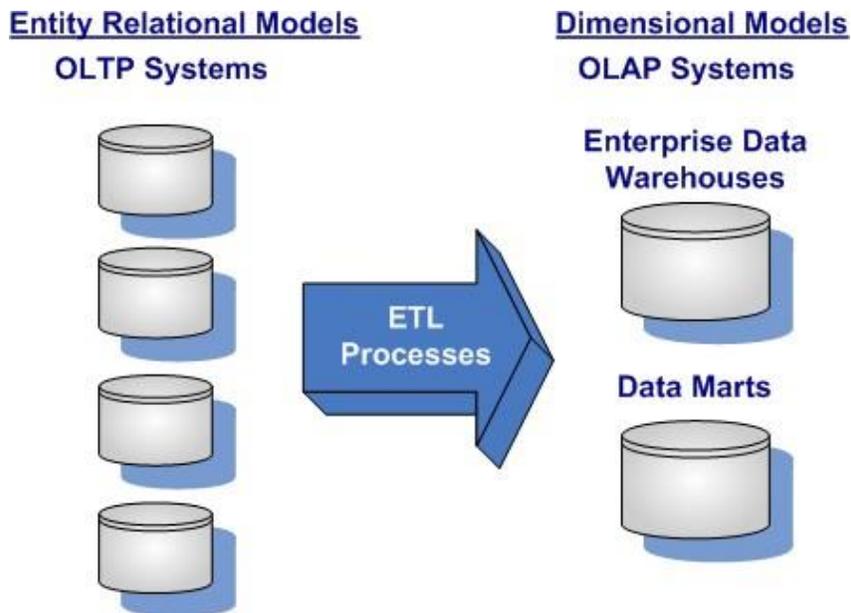
Memodelkan struktur database umumnya menggunakan model relasional, yang mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan di setiap tabel terdiri dari baris dan kolom. Model relasional memungkinkan antar table diwakili dengan nilai sama antar tabel. Sedangkan model hierarkis dan model jaringan memakai carayang lebih terbuka untuk menunjukkan keterwakilan hubungan antar tabel.



Gambar 2.3. Struktur Database

Online Analytical Processing atau disingkat OLAP adalah metode pendekatan untuk menyajikan jawaban dari permintaan proses analisis yang bersifat dimensional secara cepat, yaitu desain dari aplikasi dan teknologi yang dapat mengoleksi, menyimpan, memanipulasi suatu data multidimensi untuk tujuan analisis. OLAP adalah bagian dari kategori yang lebih global dari pemikiran bisnis, yang juga merangkum hubungan antara pelaporan dan penggalian data. Aplikasi khusus dari OLAP adalah pelaporan bisnis untuk penjualan, pemasaran, manajemen pelaporan, manajemen proses bisnis (MPB), penganggaran dan peramalan, laporan keuangan dan bidang-bidang yang serupa. Istilah OLAP merupakan

perampingan dari istilah lama database OLTP (Online Transaction Processing).



Gambar 2.4. OLTP vs OLAP

Secara singkat OLAP adalah suatu proses yang digunakan untuk melakukan permintaan terhadap data dalam bentuk yang kompleks dan menganalisa data yang bervolume besar. maka dari itu OLAP seringkali disebut analisis data multidimensi.

Online Transaction Processing atau sering dikenal dengan OLTP adalah sistem yang berorientasi proses yang memproses transaksi secara langsung melalui komputer yang terhubung dalam jaringan.

Misalnya biasa digunakan pada sebuah supermarket, kasir menggunakan mesin dalam proses transaksinya. OLTP mempunyai

karakteristik beberapa user dapat creating, updating, retrieving untuk setiap record data, OLTP sangat optimal untuk updating data.

Secara singkat OLTP adalah segala penanganan dalam proses transaksi sehari-hari, misalkan sebuah toko Alfamart yang menangani transaksi pembelian setiap harinya, bayangkan ada berapa banyak transaksi input ke dalam database? pasti banyak sekali. lantas apakah data transaksi tersebut akan di dijamin menumpuk sampai berbulan-bulan bahkan bertahun-tahun? disinilah OLAP berperan.

Pada OLAP biasanya digunakan untuk pengambilan suatu keputusan, misalkan sebagai contoh para top level di Alfamart ingin melihat data transaksi perbulan, per 3 bulan atau bahkan per tahun untuk pengambilan keputusan barang apa saja yang paling laku dijual dan barang apa saja yang tidak laku di pasar.

2.3. PERBEDAAN OLAP DAN OLTP

OLTP (Online Transaction Processing)

Fungsi aplikasi OLTP merupakan satu aplikasi yang berguna untuk mendukung operasional perusahaan sehari-hari. Contohnya adalah aplikasi untuk merekam pesanan konsumen, merekam pembelian, merekam proses produksi, merekam penggajian karyawan dan lain sebagainya. Aktivitas pemrosesan transaksi, biasanya hanya melibatkan beberapa record. Sebagai contoh, dalam aplikasi untuk merekam pesanan konsumen, aplikasi akan mengakses data dari tabel Konsumen, tabel Pesanan Konsumen, tabel Detail Pesanan, tabel Persediaan dan tabel Karyawan. Setiap tabel memiliki kunci yang berguna untuk sarana relasi antar tabel. Aplikasi pesanan konsumen, hanya akan

menambahkan beberapa baris baru dalam tabel Pesanan Konsumen dan tabel Detail Pesanan.

Desain dalam OLTP bersifat Entity Relational atau databasenya dinormalisasi dulu sebelum digunakan. Database untuk mendukung OLTP adalah database yang normal. Normalized dengan banyak table. OLTP menggunakan data model ER untuk melihat transaksi sebagai proses model yang tunggal dan sinormalisaionel untuk menjagaintegritas data. OLTP menjadi bentuk yg lebih mudah untuk di ambildan di analisis. rancangan OLTP mengharuskan perancangan logika tabel dan tipe query secara bersama-sama agar tabel tersebut dapat terakses dengan cepat.

Berdasarkan sumber data Data Operasional, data OLTP adalah data asli. Dalam OLTP datanya adalah hari ini, update setiap saat (datanya dapat di-edit, di-ganti, atau di-hapus). Berdasarkan Tujuan datanya Untuk mengendalikan dan menjalankan tugas-tugas utama. Penggunaan OLTP digunakan setiap saat. Berdasarkan Data apa yang di tampilkan Bisnis proses yang berkelanjutan. Query yg digunakan Simple Query.

Kecepatan proses Pada dasarnya sangat cepat. Data diorganisasikan berdasarkan fungsi atau operasi seperti penjualan, produksi, dan pemrosesan pesanan. Space yg dibutuhkan Relativ kecil. Data bisa saja disimpan pada beberapa platform. Pemrosesan bersifat berulang

User Melayani banyak pemakai operasional User. Dalam OLTP, penggunaanya adalah IT Proffesional. Jumlah Pengguna Untuk OLTP penggunaanya adalah puluhan. Ukuran Database Ukuran database untuk OLTP sekitar MB-GB. I. Access OLTP aksesnya bisa write, read dan lain-lain. Jumlah rekaman yang di akses OLTP sekitar ratusan sampai ribuan. Unit Pekerjaan OLTP pekerjaannya hanya sederhana misalnya transaksi dalam swalayan. Seperti

misalnya kasir pada sebuah super market yang menggunakan mesin dalam proses transaksinya.

OLAP (On Line Analytical Processing)

Fungsi aplikasi OLAP digunakan untuk pengambilan keputusan. Aplikasi OLAP berfungsi untuk mengakses data yang besar (kalau perlu data selama empat atau bahkan lima tahun). OLAP berguna untuk menganalisis hubungan antar data guna menemukan ada tidak pola tertentu dalam data yang berguna untuk pengambilan keputusan di masa yang akan datang. Sebagai contoh, dengan menganalisis data di masa lalu, aplikasi OLAP berhasil mengidentifikasi produk apa yang disukai di daerah tertentu (ternyata ada perbedaan preferensi konsumen antar wilayah pemasaran).

Sedangkan database yang pas untuk mendukung OLAP adalah database yang denormalisasi. De-normalized dengan sedikit table dan menggunakan star / snowflake schemas. OLAP menggunakan dimensional model. perancangan dan penerapan dimensional model dilakukan pada data warehouse untuk di analisis oleh OLAP. Dimensional model merupakan permodalan data yang terdiri dari tabel dimensi dan tabel fact yg relasinya dapat di gambarkan pada star schema, tabel fact merupakan tabel utama dalam dimensional model yang berisi pengukuran nilai angka dari bisnis yang di simpan. tabel dimensi merupakan tabel perlengkapan dari tabel fact yg berisi penjelasan tekstuan dari bisnis. Keuntungan dari dimensional model ialah memisahkan rancangan logika tabel dengan tipe query yang di gunakan pengguna. keuntungan lainya pengguna dimensional model ialah kemudahan pengawasan terhadap penambahan data kemudian penambahan kolom dan rancangan baru serta menagani pergantian kebutuhan bisnis,

Berdasarkan sumber data Data konsolidasi, data OLAP di peroleh dari beberapa OLTP. datanya adalah sekarang dan hari ini yang berguna untuk melakukan analisis ke depan. Berdasarkan Tujuan datanya Untuk membantu dalam perencanaan, memecahkan masalah dan mendukung keputusan. Untuk mendukung keputusan harian (operasional). biasanya OLAP digunakan seperlunya saja.

Berdasarkan Data apa yg di tampilkan Menampilkan data dari berbagai macam aktivitas bisnis. Berorientasi pada transaksi. Query yg digunakan Complex Queries. Kecepatan proses Tergantung dari data yg dilibatkan, proses akan lebih cepat dengan menggunakan fungsi indexing. Space yg dibutuhkan Lebih besar, karena membutuhkan lebih banyak indexing dibandingkan OLTP.

Kemampuan OLAP Konsolidasi melibatkan pengelompokan data. Sebagai contoh kantor-kantor cabang dapat dikelompokkan menurut kota atau bahkan propinsi. Transaksi penjualan dapat ditinjau menurut tahun, triwulan, bulan, dan sebagainya. Kadangkala istilah rollup digunakan untuk menyatakan konsolidasi j. User OLAP penggunaanya adalah Knowledge worker maksudnya penggunaanya adalah seorang yang bertindak dalam subyek tertentu, atau petinggi dalam suatu perusahaan. Jumlah Pengguna Untuk OLAP penggunaanya bisa sampai ratusan bahkan ribuan.

Ukuran Database Ukuran database untuk OLAP bisa sampai GB- I. Access aksesnya OLAP sering dibaca karena digunakan untuk analisa. Jumlah rekaman yang di akses OLAP data yang diakses bisa sampai jutaan bahkan milyaran.

Unit Pekerjaan Untuk OLAP query untuk menampilkan data sangat kompleks. yang merupakan bagian dari Business Intelligence yang berguna untuk menyediakan laporan analisis, seperti penjualan, marketing, BPM (Business Process Management), budgeting, forecasting,

keuangan dan sebagainya. model multidimensi, yang mengijinkan query analisis yang kompleks dengan kecepatan eksekusi yang tinggi.

2.4. ENTITY RELATIONSHIP MODEL

Menurut (Connolly & Carolyn, 2002), diperlukan *Entity Relationship Model* untuk menggambarkan sebuah basis data. Dalam *E-R Model*, terdapat kumpulan entitas dan relasi. Entitas adalah kumpulan objek yang bersifat sama. Entitas menggambarkan tabel-tabel di dalam basis data. Sedangkan relasi menggambarkan hubungan antara satu tabel dengan tabel lainnya. *E-R Model* digambarkan dengan sebuah diagram yang disebut *Entity Relationship Diagram* (ERD). Simbol – simbol diagram ER dapat dilihat pada **Error! Reference source not found..** Komponen-komponen utama dalam ERD, yaitu:

1. Entitas

Entitas menggambarkan kumpulan objek yang ada di dunia nyata dengan sifat yang sama. Sebuah entitas dapat berbentuk objek dengan keberadaan fisik ataupun abstrak. Setiap entitas disimbolkan dengan sebuah persegi panjang yang diberi nama entitas tersebut. SS

2. Relasi

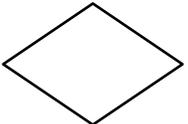
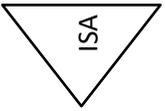
Tipe relasi adalah sekumpulan hubungan antar tipe entitas yang memiliki makna. Tipe relasi disimbolkan dengan jajaran genjang yang diberi nama pada bagian dalamnya.

3. Atribut

Atribut adalah bagian sebuah entitas atau relasi. Atribut menampung nilai yang menjelaskan setiap entitas dan bagian utama dari

data yang disimpan pada basis data. Atribut disimbolkan dengan elips yang diberi nama atribut di bagian dalamnya.

Tabel 2.1. Simbol-simbol diagram ER (Kadir, 1999)

Komponen	Keterangan	Simbol
Himpunan Entitas	Entitas adalah objek yang dapat didefinisikan dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat	
Atribut (Kunci Primer)	Elemen dari suatu entitas yang merupakan atribut kunci (<i>primary key</i>)	
Atribut	Atribut berfungsi untuk mendeskripsikan karakter entitas. Pada setiap E-R Diagram bisa terdapat lebih dari satu atribut. Isi atribut memiliki sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain.	
Garis (Konektor)	Elemen yang menghubungkan entitas dengan relasi dan atribut.	
Relasi	Relasi merupakan hubungan antar entitas satu dengan yang lain. Dalam sebuah relasi harus dibedakan antara hubungan antar entitas dengan isi dari hubungan itu sendiri.	
Relasi ISA (Silberschatz, 2011)	Relasi yang menjelaskan hubungan seperti <i>super class</i> dan <i>sub class</i> pada sebuah entitas.	

(Nugroho, 2011) menyatakan ERD digunakan untuk membuat model basis data yang menggambarkan struktur dan batasan suatu basis data. Simbol dan keterangan komponen ERD adalah sebagai berikut:

- a. Persegi panjang, menggambarkan suatu entitas.
- b. Elips, menggambarkan atribut.
- c. Belah ketupat, menggambarkan relasi antar entitas.
- d. Garis, menggambarkan hubungan antar atribut.

4. Kardinalitas

Kardinalitas adalah tingkat hubungan yang menjelaskan banyak tidaknya hubungan antar entitas. Dan menjelaskan batasan jumlah entitas yang berhubungan di suatu relasi. Kardinalitas mempunyai tiga jenis, yaitu:

- a. *One to one* (1:1)

Tingkat hubungan pada entitas pertama hanya mempunyai satu hubungan pada entitas yang kedua.

- b. *One to many / many to one* (1:n / n:1)

Tingkat hubungan pada entitas pertama memiliki banyak hubungan kejadian pada entitas yang kedua dan sebaliknya.

- c. *Many to many* (n:n)

Tingkat hubungan pada entitas pertama memiliki banyak hubungan pada entitas yang kedua.

