BAB II

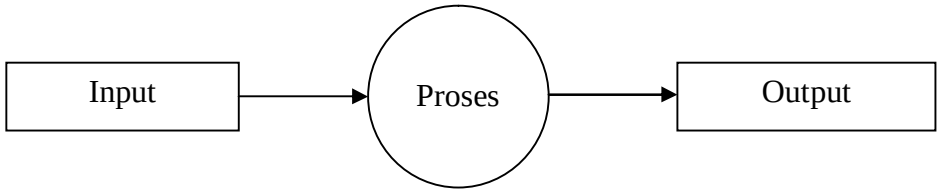
LANDASAN TEORI

2.1 Pengertian Sistem Informasi

Sebelum mengetahui pengertian sistem informasi secara keseluruhan, pertama kali yang harus dikenal adalah istilah-istilah yang tercakup di dalam tiap kata, secara berturut-turut berikut ini akan dijelaskan arti sistem dan informasi.

2.1.1 Bentuk Umum Sistem

Bentuk umum dari suatu sistem terdiri dari atas masukan *(input)*, proses dan keluaran *(output)*, dalam bentuk umum sistem ini terdapat satu atau lebih masukan yang akan diproses dan akan menghasilkan suatu keluaran



Gambar. 2.1 Bentuk Umum Sistem

2.2 Konsep Dasar Sistem Informasi

2.2.1 Sistem

Sistem adalah seperangkat elemen yang membentuk suatu kegiatan/prosedur/bagian pengolahan dalam mencapai tujuan bersama dengan mengoperasikan data pada waktu tertentu untuk menghasilkan suatu informasi.

Beberapa ciri dari sistem yaitu sebagai berikut:

1. Mengarah Pada Tujuan

   Cara kerja sistem ini adalah merangkaikan dan mengkoordinasikan fakta-fakta untuk mencapai tujuan dengan menggunakan aturan-aturan tertentu.

2. Merupakan Suatu Keseluruhan

   Sistem merupakan suatu keseluruhan yang bulat dan utuh, dimana tujuan masing-masing dari bagian yang membentuk sistem akan saling menunjang dan mencapai tujuan dari sistem secara keseluruhan, dan ini berarti bahwa pencapaian tujuan dari salah satu bagian tidak dapat dilakukan dengan mengabaikan pencapaian tujuan dengan bagian yang lainnya.

3. Adanya Keterbatasan

   Sistem memiliki sifat yang terbuka, dimana suatu sistem dapat berinteraksi dengan sistem lainnya yang lebih besar.

4. Adanya Proses Transformasi

   Suatu sistem mempunyai atau melakukan proses transformasi kegiatan yang mengubah suatu *input* atau masukan menjadi suatu *output* untuk mencapai tujuan

5. Saling Berkaitan

   Sistem terdiri dari elemen-elemen yang saling berkaitan satu elemen dengan elemen yang lain

2.2.1.1 Klasifikasi Sistem

Sistem dapat dikelompokkan atau di klasifikasikan menjadi beberapa sudut pandang diantaranya adalah sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak *(abstract sistem)* dan sistem fisik *(physical sistem)*, sistem abstrak terbentuk dari gagasan-gagasan atau konsep-konsep saling berkaitan. Sistem fisik didefinisikan sebagai suatu sistem yang unsur-unsur pendukungnya secara fisik dapat dilihat atau dirasakan dan memiliki keterkaitan antara satu unsur dengan unsur lainnya.
2. Sistem diklasifikasikan sebagai sistem alamiah dan sistem buatan manusia *(human made sistem)*. Yaitu sistem alamiah merupakan sistem yang terjadi melalui proses alam, tidak dibuat manusia, sedangkan sistem buatan manusia merupakan sistem yang dirancang oleh manusia, sistem manusia yang melibatkan interaksi manusia dengan mesin (*Komputer*) disebut *human macine sistem.*
3. Sistem diklasifikasikan sebagai sistem tertentu *(deterministic sistem)* dan sistem tak tentu *(probabilistic sistem)*, sistem tertentu beroprasi dengan tingkah laku yang sudah diprediksi, Interaksi diantara bagian-bagian nya dapat dideteksi dengan pasti, sehingga keluaran dari sistem dapat diramalkan. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. Dan yang terakhir adalah klasifikasi sistem sebagai sistem tertutup (*closed sistem)* dan sistem terbuka *(open sistem)*, dimana sistem tertutup tidak

terpengaruh oleh lingkungan lainnya, sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sebaliknya sistem terbuka merupakan sistem yang berhubungan dan terpengaruh dengan lingkungan luar.

Pengertian sistem itu sendiri sangatlah luas dan bermacam-macam disesuaikan dengan pendekatan dalam mendefinisikan sistem itu sendiri.

2.2.2 Informasi

Definisi umum informasi dalam pemakaian sistem informasi adalah data yang telah diolah menjadi sebuah bentuk yang lebih berguna dan lebih berarti bagi si pemakai dalam mengambil keputusan. Jadi sistem pengolahan informasi yaitu pengolahan data dari bentuk tak berguna menjadi berguna (informasi) bagi penerimanya.