Unified Modeling Language (UML)

Menurut (Badriyah, 2007) *Unified Modelling Language* (UML) adalah notasi lengkap untuk membuat gambaran model suatu sistem. UML

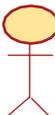
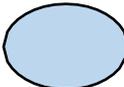
memungkinkan deskripsi sistem dibuat secara rinci pada setiap level. Notasi tersebut akan menjelaskan sistem dengan arsitektur yang berorientasi objek, namun juga permodelan spesifikasi, visualisasi, konstruksi dan dokumentasi proses sistem secara intensif. UML memiliki dua jenis diagram, yaitu diagram *structural* dan diagram perilaku. Diagram *structural* menggambarkan bagian statis dari sistem. Diagram perilaku menggambarkan bagian yang dinamis dari sistem. Pada UML yang akan digunakan dalam penelitian ini adalah *use case diagram* dan *activity diagram*. Penjelasannya sebagai berikut:

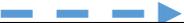
Use case diagram

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem. Komponen *use case* dapat dilihat pada

Tabel 1.2.

Tabel 1.2. Komponen use case diagram (Al-Fatih, 2012)

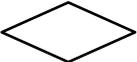
Simbol	Nama Simbol	Fungsi
	<i>Actor</i>	Menggambarkan aktor yang berinteraksi dengan sistem untuk bertukar informasi.
	<i>Use Case</i>	Menggambarkan sistem <i>use case</i> .

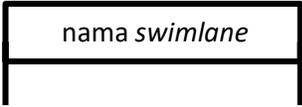
Simbol	Nama Simbol	Fungsi
	<i>Association Relationship</i>	Menggambarkan hubungan antara <i>use case</i> dengan <i>actor</i> .
	<i>Include Relationship</i>	Mengidentifikasi hubungan antar <i>use case</i> di mana yang satu memanggil yang lain.
	<i>Extend Relationship</i>	Mengidentifikasi pemanggilan fungsi dengan kondisi tertentu.
	<i>Generalization Relationship</i>	Mendefinisikan relasi antara dua <i>actor</i> atau dua <i>use case</i> yang mana salah satunya menurunkan sebuah fungsi.

Activity diagram

Diagram ini menjelaskan aliran dari aktivitas satu ke aktivitas lainnya dalam sistem. Diagram ini penting dalam pemodelan fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

Tabel 2.3. Komponen activity diagram (Al-Fatih, 2012)

Simbol	Deskripsi
Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>Decision</i> 	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu.

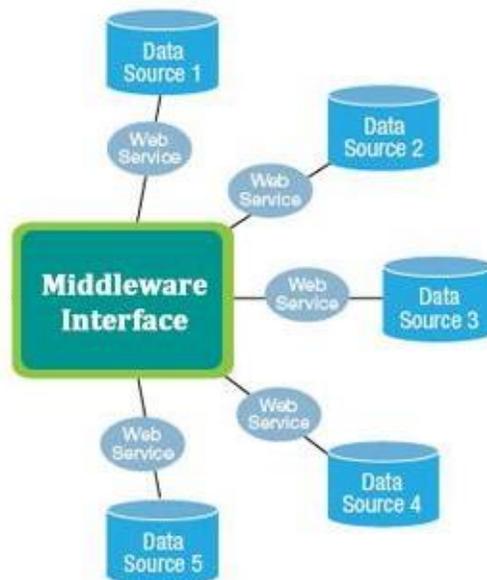
Simbol	Deskripsi
Percabangan / <i>Join</i> 	Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu.
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang dilakukan.
<i>Fork</i> 	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel.
<i>Join</i> 	Digunakan untuk menunjukkan kegiatan yang digabungkan.

BAB 3

MIDDLEWARE

3.1. KONSEP MIDDLEWARE

Middleware adalah perangkat lunak komputer yang memberikan layanan untuk menghubungkan bagian-bagian berbeda dari sebuah aplikasi dengan sistem operasi. Middleware umumnya digunakan dalam sistem terdistribusi untuk memudahkan pengembang perangkat lunak dalam melakukan komunikasi input/output.



Gambar 3.1. Struktur Middleware

Middleware bisa diibaratkan menjadi beberapa bentuk, pertama middleware seperti semacam lem yang dapat menyatukan sebuah jaringan dan komputer-komputer yang terhubung di dalamnya. Middleware bisa berupa sebuah aplikasi tunggal atau dapat berupa keseluruhan server.

3.2. TUJUAN PENGGUNAAN MIDDLEWARE

Middleware dapat digunakan untuk beberapa tujuan seperti:

1. Memberikan fasilitas untuk programmer supaya dapat mendistribusikan obyek akan dipakai pada beberapa bagian proses yang berbeda.
2. Sebagai interkoneksi ke beberapa aplikasi dan masalah interoperabilitas.

Middleware menjadi sangat dibutuhkan untuk bermigrasi dari aplikasi mainframe ke aplikasi client/server dan juga untuk menyediakan komunikasi antar platform yang berbeda.

3.3. MANFAAT MENGGUNAKAN MIDDLEWARE

Middleware dapat memberikan manfaat tertentu seperti penjelasan dibawah ini:

1. Pada sistem yang terdistribusi maka dapat dijalankan 2 buah platform atau aplikasi secara bersamaan.

2. Dapat melakukan komunikasi pada aplikasi yang berjalan di platform berbeda.
3. Adanya transparansi pada seluruh jaringan sehingga dapat menyediakan interaksi dengan layanan atau aplikasi lainnya.

Dalam dunia teknologi informasi Middleware menjadi salah satu software yang dibuat untuk menghubungkan beberapa proses pada satu atau lebih mesin agar bisa saling berinteraksi di suatu jaringan.

3.4. CONTOH LAYANAN MIDDLEWARE

1. Transaction Monitor
 - a. Produk pertama yang disebut middleware
 - b. Menempati posisi antara permintaan dari program client dan database, untuk meyakinkan bahwa semua transaksi ke database terlayani dengan baik.
2. Messaging Middleware
 - a. Menyimpan data dalam suatu antrian message jika mesin tujuan sedang mati atau overloaded
 - b. Mungkin seperti sistem messageng email, kecuali messaging middleware digunakan untuk mengirim data antar aplikasi
3. Distributed Object Middleware
 - a. Menyediakan antarmuka antara sebuah query dengan beberapa database yang terdistribusi.

Contoh : JDBC, ODBC, dan ADO.NET.

4. Application Server Middleware

J2ME Application Server, Oracle Application Server.

3.5. TUJUAN UMUM MIDDLEWARE

Tujuan utama middleware adalah untuk membantu memecahkan interkoneksi beberapa aplikasi dan masalah interoperabilitas. Middleware sangat dibutuhkan untuk bermigrasi dari aplikasi mainframe ke aplikasi client/server dan juga untuk menyediakan komunikasi antar platform yang berbeda.

Perangkat lunak ini terdiri dari serangkaian pelayanan yang mengizinkan bermacam-macam proses berjalan dalam satu atau lebih mesin untuk dapat saling berinteraksi satu sama yang lainnya. Lambat laun teknologi ini menyediakan kemampuan interoperabilitas yang mendukung pada perpindahan ke arsitektur distribusi yang berhubungan, yang biasanya sering digunakan untuk mendukung dan menyederhanakan kerumitan, aplikasi terdistribusi. Termasuk didalamnya, web server, aplikasi server dan peralatan sama yang mendukung pengembangan dan pengantaran aplikasi.

Middleware secara khusus menjadi bagian dari teknologi informasi modern berbasis XML, SOAP, web service dan pelayanan berbasis arsitektur. Middleware berada diantara aplikasi perangkat lunak yang mungkin bekerja pada system operasi yang berbeda. Middleware serupa dengan middle layer dari sebuah tiga baris sistem arsitektur tunggal, kecuali usahanya melewati bermacam-macam system atau aplikasi. Contohnya perangkat lunak EAI (*Enterprise Application*

Integration), perangkat lunak telekomunikasi, monitor transaksi dan perangkat lunak pemesanan dan antrian.

3.6. LINGKUNGAN KOMPUTASI

Pelayanan middleware menyediakan banyak set fungsi dari aplikasi antarmuka pemrograman yang mengizinkan sebuah aplikasi untuk:

1. Menemukan tempat melewati jaringan secara transparan sehingga dapat menyediakan interaksi dengan service atau aplikasi lainnya.
2. Mandiri dari service jaringan.
3. Dapat dipercaya dan selalu tersedia.

Middleware menawarkan beberapa keuntungan unik dari teknologi untuk bisnis dan industri. Sebagai contoh, sistem database tradisional biasanya diletakkan dalam lingkungan yang dekat dimana pengguna mengakses sistem menggunakan jaringan terbatas atau intranet. Dengan perkembangan fenomena dari **World Wide Web**, pengguna dapat mengakses database secara virtual dengan berbagai macam jenis akses dari belahan dunia manapun. Middleware mengamati masalah dari berbagai level interoperabilitas diantara struktur database yang berbeda. Middleware memfasilitasi akses transparan untuk melegalkan sistem manajemen database (DBMS) atau aplikasi lewat sebuah web server tanpa memperhatikan karakteristik spesifik database.

Perusahaan bisnis sering menggunakan aplikasi middleware untuk menghubungkan informasi dari database departemen, misalnya daftar pembayaran, penjualan, dan penghitungan atau database house

dalam lokasi geografi yang bermacam-macam. Dalam tingginya kompetisi komunitas kesehatan, laboratorium membuat luas penggunaan dari aplikasi middleware untuk data mining, sistem informasi laboratorium (LIS) cadangan, dan untuk menggabungkan sistem selama proses penggabungan dua rumah sakit. Middleware menolong menjembatani jarak pemisah antara LIS dalam bentuk baru jaringan kesehatan mengikuti proses pembelian rumah sakit. Pengembang jaringan wireless dapat menggunakan middleware untuk menghadapi tantangan penggabungan dengan sensor jaringan wireless (WSN) atau teknologi WSN.

Pengimplementasian sebuah aplikasi middleware mengizinkan pengembang middleware untuk menyatukan sistem operasi dan perangkat keras dengan berbagai macam aplikasi yang tersedia. Middleware dapat menolong pengembang perangkat lunak menghindari penulisan antarmuka program aplikasi (API) untuk setiap pengendali program, dengan cara melayani sebagai sebuah antarmuka pemrograman yang berdiri sendiri untuk setiap aplikasi yang dibuat.

Database middleware yang paling umum digunakan adalah ODBC (Open DataBase Connectivity). Keterbatasan ODBC adalah bahwa middleware ini didisain untuk bekerja pada tipe penyimpanan relational database. Database middleware yang lain, yang merupakan superset daripada ODBC adalah OLEDB. OLEDB bisa mengakses hampir segala macam bentuk database, kelebihan yang lain dari OLEDB adalah dia didisain dengan konsep obyek komponen (Component Object Model)

yang mengandalkan object-oriented computing dan menjadi salah satu trend di dunia komputasi.

Beberapa produk database middleware yang bisa disebutkan di sini adalah Oracle's DB Integrator (previously DIGITAL's DB Integrator), Sybase's Omni CONNECT, and International Software Group's Navigator. Kelebihan dari produk-produk ini dibandingkan dengan standard seperti ODBC dan OLEDB adalah performance, yang sangat sulit dimiliki oleh suatu produk yang mengacu pada standar.

BAB 4

ARSITEKTUR APLIKASI

4.1. KONSEP ARSITEKTUR APLIKASI

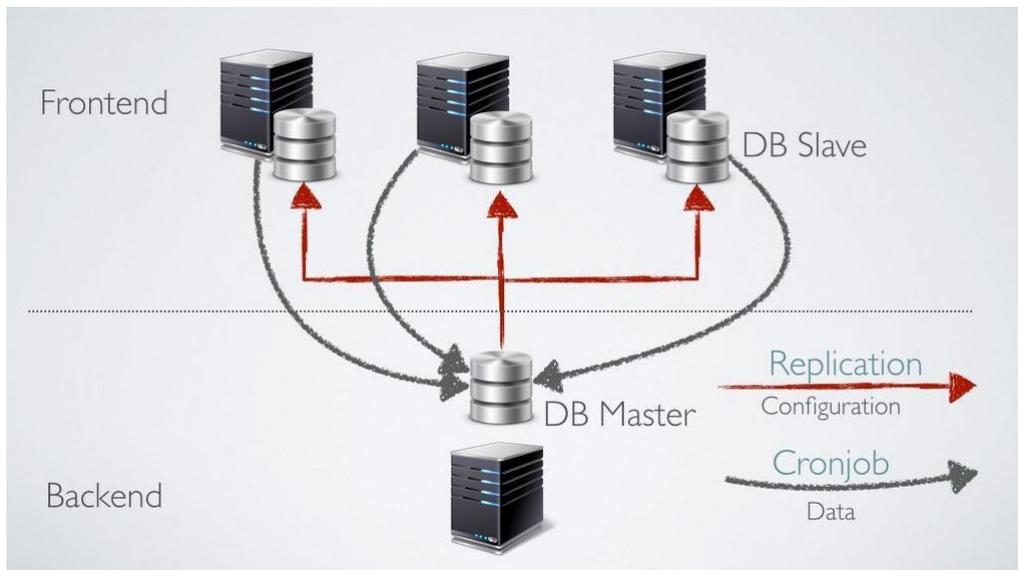
Arsitektur aplikasi adalah sebuah teknologi spesifikasi yang akan digunakan untuk mengimplementasikan sistem informasi. Arsitektur aplikasi menjadi suatu desain aplikasi yang terdiri dari komponen-komponen yang saling berinteraksi antara satu dengan yang lain. Arsitektur aplikasi disebut juga dengan infrastuktur aplikasi. Selain itu, arsitektur aplikasi juga merupakan cara komunikasi komponen-komponen melalui network atau jaringan yang saling terhubung.

Dalam kaitannya dengan GRMS (*Government Resource Management System*), arsitektur aplikasi menggambarkan bisnis proses yang berjalan. Sebuah arsitektur mendeskripsikan bagaimana sistem dibangun, termasuk bagaimana macam – macam komponen di dalamnya. Dewasa ini, arsitektur aplikasi biasanya digunakan sebagai petunjuk operasi sekarang dan sebagai blueprint untuk langkah kedepan.

Karakteristik pada setiap aplikasi berbeda-beda, ada yang sebagian besar loadnya berasal dari traffic untuk membaca konten (proses baca database) dibandingkan dengan meng-update konten (proses tulis database) seperti Wikipedia. Dan ada juga aplikasi yang load proses tulis

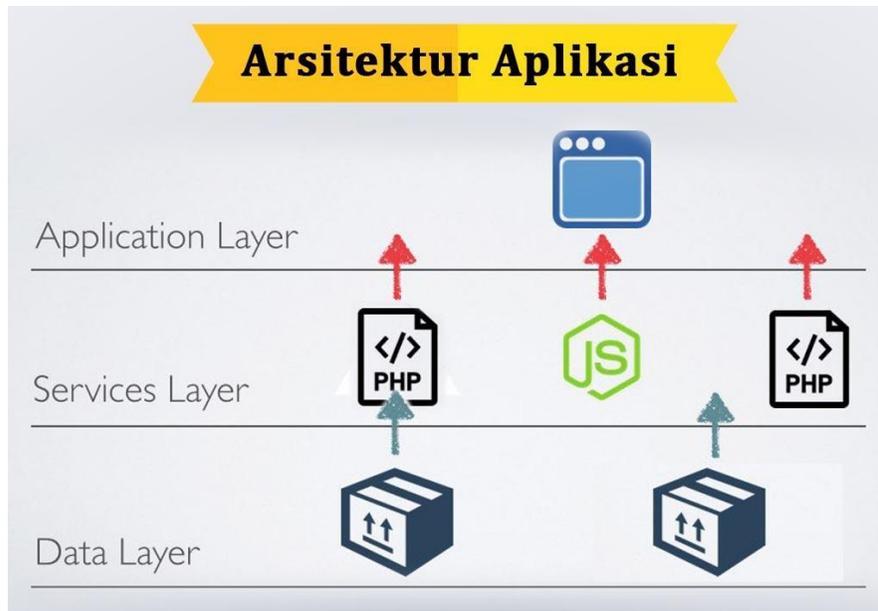
databasenya besar seperti pada aplikasi Revive atau pernah dikenal dengan nama OpenX dan OpenAds, adalah aplikasi ad management yang berbasis Php dan Mysql. Menyediakan fitur-fitur untuk mengelola inventori iklan di website, berikut dengan statistik Impression, Click, CPC dan fitur lainnya.

Pada Revive Load pada aplikasi akan lebih besar untuk proses pencatatan log dan statistik (proses tulis database), ketimbang proses membaca inventory iklan (proses baca database). Bahkan proses yang terjadi di aplikasi ini mencapai n kali dari proses website utama. Apabila kita memiliki 5 banner dalam satu halaman static yang memiliki real time user sebanyak 1000 request, maka load di aplikasi ad management ini sebesar $5 \times 1000 = 5000$ request, atau sebanyak 5x dari halaman tersebut.



Gambar 4.1. contoh arsitektur dari revive

Dan Pada GRMS ini arsitektur aplikasi bisa kita bagi kedalam 3 layer seperti pada (Gambar 2.xx). Pada rancangan arsitektur aplikasi e-government ini terdiri dari data layer, service layer, dan application layer.



Gambar 4.2. Arsitektur Aplikasi

Pada *layer* yang pertama adalah *data layer*, pada section ini database yang digunakan dalam pembangunan aplikasi grms ini yaitu *postgresql versi 10.5 (stable version)*. Dan pada *layer / section* selanjutnya yaitu *service layer*, *service layer* terdiri dari *php (Hypertext Preprocessor)* dan *javascript*. Dan yang terakhir yaitu *application layer*, *application layer* adalah *layer* yang menyediakan *interface* atau antarmuka dari setiap aplikasi yang berjalan.

4.2. BAHASA PEMROGRAMAN

➤ HTML

Menurut (Powers, 2012), *Hypertext Markup Language* atau HTML adalah bahasa *markup*. Bahasa *markup* adalah bahasa yang digunakan untuk menerjemahkan Informasi dari *web browser* tentang struktur halaman, konten dan sebagainya. Jadi, HTML adalah sebuah dokumen teks yang berisi Informasi dari *web browser*. Perintah-perintah yang ada di dokumen HTML diawali dengan *tag* pembuka `<html>` dan diakhiri dengan *tag* penutup `</html>`.

Bahasa pemrograman *web* terus mengalami perkembangan. Versi yang terbaru adalah HTML5. Dalam HTML5 terdapat kelas-kelas di dalam *web* sehingga aplikasi yang dibuat menjadi terlihat lebih nyata (Kessin, 2012). HTML5 didukung dengan javascript, API *browser*, dan AJAX sehingga memudahkan dalam membuat aplikasi. HTML5 membuat aplikasi *web* menjadi lebih interaktif pada *browser* dari sisi pengguna dengan adanya fitur-fitur multimedia dan grafis.

➤ PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan untuk membuat *server* menghasilkan keluaran yang dinamis dan ditanam pada bahasa HTML. Dalam penggunaannya, halaman *web* dihasilkan dengan kombinasi dari bahasa PHP, HTML, javascript dan beberapa perintah MySQL. PHP menggunakan beberapa jenis *tag*, *tag* pembuka dan penutup yang standar yaitu `<?php ?>`. Setiap perintah dalam dokumen PHP harus diakhiri dengan tanda titik koma (;). Dokumen

PHP disimpan dengan ekstensi *.php. Dokumen PHP dapat mengambil dokumen HTML seperti index.html disimpan dengan index.php dan menghasilkan halaman *web* yang sama dengan aslinya.

➤ **CSS3**

Cascading Sytle Sheet atau disingkat CSS merupakan bahasa pemrograman yang ditanam pada halaman HTML dan berjalan pada *browser*. (Mansfield, 2005). CSS memungkinkan desain halaman *web* terlihat lebih baik dan efektif. CSS memungkinkan pemisahan desain dengan konten ketika aplikasi *web* dibangun karena keterbatasan dalam HTML mengelola tampilan. CSS mengelola desain *web* menjadi halaman visual yang menarik. Dokumen CSS dapat dibuat terpisah dari halaman HTML dengan ekstensi *.css.

CSS menawarkan beberapa keuntungan, yaitu menampilkan tampilan yang lebih baik dari HTML, menghemat *tag* yang digunakan di dalam dokumen HTML untuk pengaturan tampilan, mempermudah dalam proses memperbaiki tampilan halaman dengan dokumen CSS yang terpisah. (Mc Farland, 2013).

CCS3 adalah versi terbaru dari CSS. CSS3 terdiri dari beberapa modul yang menawarkan fitur baru, yaitu CSS3 *selectors*, pemilihan warna, nilai dan unit, *media*, *background*, *teks*, *huruf*, *gradien*, *multi kolom*, *transformasi*, *transisi*, *animasi*, dan *flexible box*.

➤ **Javascript**

Menurut (Nixon, 2009), javascript adalah bahasa pemrograman yang berjalan sepenuhnya di dalam *web browser*. Pemrograman javascript membuat halaman *web* lebih dinamis dan interaktif pada *web browser*, misalnya ketika pengguna mengarahkan *mouse item* dalam *browser*, melihat teks baru, warna, gambar yang muncul di halaman *web* atau mengambil sebuah objek pada halaman dan menariknya ke lokasi baru. Pemrograman javascript dipanggil dengan *tag* `<script> </script>` dalam dokumen HTML. Dalam *tag script* perintah pada baris kode javascript setara dengan perintah *echo* atau *document.write* pada PHP. Perintah javascript diakhiri dengan tanda titik koma seperti pada PHP.

➤ **MySQL**

Menurut (Nixon, 2009), MySQL adalah basis data yang paling banyak digunakan dalam pemrograman *web* dengan bahasa pemrograman PHP. SQL pada MySQL adalah *Stuctured Query Language*, yaitu bahasa terstruktur yang menjelaskan permintaan untuk mengakses dan mengelola basis data. MySQL sangat *scalable*, yang berarti dapat berkembang sesuai aplikasi *web* yang dibuat. Tujuan MySQL dibuat adalah untuk memungkinkan permintaan sederhana dari basis data melalui perintah DDL maupun DML.

➤ **Framework CodeIgniter**

Framework adalah kumpulan kode program yang merupakan pondasi dan bersifat universal yang digunakan *programmer* atau *software engineer* untuk membangun aplikasi perangkat lunak. Berbeda dengan *class library* yang dibutuhkan oleh para *programmer* untuk dipanggil dan dikendalikan olehnya, *framework*-lah yang nantinya mengendalikan alur *software* secara keseluruhan. Sifat itu disebut *Inversion of Control*. *Framework* mempunyai perilaku dasar (*default behavior*) di mana kebutuhan umum diatasi/dikerjakan oleh *framework*. *Framework* juga mempunyai ruang-ruang kosong yang dapat diisi oleh kode-kode tertentu sebagai pengembangan dari perilaku dasar. Dengan kata lain, *framework* mempunyai kemampuan untuk dikembangkan oleh penggunanya (*extensibility*) (Basuki, 2010)

CodeIgniter merupakan salah satu *web application framework* yang bersifat *open source* untuk bahasa pemrograman PHP. Tujuan utama pengembangan. *CodeIgniter* adalah untuk membantu *developer* untuk mengerjakan aplikasi lebih cepat daripada menulis semua *code* dari awal. *CodeIgniter* menyediakan berbagai macam *library* yang dapat mempermudah dalam pengembangan. *CodeIgniter* diperkenalkan pada publik tanggal 28 Februari 2006. *CodeIgniter* dibangun menggunakan konsep *Model-View-Controller development pattern* (Basuki, 2010).

➤ **Bootstrap**

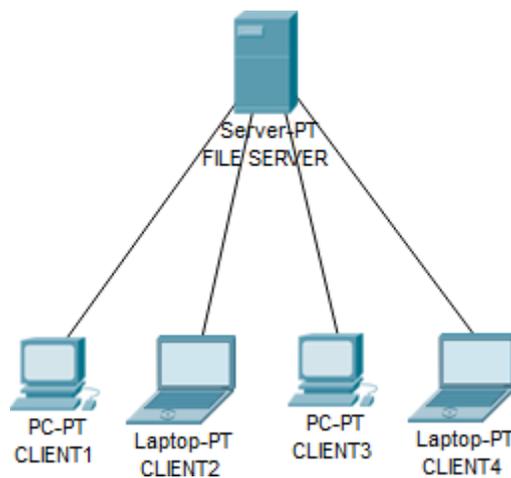
Bootstrap adalah suatu *library framework* css yang dibuat khusus untuk bagian pengembangan *front-end website*. Bootstrap merupakan sebuah *framework* css yang memudahkan pengembang untuk membangun *website* yang menarik dan responsif. Bootstrap telah menyediakan kumpulan komponen class interface dasar yang telah dirancang sedemikian rupa untuk menciptakan tampilan yang menarik, bersih dan ringan. Selain komponen *class interface*, bootstrap juga memiliki fitur *grid* yang berfungsi untuk mengatur *layout* pada halaman *website* yang bisa digunakan dengan sangat mudah dan cepat. Dengan menggunakan bootstrap kita juga di beri keleluasaan dalam mengembangkan tampilan *website* yang menggunakan bootstrap yaitu dengan cara mengubah tampilan bootstrap dengan menambahkan *class* dan CSS. Bootstrap dapat dikembangkan dengan tambahan lainnya karena ini cukup fleksibel terhadap pekerjaan *design* yang dibutuhkan (Otto, 2011).

4.3. TOPOLOGI

Client – Server

Server adalah komputer database yang berada di pusat, dimana informasinya dapat digunakan bersama-sama oleh beberapa user yang menjalankan aplikasi di dalam komputer lokalnya yang disebut dengan Client. (Suwandi, 2010)

Server didukung dengan prosesor yang bersifat scalable dan RAM yang besar, juga dilengkapi dengan sistem operasi khusus, yang disebut sebagai sistem operasi jaringan (network operating system). Server juga menjalankan perangkat lunak administratif yang mengontrol akses terhadap jaringan dan sumber daya yang terdapat di dalamnya, seperti halnya berkas atau alat pencetak (printer), dan memberikan akses kepada workstation anggota jaringan. (idwebhost, 2018). Berikut contoh sederhana penerapan client – server pada suatu jaringan.

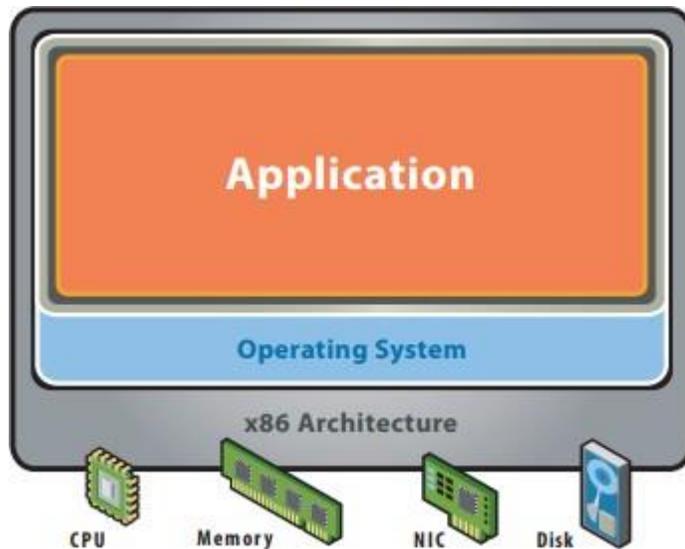


Gambar 4.3. Konsep client –server

4.4. VIRTUALISASI

➤ Virtual Machine

Sebelum adanya teknologi virtualisasi, resource yang ada pada perangkat keras suatu mesin hanya dapat menjalankan 1 Sistem Operasi fisik.

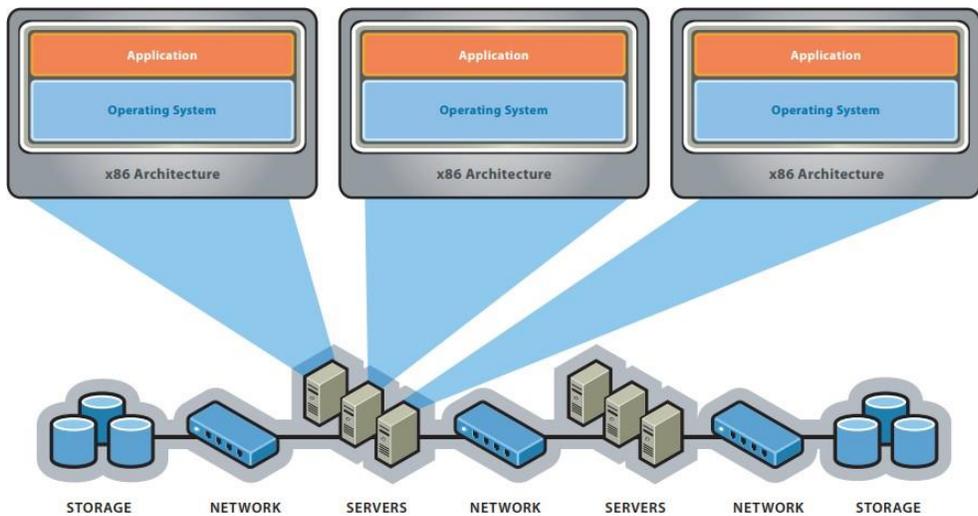


Gambar 4.4. Layer Sebelum ada Virtualisasi (VMware, 2006)

Berikut adalah kondisi Sebelum

- 1 Sistem Operasi per mesin.
- Perangkat lunak dan perangkat keras digabungkan secara langsung.
- Menjalankan beberapa aplikasi pada mesin yang sama seringkali menciptakan konflik.
- Sumber daya yang kurang dimanfaatkan
- Infrastruktur yang tidak fleksibel dan mahal

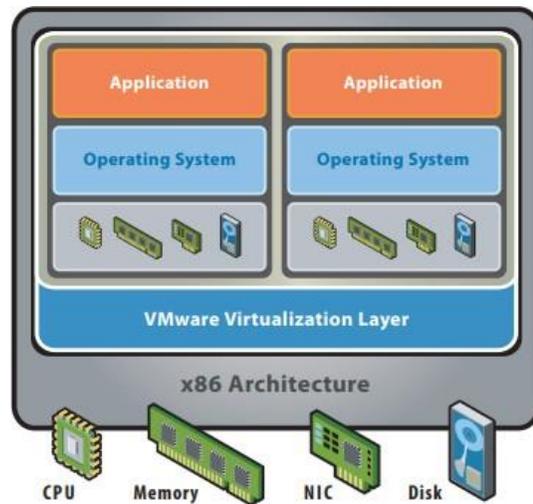
Pada konsep infrastruktur tradisional berdasarkan penerapan teknologi pada gambar 4.4. yang di implementasikan pada gambar 4.5. dengan skala yang besar dapat di lihat bahwa 1 server fisik hanya dapat menghandle 1 Sistem Operasi dengan menggunakan teknologi shared storage



Gambar 4.5. *Traditional Infrastructure* (VMware, 2006)

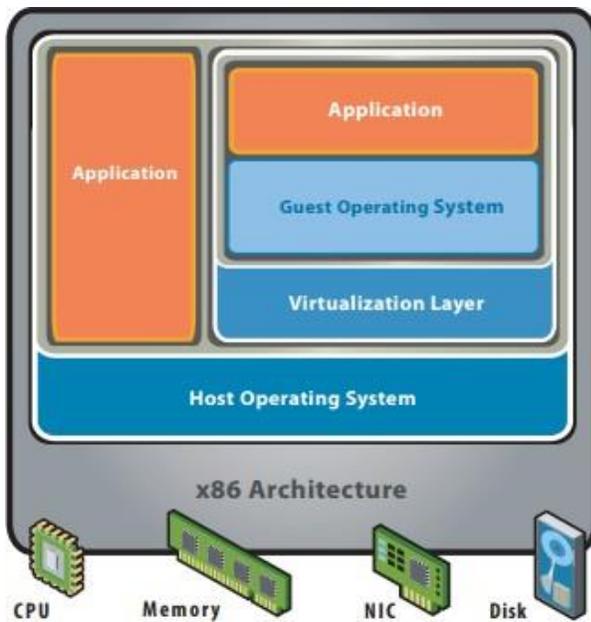
Seiring berjalannya waktu teknologi semakin berkembang, teknologi virtualisasi semakin dikenal para praktisi pusat pengelola data center. Di mana dalam teknologi virtualisasi dapat menggunakan resource yang dimiliki pada server fisik dapat dibagi menjadi beberapa mesin virtual dengan alokasi CPU, Memory, Storage & Device I/O yang berbeda yang dapat di gambarkan pada gambar 4.5.

Pada gambar 4.6., level hardware memiliki resource CPU, Memory, Disk/Storage serta I/O device (NIC). Satu level di atasnya *Virtualization Layer* memiliki tugas untuk melakukan proses *read & hardware resource management* yang ada pada level hardware. *Virtualization Layer* dapat membagi resource yang ada pada level hardware menjadi beberapa *virtual cluster hardware* dengan alokasi CPU, Memory, Disk/Storage & NIC yang berbeda-beda. Hasil dari *virtual cluster hardware* memungkinkan beberapa Sistem Operasi beserta aplikasi berjalan pada 1 hardware.



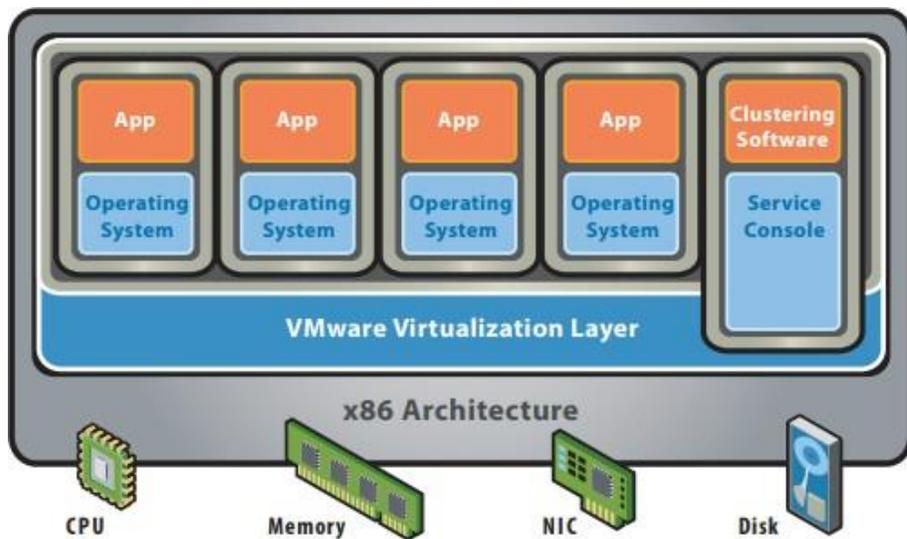
Gambar 4.6. *Virtualization Infrastructure Architecture*

Service virtualisasi dapat berjalan dalam 2 jenis service, yaitu Hosted Architecture (Gambar 4.6.) & Hypervisor Architecture (Gambar 4.7.). Pada model hosted architecture, virtualisasi berjalan pada Level aplikasi dalam suatu Sistem Operasi yang di install pada suatu mesin. Sedangkan pada model Hypervisor Architecture, virtualisasi berjalan secara langsung di atas layer *hardware resource* untuk melakukan manajemen penggunaan virtual *hardware resource* terhadap *physical hardware resource*.



- *Installs and runs as an application*
- *Relies on host OS for device support and physical resource management*

Gambar 4.7. Hosted Architecture (VMware,

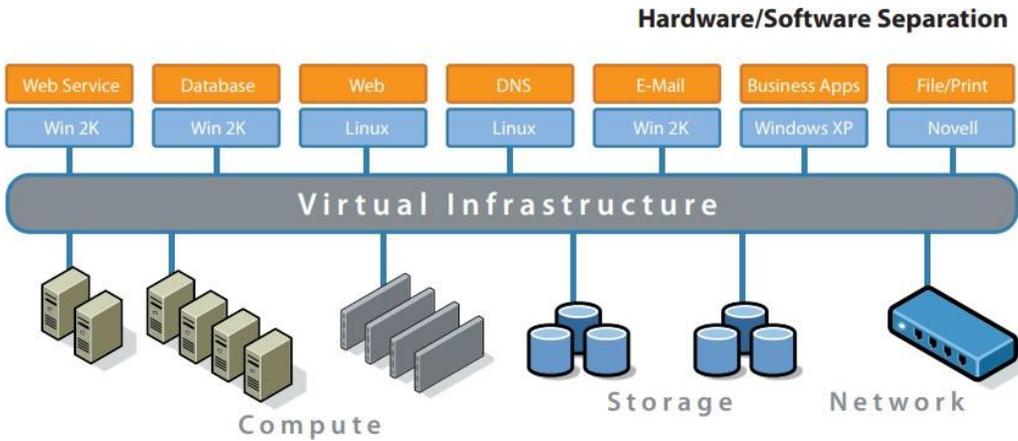


Gambar 4.8. Bare-Metal (Hypervisor) Architecture

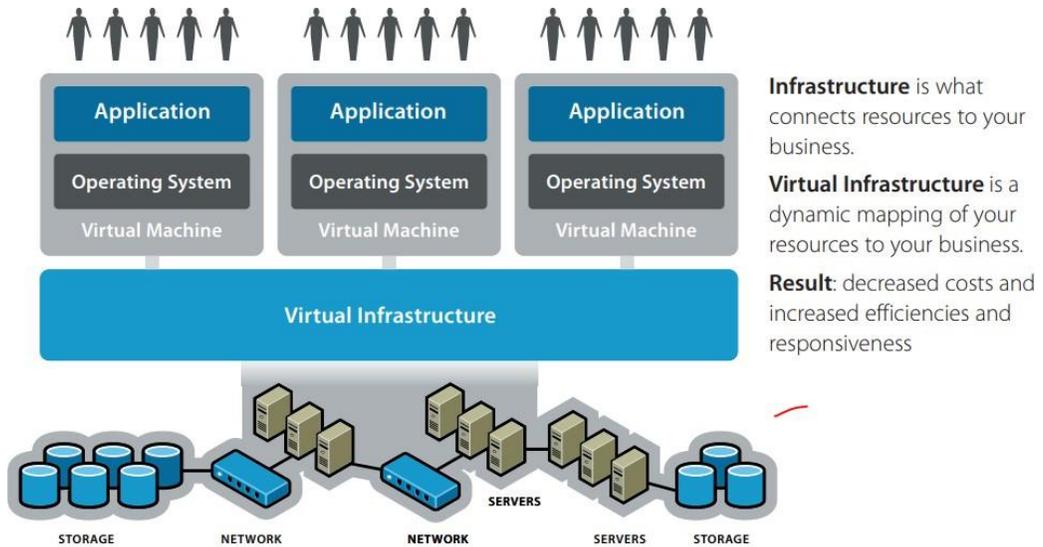
- Lean virtualization-centric kernel

- Service Console for agents and helper applications

Pada skala yang lebih besar penerapan virtualisasi dapat dilihat pada gambar 4.9..



Gambar 4.9. Virtual Infracrucure (VMware, 2006) (1)



Transforms farms of individual x86 servers, storage, and networking into a pool of computing resources

Gambar 4.10. Virtual Infrastructure (VMware, 2006) (2)

Keuntungan dan Kerugian menggunakan Virtualisasi

Menurut (Hagen, 2008) berikut keuntungan dan kerugian menggunakan virtualisasi:

1. Keuntungan

- a. Menggunakan perangkat keras yang ada dengan lebih baik
- b. Mengurangi Harga Perangkat Keras
- c. Mengurangi Infrastruktur IT
- d. Menyederhanakan Sistem Administrasi
- e. Meningkatkan Uptime Dan Mempercepat Failure Recovery
- f. Menyederhanakan Ekspansi Kapasitas
- g. Lebih sederhana dalam dukungan perangkat lunak asli
- h. Menyederhanakan Pengembangan System-Level
- i. Menyederhanakan Instalasi dan Deployment Sistem
- j. Menyederhanakan sistem dan Testing Aplikasi

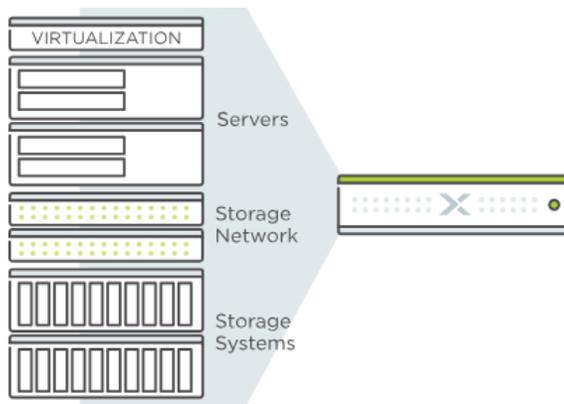
2. Kerugian

- a. Satu Titik Kesalahan Terpusat Pada Server Fisik
- b. Kepadatan Saluran Jaringan
- c. Menaikkan Kompleksitas Jaringan dan Waktu Debug
- d. Menaikkan kompleksitas Administrasi

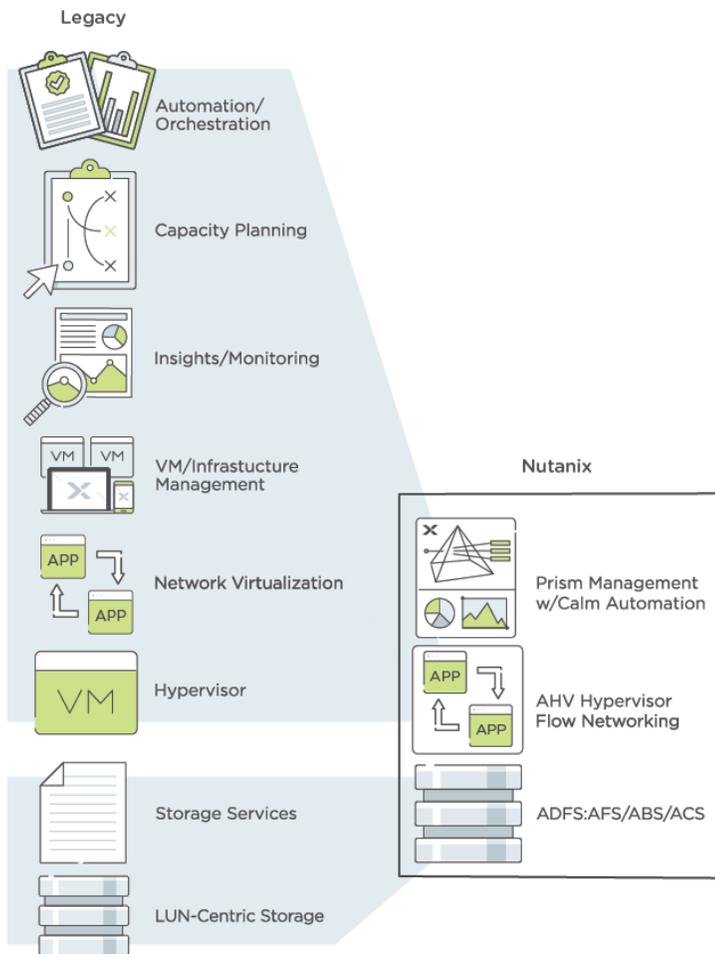
Hyperconverged Infrastructure (HCI)

Server

Hyperconverged infrastructure (Nutanix, 2018) adalah sebuah teknologi yang merampingkan penyebaran, manajemen dan penskalaan sumber daya pusat data dengan menggabungkan server berbasis x86 dan sumber daya penyimpanan dengan intelligent software. Server yang terpisah, storage network, dan storage array dapat diganti dengan satu solusi hyperconverged mulai dari skala kecil - enterprise.



Gambar 4.11. Virtualization Model



Gambar 4.12. Detail Native Virtualization

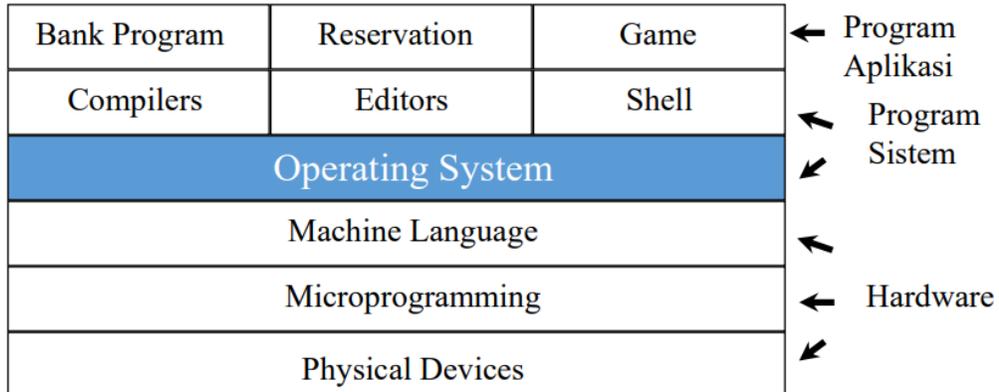
Nutanix AHV adalah sebuah virtualisasi enterprise yang disertakan dengan Acropolis Enterprise Cloud OS, tanpa komponen perangkat lunak tambahan untuk lisensi, instalasi serta pengelolaan kelola. Dimulai dengan teknologi virtualisasi open source, AHV dapat menggabungkan jalur data yang disempurnakan untuk kinerja optimal, keamanan, Flow Virtualization, serta manajemen yang lengkap untuk menghadirkan tumpukan virtualisasi yang lebih kuat, ramping serta biaya virtualisasi yang lebih rendah.

4.5. PERANGKAT LUNAK

➤ Sistem Operasi

Sistem Operasi adalah salah satu faktor terpenting dalam sebuah infrastruktur layanan Internet. Menurut (Barka, 2010) Sistem Operasi merupakan program utama yang menghubungkan Software Aplikasi yang digunakan oleh user dengan hardware. Pengertian sistem operasi secara umum ialah pengelola seluruh sumber-daya yang terdapat pada sistem komputer dan menyediakan sekumpulan layanan (system calls) yang sering disebut "*tools* atau *utility*" berupa aplikasi kepemakai sehingga memudahkan dan menyamankan penggunaan ketika memanfaatkan sumber-daya sistem komputer tersebut.

Pada Gambar 4.13 Di mana layer bekerja sistem operasi bekerja menurut (Michrandi, 2014)



Gambar 4.13. *Operating System Layer* (Michrandi, 2014)

Dalam dunia data center, ada beberapa Sistem operasi yang handal untuk digunakan pada server dalam memenuhi kebutuhan client, di antaranya Linux, FreeBSD & Windows.

➤ **Free BSD**



Gambar 4.14. Logo FreeBSD (*FreeBSD*, 2016)

FreeBSD merupakan sistem operasi berbasis POSIX (Portable Operating System based on UNIX) yang dikembangkan oleh Universitas Barkeley California.

Dalam jurnal (Hutajulu, 2015) mengatakan ciikal bakal Unix ditulis dalam bahasa

Assembly oleh Ken Thompson dan Dennis Ritchie. Dan Sistem operasi ini dinamai UNICS (Uniplexed Information and Computer System), akan tetapi pada tahun 1973 atas usul dari Brian Kernighan, namanya dirubah menjadi Unix, dengan alasan portabilitas, Unix ditulis ulang dalam bahasa C. Selanjutnya, perusahaan AT&T (yang lab-nya, Bell, dipakai untuk mengembangkan Unix oleh Dennis Ritchie) memberikan source code Unix kepada instalasi pendidikan dan pemerintahan untuk lebih mengembangkan penggunaannya.

Pada tahun 1981, Kelompok Peneliti Sistem Komputer, Computer Systems Research Group (CSRG), Universitas California di Berkeley melakukan modifikasi sistem operasi UNIX AT&T untuk dijalankan di mesin DEC VAX II/780. Fitur yang paling menonjol pada release sistem operasi UNIX baru ini adalah dukungan terhadap virtual memory dan implementasi protokol baru dari ARPANET yang selanjutnya dikenal

sebagai Internet Protocols (IP). Sistem operasi UNIX baru ini berbeda sama sekali dengan sistem operasi UNIX AT&T. Sistem operasi ini kemudian dikenal sebagai Berkeley UNIX. Pada proses pengembangan berikutnya, tim BSD berhasil membuat suatu sistem file dengan performa cepat yang diberi nama UNIX File System (UFS).

Perubahan dalam regulasi legal di Amerika Serikat antara tahun 1977 sampai dengan tahun 1984 telah membolehkan pihak AT&T untuk memberikan lisensi UNIX ke vendor lain sehingga pada tahun 1981 Microsoft meluncurkan XENIX yang merupakan UNIX hasil pengembangan tim Santa Cruz Operation (SCO). Pada tahun 1982 AT&T mengembangkan UNIX System III dan System V di tahun 1983 untuk keperluan pasarnya sendiri. Akibat kejadian-kejadian di atas maka pada pertengahan tahun 1980-an terdapat empat versi berbeda dari UNIX yaitu: Research Version, digunakan hanya di lingkungan internal AT&T, Berkeley Software Distribution (BSD) dari Berkeley, System V komersial dari AT&T dan XENIX yang tidak lagi diminati oleh Microsoft, yang dipasarkan oleh perusahaan yang telah mengembangkannya yaitu SCO. Pada 1984, AT&T mulai memasarkan Unix dalam versi-versi baru dengan nama semacam System III, System V, dll. Pada saat yang sama, Ken Thompson, salah seorang pengembang Unix pindah ke UCB (University of California at Berkeley) dan mulai membuat modifikasi untuk Unix.

Versi Berkeley buatan CSRG (Computer Systems Research Group) ini kemudian dinamai seperti 2BSD, 3BSD, dan lain-lain. BSD adalah singkatan dari Berkeley Software Distribution.

Istilah Unix kini telah menjadi nama sebuah keluarga yang terdiri dari puluhan sistem operasi. Dewasa ini, pada dasarnya terdapat dua cabang keluarga Unix yang utama, yaitu keluarga System V buatan AT&T dan keluarga BSD. Sebuah pengembangan dari Unix BSD yaitu 4.4 BSD-Lite yang dikembangkan untuk platform PC compatible inilah yang kemudian dikenal sebagai FreeBSD, yaitu versi gratis dari Unix BSD. Berbeda dengan Linux yang dibuat secara “keroyokan” oleh para programmer di seluruh dunia,

FreeBSD dikembangkan oleh sebuah tim yang terbagi atas tiga kelompok besar, masing-masing adalah FreeBSD Core Team, FreeBSD Developers, dan FreeBSD Documentation Project. FreeBSD Core Team bertindak sebagai semacam “dewan direksi” dari proyek FreeBSD. Tim ini bertanggung jawab untuk menentukan tujuantujuan proyek secara keseluruhan serta aturan pelaksanaannya. Tim kedua, FreeBSD Developers, bertanggung jawab terhadap tugas teknis dalam pembuatan FreeBSD, sementara FreeBSD Documentation Team menangani tugas-tugas yang berkenaan dengan pendokumentasian proyek, pembuatan manual, FAQ, dan sebagainya. Saat ini FreeBSD banyak dipercaya sebagai OS untuk server yang menangani beban yang cukup tinggi.

Tercatat beberapa situs internet tersibuk di dunia, seperti Yahoo.com, Hotmail.com, dan ftp.cdrom.com menggunakan FreeBSD sebagai sistem operasi bagi servernya. Sementara itu, di Indonesia FreeBSD menjadi tulang punggung dari jaringan AI3 (Asian Internet Interconnection Initiative). Jaringan yang di Indonesia berpusat di ITB (Institut Teknologi Bandung) ini bertujuan untuk menghubungkan

perguruan tinggi serta lembaga pendidikan maupun riset di Indonesia ke jaringan internet melalui gateway internet AI3 di ITB.

FreeBSD adalah sebuah sistem operasi komputer berbasis 4.4BSD-Lite2, sebuah versi dari UNIX yang dikembangkan oleh University of California di Berkeley (Hutajulu, 2015). Saat ini FreeBSD dikembangkan oleh sebuah group developer dari seluruh penjuru dunia. Dalam melengkapi diri sebagai sebuah sistem operasi yang lengkap bagi penggunanya, FreeBSD memberikan dukungan sebuah koleksi metapackage yang berguna untuk membantu instalasi aplikasi tambahan yang disebut sebagai Ports Collection serta Packages sebagai bentuk paket siap instal. Di Internet terdapat banyak sekali aplikasi gratis yang umumnya didistribusikan dalam bentuk source code.

Hal ini menjadi masalah baru: bagaimana cara mendownload, melakukan build, dan menginstalnya. Dalam tulisan ini dibahas mengenai bagaimana cara menginstal aplikasi menggunakan packages dan ports, dan melakukan pemeliharaan packages terinstal. Juga dibahas mengenai masalah dependensi, mengupdate repository ports. Yang tidak dibahas dalam tulisan ini adalah bagaimana menginstall FreeBSD, melakukan initial setup pada sistem operasi, dan bagaimana cara membuat ports. Sebuah bagian penting dalam sistem operasi FreeBSD adalah koleksi Ports, yang saat ini memiliki jumlah 8.633 buah dari aplikasi UNIX yang populer digunakan.

Koleksi ports memberikan otomatisasi proses porting aplikasi untuk digunakan pada sistem FreeBSD. Sebuah kombinasi dari berbagai macam programming tools yang telah tersedia dalam instalasi dasar

FreeBSD yang memudahkan pengguna dengan melakukan sebuah perintah untuk menginstal sebuah aplikasi, dan mekanisme ports yang akan melakukan pekerjaan selebihnya. Jika pengguna lebih memilih penggunaan aplikasi siap instal, biasanya karena keterbatasan waktu atau resource, maka ada alternatif lain yaitu dengan menggunakan packages, dimana dalam waktu singkat pengguna dapat mendapatkan aplikasi yang diinginkannya terinstal berikut dengan :

➤ **File Sistem FreeBSD**

Sistem FreeBSD (Hutajulu, 2015) adalah suatu sistem yang mempunyai kemampuan menyimpan log file sistem transaksi, yakni perubahan yang menjadi pelengkap disk untuk menulis operasi sebelum meta-data dan file menulis berkomitmen untuk disk yang tepat. Transaksi log ini dikemudian hari dapat mengikuti langkah untuk file sistem transaksi, mencegah ketidakseimbangan sistem file. Model pengembangan FreeBSD diarahkan untuk menciptakan produk yang stabil dan mudah digunakan. Sebagai salah satu sistem Unix yang reliabel untuk platform x86, FreeBSD harus menjaga kompatibilitas program-program sebaik mungkin di antara sistem. memiliki wewenang langsung untuk melakukan perubahan-perubahan pada sistem. Selain FreeBSD, Sistem operasi lain yang berbasis BSD adalah NetBSD dan OpenBSD. Perbedaan dari ketiga OS tersebut dapat di lihat dari kelebihanannya masing-masing (Hutajulu, 2015):

- a. FreeBSD: Mendukung banyak party software dengan semboyan "ready to serve".

- b. OpenBSD: Menitikberatkan pada security, dengan slogan "secure by default".
- c. NetBSD: Mendukung banyak hardware dan berbagai arsitektur.

➤ **Apache Web Server**

Web server (Evy, 2012) adalah software yang menjadi tulang belakang dari world wide web (www) yang pertama kali tercipta sekitar tahun 1980an. Web server menunggu permintaan dari client yang menggunakan browser seperti Netscape Navigator, Internet Explorer, Mozilla Firefox, dan program browser lainnya. Jika ada permintaan dari browser, maka web server akan memproses permintaan itu kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke browser.

Data ini mempunyai format yang standar, disebut dengan format SGML (Standar General Markup Language). Data yang berupa format ini kemudian akan ditampilkan oleh browser sesuai dengan kemampuan browser tersebut. Contohnya, bila data yang dikirim berupa gambar, browser yang hanya mampu menampilkan teks (misalnya lynx) tidak akan mampu menampilkan gambar tersebut, dan jika ada akan menampilkan alternatifnya saja.

Web server, untuk berkomunikasi dengan client-nya (web browser) mempunyai protokol sendiri, yaitu HTTP (hypertext transfer protocol). Dengan protokol ini, komunikasi antar web server dengan client-nya dapat saling dimengerti dan lebih mudah. Seperti telah dijelaskan diatas, format data pada world wide web adalah SGML. Tapi para pengguna

internet saat ini lebih banyak menggunakan format HTML (hypertext markup language) karena penggunaannya lebih sederhana dan mudah dipelajari.

Standarisasi web server dalam penerapan penggunaannya antara lain dikeluarkan oleh W3C (World Wide Web Consortium), IETF (Internet Engineering Task Force), dan beberapa organisasi lainnya. Sampai saat ini, sudah lebih dari 110 spesifikasi yang dirilis oleh W3C (W3C Recommendations). Contoh standarisasi web server antara lain :

1. Spesifikasi HTML, CSS, DOM dan XHTML (W3C)
2. Spesifikasi Javascript (ECMA)
3. URL, HTTP (IETF) dalam bentuk dokumen RFC



Apache (Silitonga, Suswaini, & Kurniawan, 2013) adalah sebuah nama web server yang bertanggung jawab pada request-response HTTP

dan logging informasi secara detail. Selain itu, Apache juga diartikan sebagai suatu web server yang kompak, modular, mengikuti standar protokol HTTP, dan paling banyak digunakan

BAB 5

DED MIDDLEWARE

5.1. STRUKTUR DAN ALUR PROSES BISNIS

Perancangan *Government Resources Management System* memiliki 7 (tujuh) segmen yang memiliki kegiatan pengelolaan keuangan yang terintegrasi, antara lain:

1. Sistem Informasi Perencanaan Pembangunan (E-Planning)

Sistem Informasi Perencanaan Pembangunan yang dikembangkan dan dikelola oleh BAPPEDA. **E-Planning** merupakan alat penyusunan berdasarkan Musyawarah Perencanaan Pembangunan Pemerintah Aceh, RKPA s/d KUA PPAS dan perubahannya untuk membantu Satuan Kerja Perangkat Aceh (SKPA) dan kabupaten/kota agar sesuai dengan RPJMA dan arahan yang terkandung dalam Permendagri 86 Tahun 2017, sehingga dapat terselesaikan dengan mudah, cepat, tepat dan sesuai skedul perencanaan.

2. Sistem Informasi Rencana Anggaran (E-Budgeting)

Sistem Informasi Rencana Anggaran (**E-Budgeting**) merupakan sebuah sistem elektronik budgeting untuk penyusunan anggaran yang di dalamnya termasuk aplikasi program komputer berbasis web untuk

memfasilitasi proses penyusunan anggaran belanja daerah. **E-Budgeting** bertujuan untuk memudahkan Tim Anggaran dalam penyusunan anggaran dan meningkatkan kualitas Anggaran Pendapatan dan Belanja Daerah (APBA) dari sisi kesesuaian dengan Rencana Pembangunan Jangka Menengah Aceh (RPJMA), keakuratan nilai dan rekening serta akuntabilitas alokasi belanja. Manfaat dari E-budgeting sebagai berikut:

- a. Proses yang transparan,
- b. Proses penyesuaian singkat dan tepat waktu,
- c. Anggaran sesuai dengan kebutuhan dalam kegiatan,
- d. Kronologis anggaran jelas,
- e. *Report* sesuai permintaan Ketua Tim Anggaran dapat terpenuhi.

3. Sistem Informasi Rencana Pekerjaan (E-Project Planning)

Sistem Informasi Rencana Pekerjaan (**E-Project Planning**) merupakan sebuah sistem elektronik *planning* berbasis web untuk memfasilitasi perencanaan kegiatan yang di dalamnya termasuk pencatatan rincian rencana kegiatan yang ditetapkan berdasarkan faktor waktu, alokasi anggaran, dan volume kegiatan. Program ini dibuat dengan latar belakang kebutuhan sistem penyerapan pekerjaan SKPA yang terintegrasi dengan Dokumen Pelaksana Anggaran (DPA), pengalokasian pelaksanaan anggaran ke dalam paket-paket pekerjaan dan kebutuhan akan informasi pelaksanaan anggaran kegiatan secara triwulan yang tertuang dalam DPA.

Berikut tujuan dari **E-Project Planning**:

- a. Menyediakan sistem untuk perencanaan pelaksanaan anggaran secara terintegrasi

- b. Meminimalisasi tingkat kesalahan
- c. Meningkatkan efisiensi dan efektifitas
- d. Mengalokasikan anggaran ke dalam paket-paket pekerjaan
- e. Merencanakan pelaksanaan anggaran kegiatan selama satu tahun anggaran
- f. Menyediakan data pelaksanaan anggaran kegiatan secara triwulan
- g. Menyediakan dokumen pendukung dalam kontrak kinerja.
- h. Melakukan transfer data ke E-Procurement

4. Sistem Informasi Pengadaan Barang (E-Procurement)

Sistem Informasi Pengadaan Barang (**E-Procurement**) adalah sistem lelang sebuah proyek yang dilakukan melalui internet. Proses pelaksanaan pelelangan umum melalui SPSE (Sistem Pengadaan Secara Elektronik) dimulai dari Unit Kerja/SKPA memasukkan paket kerja yang akan dilelang ke Unit Layanan Pengadaan (ULP). Kemudian ULP mengundang SKPA tersebut beserta Gugus Tugas Pelelangan (GTP) yang telah ditunjuk untuk membahas dokumen pengadaan pekerjaan yang akan dilelang untuk kemudian data dari GTP dan dokumen pengadaan pekerjaan yang akan dilelang tersebut dimasukkan ke portal **E-Procurement** oleh ULP. Data yang dimasukkan kemudian dicek oleh Sekretariat SIPB-A sebelum disetujui oleh pihak Pejabat Pembuat Komitmen (PPK) dan dapat di download dokumen lelang dan tawar menawar oleh pihak penyedia barang/jasa yang telah teregistrasi. Selanjutnya, GTP memproses pelelangan paket sampai usulan calon pemenang kepada PPK untuk dimasukkan ke dalam portal e-

Procurement data pemenang dari lelang paket pekerjaan tersebut. Kemudian PPK membuat Surat Penunjukan Penyedia Barang/Jasa (SPPBJ) berikut juga membuat kontrak pengadaan barang/jasa dengan penyedia barang/jasa.

5. Sistem Informasi Administrasi Kontrak (E-Delivery)

Sistem Pendukung Administrasi Kegiatan Kontrak (**E-Delivery**) adalah yang di dalamnya termasuk program komputer berbasis web untuk memfasilitasi kebutuhan pembuatan kontrak pengadaan barang/jasa dan penyediaan dokumen-dokumen kelengkapan.

6. Sistem Informasi Monitoring Evaluasi (E-Controlling)

Sistem Informasi Monitoring Evaluasi (**E-Controlling**) adalah salah satu bagian dari skenario besar *e-Government*, melalui konsep GRMS (Government Resource Management System). Sistem ini menjembatani antara perencanaan pelaksanaan pekerjaan yang ada di **E-Planning** dengan hasil pekerjaan yang diinput melalui sistem **E-Delivery**. Melalui **E-Controlling** dapat digunakan untuk mengetahui progress fisik masing-masing kegiatan setiap bulan, apakah sesuai e-project planning dan e-delivery atau tidak. Hal ini tak bisa di manipulasi, karena yang bisa memasukkan hanya penyedia barang dan jasa.

7. Sistem Informasi Kinerja (E-Performance)

Sistem Informasi Kinerja (**E-Performance**) merupakan sistem informasi manajemen kinerja dalam rangka penilaian prestasi kinerja pegawai yang lebih objektif, terukur, akuntabel, partisipatif dan transparan sehingga bisa terwujud pembinaan pegawai berdasarkan prestasi kerja

dan sistem karir kerja PNS di tentang Petunjuk Teknis Pemberian Uang Kinerja pada Belanja Langsung kepada PNS di Lingkungan Pemerintah Provinsi Aceh.



Gambar 5.1. Model GRMS Provinsi Aceh

Berdasarkan gambar diatas, maka dapat dijelaskan secara ringkas alur model GRMS Provinsi Aceh sebagai berikut :

1. Usulan program atau kegiatan berdasarkan Musrembang di kumpulkan lalu di input ke dalam E-Planning.
2. Usulan yang telah masuk di E-Planning akan diolah masuk pada perencanaan anggaran pada E-Budgeting.
3. Langkah selanjutnya Proses pemaketan program berdasarkan waktu, anggaran serta volume kegiatan dilakukan pada E-Project Planning.
4. Paket yang dilelang akan di masukkan ke dalam E-Procurement berdasarkan referensi data dari E-Project Planning.
5. Selanjutnya, untuk dokumentasi administrasi paket-paket yang telah terkontrak pada program E-Procurement akan di fasilitasi pada E-Delivery seperti membuat Surat Penunjukan Penyedia Barang/Jasa (SPPBJ), dll
6. Paket-paket terkontrak akan di monitoring realisasi fisik & keuangannya pada E-Monitoring.
7. Paket-paket yang melibatkan kinerja personil pada suatu SKPA akan dimonitoring Target & Realisasi, Ketepatan Waktu serta Efisiensi Anggaran pada E-Performance sehingga nantinya Score Kinerja instansional didapatkan.

5.2. ARSITEKTUR APLIKASI

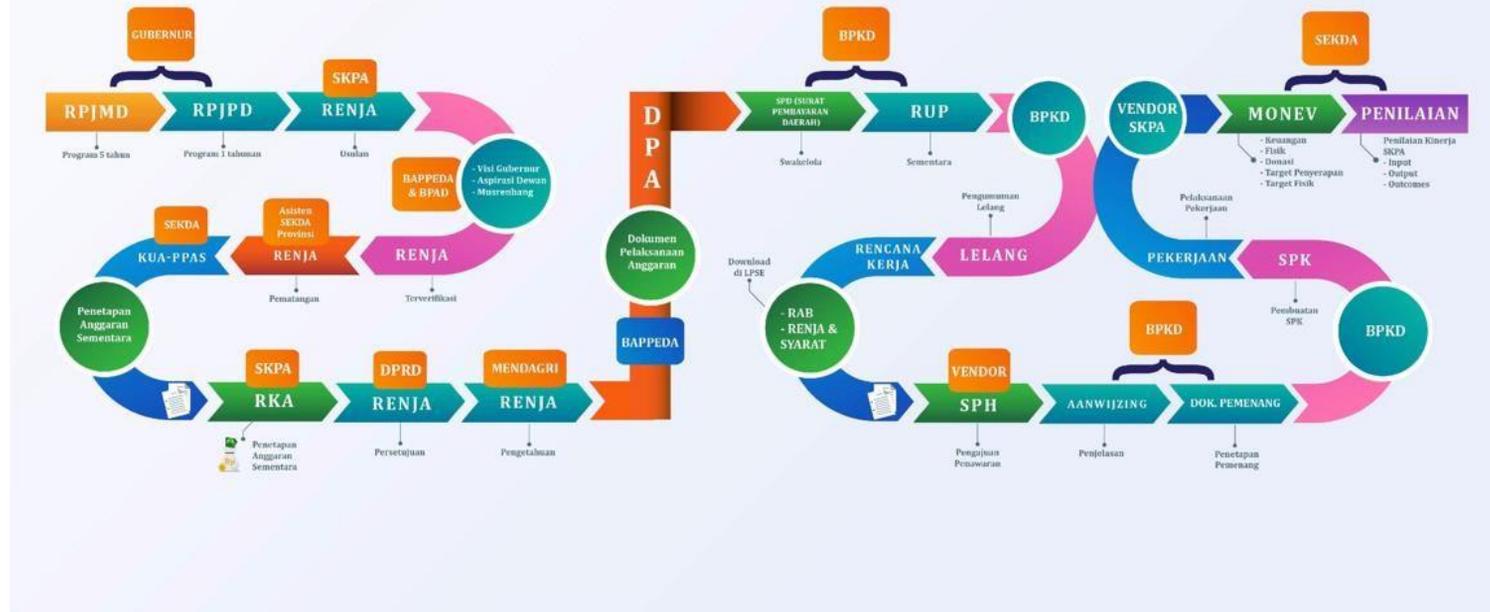
➤ Alur Proses GRMS

Pemerintah Aceh akan menerapkan sistem GRMS (*Government Resources Management System*) atau sistem informasi manajemen sumber daya pemerintahan dalam hal pengelolaan keuangan daerah yang terintegrasi dari aktivitas birokrasi hulu sampai hilir. Perancangan GRMS dikelola oleh

Dinas Komunikasi, Informatika dan Persandian Aceh dan aktivitas-aktivitasnya meliputi proses penyusunan anggaran (e-budgeting), perencanaan pekerjaan (e-project planning), pelelangan secara elektronik (e-procurement), administrasi kontrak dan pencairan pekerjaan (e-delivery), pemantauan pelaksanaan kegiatan (e-controlling) dan pengukuran kinerja PNS (e-performance). GRMS dapat diakses dimana pun dan kapan pun melalui media internet. Secara berkelanjutan, Pelaksanaan pengembangan sistem yang dilakukan secara berkelanjutan bertujuan untuk penyempurnaan dan penyesuaian terhadap peraturan perundangan-undangan dan kebutuhan para pengguna.

GRMS memiliki 7 (tujuh) segmen yang memiliki aktifitas perencanaan program kerja dan proses keuangan hingga kepada penilaian kinerja Satuan Kerja Perangkat Aceh (SKPA). Bentuk integrasi dapat dilihat pada gambar berikut:

ALUR GRMS

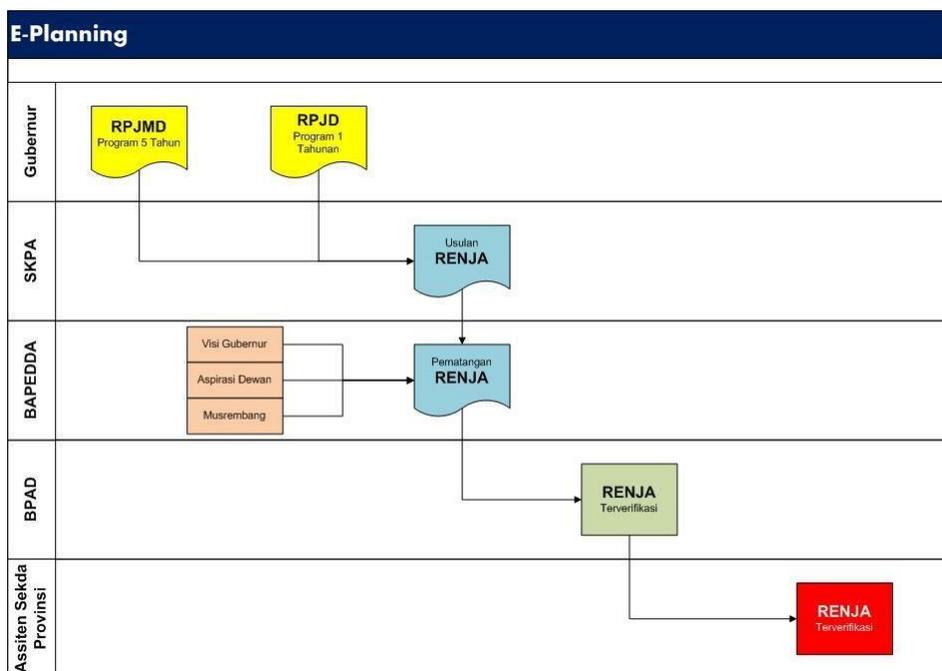


Gambar 5.2. Alur GRMS

Berdasarkan gambar diatas, dapat dijelaskan masing-masing proses dan fungsi sistem, diantaranya adalah :

1. E-Planning

E-Planning dikembangkan dan dikelola oleh BAPPEDA. E-Planning merupakan alat penyusunan berdasarkan Musyawarah Perencanaan Pembangunan Pemerintah Aceh, RKPA s/d KUA PPAS dan perubahannya untuk membantu Satuan Kerja Perangkat Aceh (SKPA) dan kabupaten/kota agar sesuai dengan RPJMA dan arahan yang terkandung dalam Permendagri 54 th 2010 dan Permendagri 86 Tahun 2017, sehingga dapat terselesaikan dengan mudah, cepat, tepat dan sesuai jadwal perencanaan. Berikut ini gambar alur proses penerapan E-Planning.

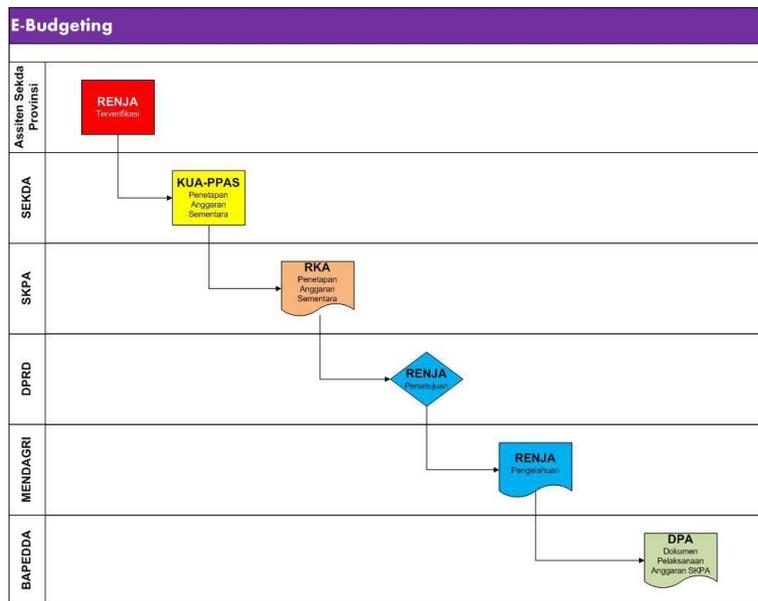


Gambar 5.3. Diagram Alur Sistem E-Planning

8. E-Budgeting

E-Budgeting merupakan sebuah sistem elektronik budgeting untuk penyusunan anggaran yang di dalamnya termasuk aplikasi program komputer berbasis web untuk memfasilitasi proses penyusunan anggaran belanja daerah. E-budgeting bertujuan untuk memudahkan Tim Anggaran dalam penyusunan anggaran dan meningkatkan kualitas Anggaran Pendapatan dan Belanja Daerah (APBA) dari sisi kesesuaian dengan Rencana Pembangunan Jangka Menengah Aceh (RPJMA), keakuratan nilai dan rekening serta akuntabilitas alokasi belanja. Manfaat dari e-budgeting sebagai berikut:

- a. Proses yang transparan,
- b. Proses penyesuaian singkat dan tepat waktu,
- c. Anggaran sesuai dengan kebutuhan dalam kegiatan,
- d. Kronologis anggaran jelas,
- e. *Report* sesuai permintaan Ketua Tim Anggaran dapat terpenuhi.



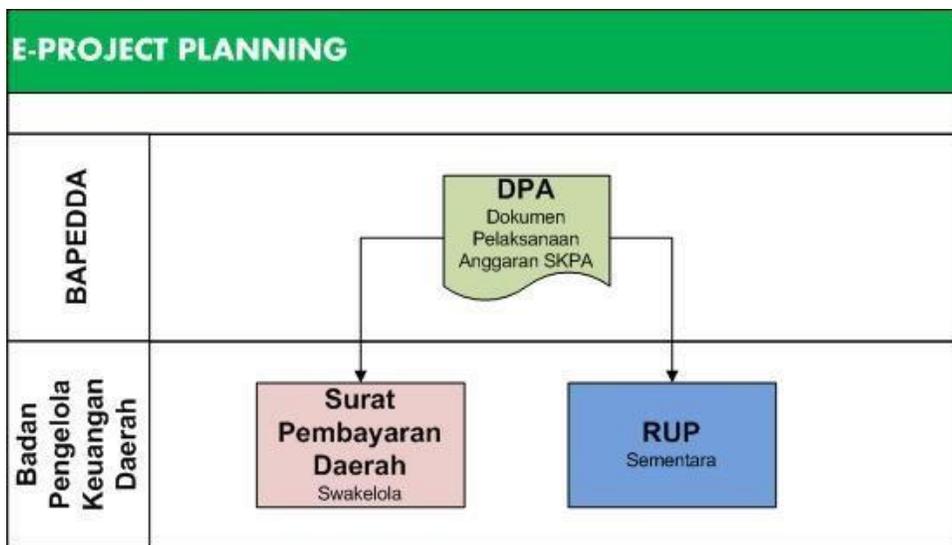
Gambar 5.4. Diagram Alur Sistem E-Budgeting

9. e-Project Planning

E-Project Planning merupakan sebuah sistem elektronik *planning* berbasis web untuk memfasilitasi perencanaan kegiatan yang didalamnya termasuk pencatatan rincian rencana kegiatan yang ditetapkan berdasarkan faktor waktu, alokasi anggaran, dan volume kegiatan. Program ini dibuat dengan latar belakang kebutuhan sistem penyerapan pekerjaan SKPA yang terintegrasi dengan Dokumen Pelaksana Anggaran (DPA), pengalokasian pelaksanaan anggaran ke dalam paket-paket pekerjaan dan kebutuhan akan informasi pelaksanaan anggaran kegiatan secara triwulan yang tertuang dalam DPA. Berikut tujuan dari sistem e-Project Planning :

- Menyediakan sistem untuk perencanaan pelaksanaan anggaran secara terintegrasi
- Meminimalisasi tingkat kesalahan

- c. Meningkatkan efisiensi dan efektifitas
- d. Mengalokasikan anggaran ke dalam paket-paket pekerjaan
- e. Merencanakan pelaksanaan anggaran kegiatan selama 1 tahun anggaran
- f. Menyediakan data pelaksanaan anggaran kegiatan secara triwulan
- g. Menyediakan dokumen pendukung dalam kontrak kinerja.
- h. Melakukan transfer data ke e-Procurement

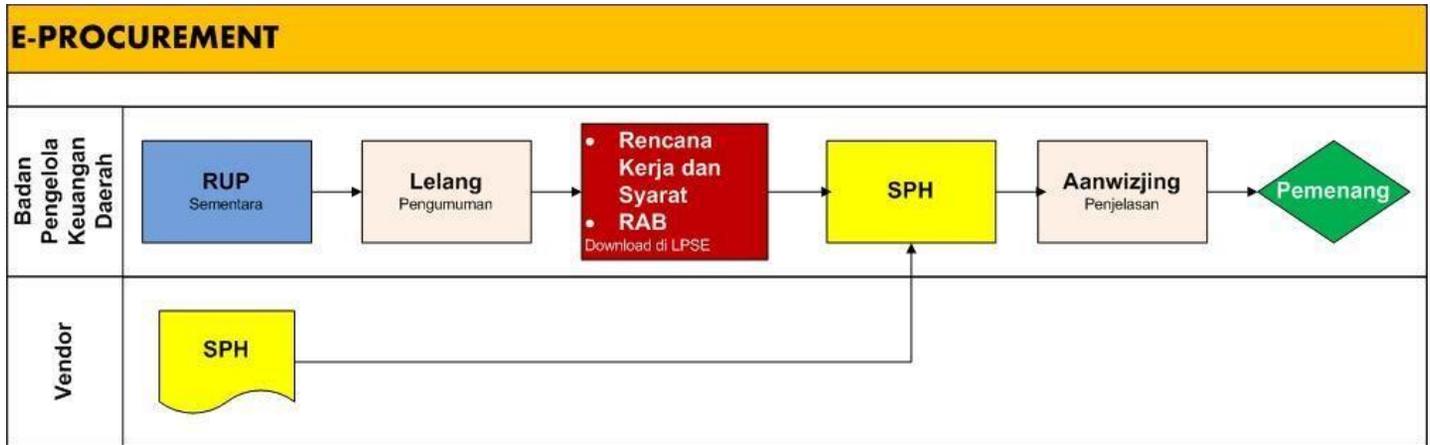


Gambar 5.5. Diagram Alur Sistem E-Project Planning

10.e-Procurement

e-Procurement adalah sistem lelang sebuah proyek yang dilakukan melalui internet. Proses pelaksanaan pelelangan umum melalui SPSE (Sistem Pengadaan Secara Elektronik) dimulai dari Unit Kerja/SKPA memasukkan paket kerja yang akan dilelang ke Unit Layanan Pengadaan (ULP). Kemudian ULP mengundang SKPA tersebut beserta Gugus Tugas Pelelangan (GTP) yang telah ditunjuk untuk membahas

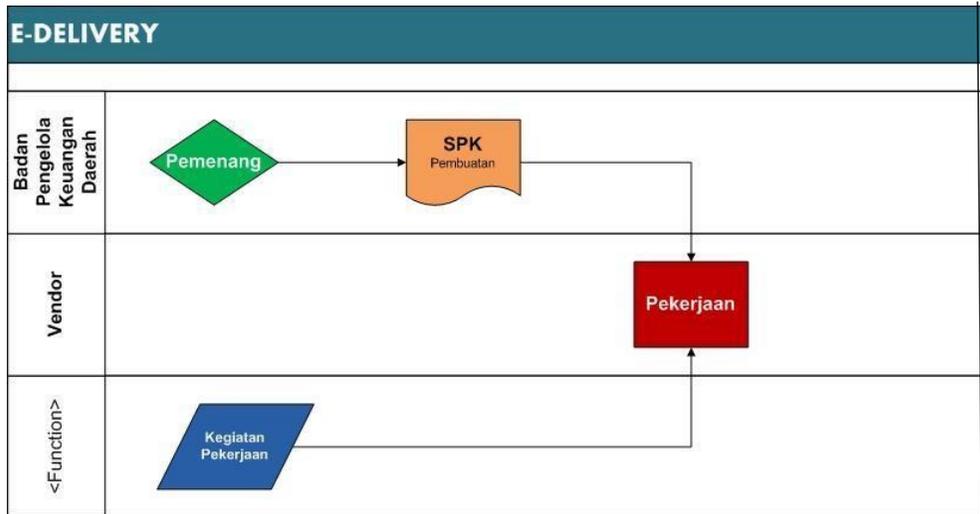
dokumen pengadaan pekerjaan yang akan dilelang untuk kemudian data dari GTP dan dokumen pengadaan pekerjaan yang akan dilelang tersebut dimasukkan ke portal e-Procurement oleh ULP. Data yang dimasukkan kemudian dicek oleh Sekretariat sebelum disetujui oleh pihak Pejabat Pembuat Komitmen (PPK) dan dapat di download dokumen lelang dan tawar menawar oleh pihak penyedia barang/jasa yang telah teregistrasi. Selanjutnya, GTP memproses pelelangan paket sampai usulan calon pemenang kepada PPK untuk dimasukkan ke dalam portal e- Procurement data pemenang dari lelang paket pekerjaan tersebut. Kemudian PPK membuat Surat Penunjukan Penyedia Barang/Jasa (SPPBJ) berikut juga membuat kontrak pengadaan barang/jasa dengan penyedia barang/jasa.



Gambar 5.6. Diagram Alur Sistem E-Procurement

11.e-Delivery

E-Delivery adalah yang di dalamnya termasuk program komputer berbasis web untuk memfasilitasi kebutuhan pembuatan kontrak pengadaan barang/jasa dan penyediaan dokumen-dokumen kelengkapan



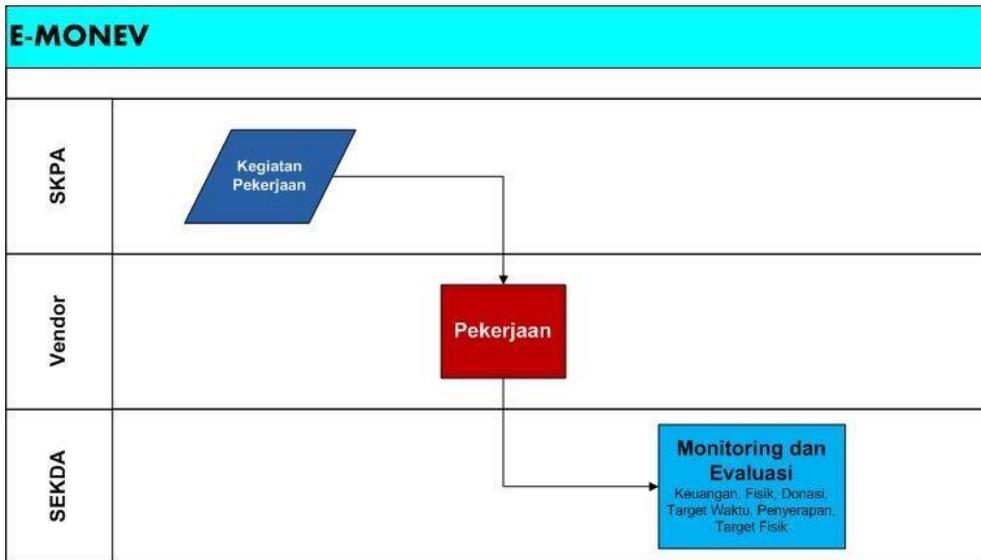
Gambar 5.7. Diagram Alur Sistem E-Delivery

12.e-Monev

67

E-Monev adalah salah satu bagian dari skenario besar *e-Government*, melalui konsep GRMS (Government Resource Management System). Sistem ini menjembatani antara perencanaan pelaksanaan pekerjaan yang ada di e-Project Planning dengan hasil pekerjaan yang diinput melalui sistem e-Monev. Melalui e-Monev dapat digunakan untuk mengetahui progress fisik masing-masing kegiatan setiap bulan, apakah sesuai e-project planning dan e-delivery atau tidak. Hal ini tak

bisa di manipulasi, karena yang bisa memasukkan hanya penyedia barang dan jasa.



Gambar 5.8. Diagram Alur Sistem E-Monev

13.e-Performance

E-Performance merupakan sistem informasi manajemen kinerja dalam rangka penilaian prestasi kinerja pegawai yang lebih objektif, terukur, akuntabel, partisipatif dan transparan sehingga bisa terwujud pembinaan pegawai berdasarkan prestasi kerja dan sistem karir kerja PNS di tentang Petunjuk Teknis Pemberian Uang Kinerja pada Belanja Langsung kepada PNS di Lingkungan Pemerintah Provinsi Aceh.



Gambar 5.9. Diagram Alur Sistem E-Monev

Integrasi Data Aplikasi GRMS

1. e-planning

Tabel 2.1. Parameter e-planning

No	Parameters	Type	Length	Description
1	id_usulan	numeric	10	ID Usulan Program
2	usulan_program	text	10	Usulan Program
3	jenis_program	text	10	Jenis Usulan Program (Fisik, Non-Fisik)
4	kecamatan	text	50	Kecamatan Usulan Program
5	kelurahan	text	50	Kelurahan Usulan Program
6	lokasi	text	200	Alamat Lokasi Usulan Program
7	volume	numeric	20	Volume Usulan Program
8	satuan_volume	text	10	Satuan Volume Usulan Program
9	anggaran	numeric	50	Satuan Volume Usulan Program
10	id_skpa	numeric	10	Kode SKPA
11	nama_skpa	text	50	Nama SKPA

2. e-budgeting

Tabel 5.2. Parameter e-budgeting

No	Parameters	Type	Length	Description
1	kode_rekening_program	numeric	20	Kode Rekening Program
2	nama_program	text	200	Nama Program
3	id_detail_kegiatan	numeric	20	Id / Kode Sub Rekening Kegiatan Suatu Program
4	nama_detail_kegiatan	text	200	Nama Kegiatan Suatu Program
5	total_kegiatan	numeric	10	Total Kegiatan Pada Suatu Program

No	Parameters	Type	Length	Description
6	anggaran	numeric	20	Total Anggaran Suatu Program
7	status_kegiatan	text	20	Status Pengajuan Kegiatan(Revisi, Approve, dll)
8	jenis_program	text	10	Jenis Usulan Program (Fisik, Non-Fisik)
9	sumber_dana	numeric	10	Sumber Pendanaan(Rupiah Murni, Loan, SBSN)
10	tahun_rencana	numeric	4	Tahun Perencanaan
11	id_skpa	numeric	10	ID SKPA Pada Program Berjalan
12	nama_skpa	text	50	Nama SKPA
13	id_usulan	numeric	20	ID_usulan program yang disetujui pada e-planning
14	usulan_program	text	200	Nama usulan program yang disetujui pada e-planning
15	lokasi	text	200	Alamat Lokasi Usulan Program

3. e-project planning

Tabel 5.3. Parameter e-project planning

No	Parameters	Type	Length	Description
1	id_paket	numeric	20	ID Paket Berdasarkan Program-Program yang telah disetujui & menyelesaikan tahap administrasi pada e-Budgeting

No	Parameters	Type	Length	Description
2	nama_paket	text	200	Nama Paket Berdasarkan Program-Program yang telah disetujui & menyelesaikan tahap administrasi pada e-Budgeting
3	kode_rekening_program	numeric	20	Kode Rekening Program
4	nama_program	text	200	Nama Program
5	sumber_dana	numeric	10	Sumber Pendanaan(Rupiah Murni, Loan, SBSN)
6	anggaran	numeric	20	Total Anggaran Suatu Program
7	tahun_anggaran	numeric	4	Tahun Anggaran Pelaksanaan Lelang
8	jenis_kontrak	text	10	MYC Baru, MYC Lanjutan, Tahunan
9	id_skpa	numeric	10	ID SKPA Pada Usulan Paket
10	nama_skpa	text	50	Nama SKPA Pada Usulan Paket

4. e-procurement

Tabel 5.4. Parameter e-procurement

No	Parameters	Type	Length	Description
1	kode_lelang	numerik	20	ID Lelang Pekerjaan
2	nama_paket_lelang	text	50	Nama Paket Pada Sistem
3	kode_paket	numeric	10	Kode Paket yang telah di setujui pada e-Project Planning
4	paket	text	150	Nama Paket yang telah di setujui pada e-Project Planning

5	sumber_dana	numeric	10	Sumber Pendanaan(Rupiah Murni, Loan, SBSN)
6	jenis	text	20	Jenis Kontrak (Tahunan, MYC Lanjutan, MYC Baru)
7	kategori	text	25	Kategori (Konsultasi, Konstruksi, Barang, Jasa Lainnya)
8	nilai_pagu_paket	numeric	50	Besaran Nilai Pagu Paket
9	nilai_kontrak	numeric	50	Besaran Nilai Paket Terkontrak
10	pemenang_lelang	numeric	50	Nama Rekanan Pemenang Lelang
11	id_skpa	numeric	10	ID SKPA Pada Paket Terkontrak
12	nama_skpa	text	50	Nama SKPA Pada Paket Terkontrak

5. e-delivery

Tabel 5.5. Parameter e-delivery

No	Parameters	Type	Length	Description
1	kode_lelang	numerik	20	ID Lelang Pekerjaan e-Procurement
2	nama_paket_lelang	text	50	Nama paket Pada Sistem e-Procurement
3	kode_paket	numerik	20	ID Paket Pekerjaan
4	nama_paket	text	100	Nama Paket Pekerjaan
5	id_skpa	numerik	20	ID Paket Pekerjaan Pada SKPA
6	nama_skpa	text	50	Nama SKPA pada Paket Pekerjaan dilaksanakan
7	kode_kontrak	numerik	20	ID Kontrak Paket Pekerjaan
8	nilai_kontrak	numerik	25	Nilai Kontrak PaketPekerjaan
9	pemenang_lelang	text		
10	lampiran_documento_kontrak	blob	-	Lampiran-lampiran Dokumen Kontrak Paket Pekerjaan

6. e-monev

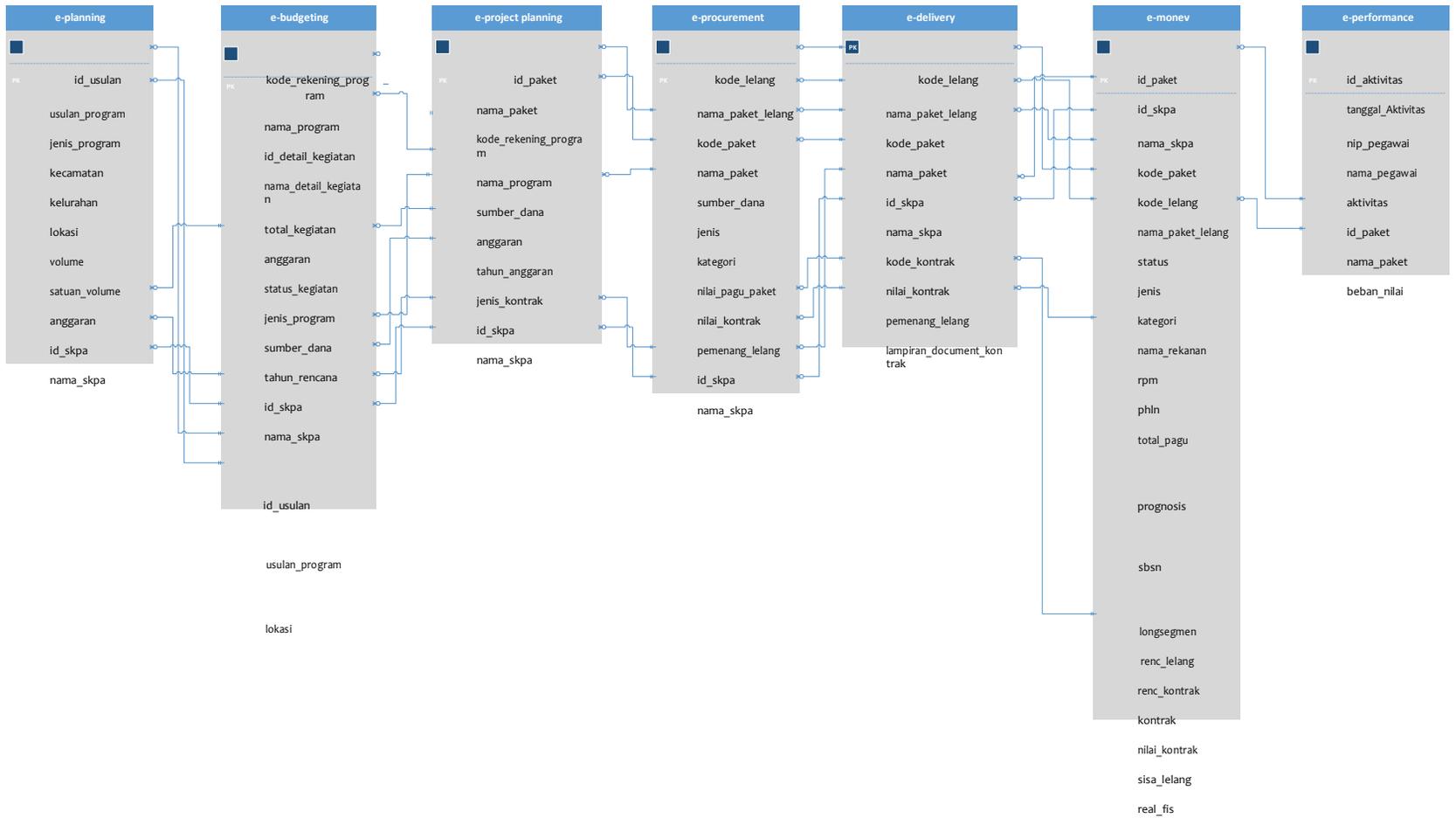
Tabel 5.6. Parameter e-monev

No	Parameters	Type	Length	Description
1	id_paket	numerik	10	ID Paket Pada Sistem e-Monitoring & Evaluasi
2	id_skpa	numeric	10	Kode SKPA
3	nama_skpa	text	50	Nama SKPA
4	kode_paket	numeric	10	Kode Paket Pada Sistem e-Delivery
5	kode_lelang	numerik	20	ID Lelang Pekerjaan e-Procurement
6	nama_paket_lelang	text	50	Nama Paket Pada Sistem e-Procurement
7	status	text	10	Status Kontrak (Terkontrak, Lelang, Belum Lelang)
8	jenis	text	20	Jenis Kontrak (Tahunan, MYC Lanjutan, MYC Baru)
9	kategori	text	25	Kategori (Konsultasi, Konstruksi, Barang, Jasa Lainnya)
10	nama_rekanan	text	150	Swakelola (Vendor)
11	rpm	numeric	50	RPM (Rupiah Murni)
12	phln	numeric	50	PHLN (Loan)
13	total_pagu	numeric	50	Pagu
14	prognosis	numeric	3	Prognosis (%)
15	sbsn	numeric	50	SBSN
16	longsegmen	numeric	50	Longsegmen
17	renc_lelang	date time	25	Tanggal Rencana Lelang
18	renc_kontrak	date time	25	Tanggal Rencana Kontrak
19	kontrak	date time	25	Tanggal Kontrak
20	nilai_kontrak	numeric	50	Nilai Kontrak
21	sisa_lelang	numeric	50	Sisa Lelang
22	real_fis	numeric	50	Realisasi Fisik
23	real_keu	numeric	50	Realisasi Keuangan

7. e-performance

Tabel 5.7. Parameter e-performance

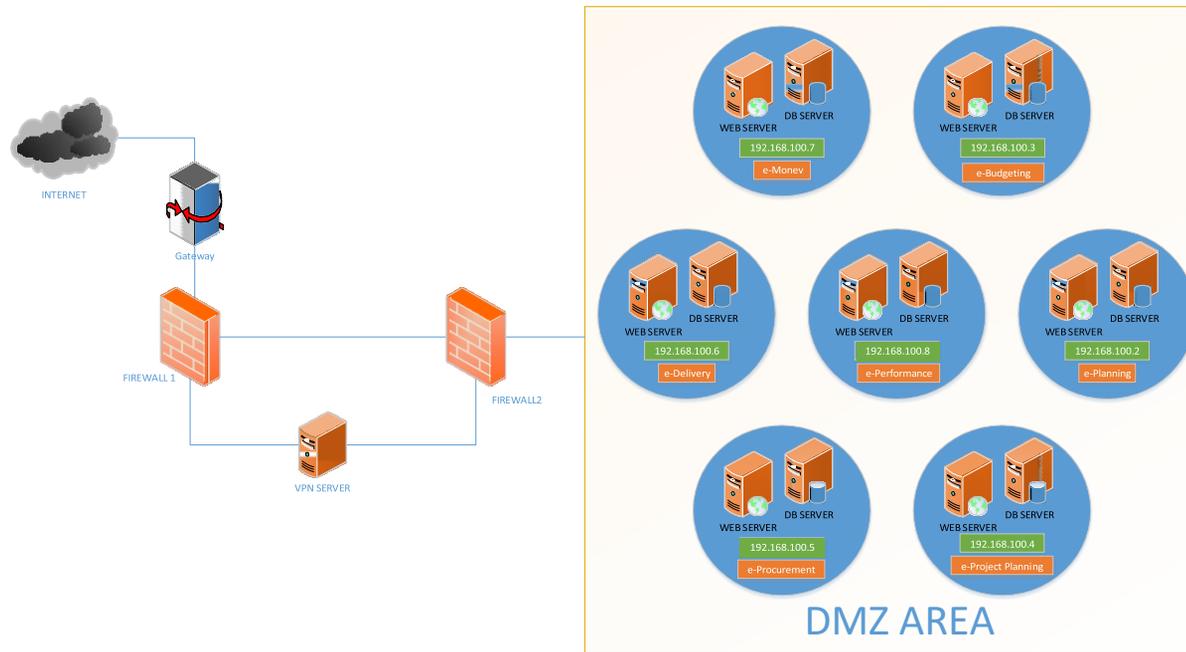
No	Parameters	Type	Length	Description
1	id_aktivitas	numerik	10	Id aktivitas
2	tanggal_Aktivitas	date	20	Tanggal Aktivitas
3	nip_pegawai	numeric	20	NIP Pegawai
4	nama_pegawai	text	50	Nama Pegawai
5	aktivitas	text	200	Aktivitas
6	id_paket	numeric	20	ID Paket Terkontrak Dari e-Monev
7	nama_paket	text	200	Nama Paket Terkontrak Dari e-Monev
8	beban_nilai	numeric	10	Bobot Penilaian Aktivitas yang dilakukan Pegawai Terhadap Suatu Paket



ambar 5.10. Relasi Usulan

real_keu

Topologi & Sistem Keamanan



Gambar 5.11. Usulan Topologi & Sistem Keamanan GRMS

DAFTAR PUSTAKA

- Afrianto, I., & Setiawan, E. B. (2014). Kajian Virtual Private Network (VPN) Sebagai Sistem Pengamanan Data Pada Jaringan Komputer. *Majalah Ilmiah UNIKOM Vol.12*, 43-52.
- Alter, S. (1992). *Information Systems a Management Perspective*. Addison-Wesley.
- Apache. (2018). Retrieved from <https://www.apache.org/>
- Badriyah, T. (2007). *Software Engineering- Topik: Unified Modeling Language (UML)*. Surabaya: Institut Teknologi Sepuluh November.
- Barka, S. (2010). Mengenal Sistem Operasi Yang Beredar Disekitar Kita. *JURNAL DASI*, 19-32.
- Basuki, A. P. (2010). *Membangun Web Berbasis PHP Dengan Framework Codeigniter*. Yogyakarta: Penerbit Lokomedia.
- Connolly, T. M., & Carolyn, E. B. (2002). *Database System: A Practical Approach to Design, Implementation, and Management, Third Edition*. Addison-Wesley: Reading, Massachusetts.
- Davis, G. B. (1991). *Kerangka Dasar Sistem Informasi Manajemen*. Jakarta: Pustaka Binaman Pressindo.
- Evy, N. (2012). Analisis Dan Perancangan Web Server Pada Handphone. *Studia Informatika: Jurnal Sistem Informasi*, 5(2), 1-17.
- FreeBSD. (2016, 10 13). Retrieved from <https://www.freebsd.org/logo.html>
- Gaol, C. (2008). *Sistem Informasi Manajemen: Pemahaman dan Aplikasi*. Grasindo: Jakarta.
- Hagen, W. v. (2008). *Professional Xen® Virtualization*. Indiana: Wiley Publishing, Inc.

- Hall, J. A. (2001). *Sistem Informasi Akuntansi, Edisi Ketiga*. Jakarta: Salemba Empat.
- Hutajulu, B. M. (2015). Membangun Jaringan Komputer Nirkabel Dengan Pengoperasian Sistem Operasi Unix Dengan Implementasi Ipv6 Pada Freebsd. *Faktor Exacta* 8(1), 23-36.
- idwebhost. (2018, 05 2018). Retrieved from <https://idwebhost.com/blog/tips-keren/apa-yang-dimaksud-dengan-server/>
- Ikhwan, S., & Elfitri, I. (2014). Analisa Delay Yang Terjadi Pada Penerapan Demilitarized Zone (DMZ) Terhadap Server Universitas Andalas. *Jurnal Nasional Teknik Elektro, Vol: 3 No. 2*, 118-124.
- Jepersen, H. (2014). *Konsep Sistem Informasi*. Yogyakarta: Deepublish.
- Jogiyanto, H. M. (2005). *Analisis dan Desain Sistem Informasi*. Yogyakarta: ANDI.
- Kertahadi. (2007). *Sistem Informasi Manajemen*. Jakarta: PT Pustaka Binaman Pressindo.
- Kessin, Z. (2012). *Programming HTML 5 Applications*. O'Reilly Media, Inc.
- Lex, D. S. (2017). Rancang Bangun Ip Public Berbasis VPN Server Dan Port Forwarding Untuk Mail Server Pada CV Pacific Computer Batam.
- Michrandi. (2014, 11). Retrieved from <https://michrandi.staff.telkomuniversity.ac.id/files/2014/11/OS-1-SMC.pdf>
- Nixon, R. (2009). *PHP, MySQL, & Javascript*. London: O'Reilly Media.
- Nugroho, A. (2011). *Perancangan dan Implementasi Sistem Basis Data*. Yogyakarta: ANDI.
- Nutanix. (2018). *Nutanix Official Website*. Retrieved from <https://www.nutanix.com>

- Otto, M. (2011). *Bootstrap from Twitter*. Retrieved from Twitter for Development:
<https://blog.twitter.com/developer/enus/a/2011/bootstrap-twitter.html>
- Powers, D. (2012). *Beginning CSS3: Mastering The Language of Web Design*. Apress.
- Silitonga, J., Suswaini, E., & Kurniawan, H. (2013). Pendaftaran Mahasiswa Baru Berbasis Mobile (Studi Kasus : Universitas Maritim Raja Ali Haji). 1-3.
- Supriyanto, A. (2007). *Pengantar Teknologi Informasi*. Jakarta: Salemba Infotek.
- Suwandi, Y. (2010). *Pembangunan Sistem Informasi Pengolahan Data Pada Pelatihan Dan Kursus TRP Di LKP Anglia*. Bandung.
- VMware. (2006). *Virtualization Overview*. Palo Alto: VMware Inc.