Dalam mendapatkan suatu informasi, informasi tersebut harus yang berkualiatas, yaitu yang memiliki ciri-ciri sebagai berikut:

1. Akurat

   Berarti informasi harus bebas dari kesalahan-kesalahan yang biasanya terjadi dan selain itu harus jelas maksud dan tujuannya sehingga *output/*keluaran bisa dipertanggung jawabkan.

2. Tepat Waktu

   Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat (usang) informasi yang usang tidak akan mempunyai nilai yang baik, sehingga jika digunakan sebagai dasar dalam pengambilan keputusan akan berakibat

fatal atau mengalami kesalahan dalam pengambilan keputusan dan tindakan

3. Relevan

   Berarti infornasi tersebut mempunyai manfaat dan informasi yang diterima si pemakai dan pemakai lainnya bisa berbeda-beda.

Dari uraian tersebut diatas maka sistem informasi dapat didefinisikan sebagai suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan data (kejadian), mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu sistem dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Sistem informasi dapat dikatakan sebagai suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam suatu sistem organisasi untuk mencapai suatu tujuan yaitu menyajikan iformasi.

Kegiatan sistem informasi mencakup

1. *Input*, menggambarkan kegiatan untuk menyediakan data untuk diproses
2. Proses, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah
3. *Output*, suatu kegiatan untuk menghasilkan laporan proses diatas tesebut
4. Penyimpanan, suatu kegiatan untuk memelihara dan menyajikan data
5. Kontrol, suatu aktifitas untuk menjamin bahwa sistem informasi tersebut sesuai yang diharapkan

2.3 Komponen Sistem Informasi

Sistem informasi menpunyai beberapa komponen yaitu :

1. Perangkat keras *(Hardware)* yaitu semua alat komputer yang secara fisik dapat dilihat dengan jelas seperti: *keyboard, harddisk, monitor*, alat *printer*, *mouse*, dan lain-lain.
2. Perangkat lunak *(Software)* yaitu semua perangkat komputer yang dapat membuat perangkat keras komputer bekerja menjalankan fungsinya
3. Manusia *(Brainware)* yaitu orang yang mengoperasikan komputer.

2.4 Alat-alat Pengembangan Sistem

Pengembangan sistem merupakan hal yang penting bagi kelangsungan sistem itu sendiri. Pengembangan sistem adalah suatu upaya untuk menjaga efektivitas sistem dalam memenuhi kebutuhan pengguna sistem. Pengembangan sistem dapat berarti menyusun suatu sistem yang baru untuk mnggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang ada.

Alat –alat yang digunakan dalam pengembangan sistem antara lain adalah:

1. *Data Flow Diagram* (DFD)
2. Kamus Data (*Data Dictionary*)

2.4.1 Diagram Arus Data (*Data Flow Diagram*)

Data Flow Diagram (DFD) digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan. DFD memperlihatkan suatu sistem dalam komponen–komponennya serta *interface*

(penghubung) antara komponen tersebut. Dalam memperlihatkan aliran data dan pengembangan suatu sistem yang ditinjau dari segi data yang ditampilkan dengan simbol dan aturan tertentu.

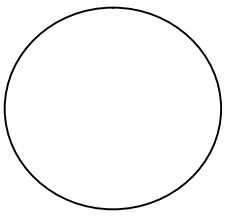
a. Arus Data (*Data Flow*)

Arus data ini menunjukan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem. Pada DFD, arus data digambarkan dengan tanda panah pada gambar berikut :

Gambar 2.2 Data Flow

b. Proses

Proses adalah kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Pada DFD, di bawah ini proses digambarkan dalam simbol lingkaran



Gambar 2.3 Proses

c. Kesatuan Luar *(External Entity)*

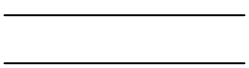
Kesatuan luar merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lain yang akan memberikan masukan *(input)* atau menerima keluaran *(output)* dari sistem. Di bawah ini kesatuan luar digambarkan dengan simbol kotak persegi panjang.



Gambar 2.4 External Entity

d. *File*

Digambarkan dengan dua garis horizontal



Gambar 2.5 File

Konsep dasar DFD dapat dilakukan dengan analisa *Top Down*, yaitu pemecahan sistem yang besar menjadi beberapa sub-sub sistem yang lebih kecil DFD terdiri dari :

1. Diagram Konteks

Merupakan gambaran sistem secara garis besar dan hanya memperlihatkan kelompok data *input* dan *output*. Diagram konteks adalah suatu diagram yang menggambarkan hubungan sistem dengan sejumlah komponen menjadi bagian penting dalam mendukung keberadaan sistem tersebut, yang akan memberikan

masukan atau diberikan keluaran dari sistem sehingga data yang masuk dapat menjadi informasi bagi komponen-komponen tadi. Konteks diagram merupakan *level* teratas dari diagram arus data .

2. Midle Level

Merupakan pemecahan dari tiap–tiap proses yang mempunyai fungsi sama. Pada *middle level* diagaram 0 dipecah menjadi diagram 1,2,3 dan seterusnya yang merupakan penguraian dari diagram konteks

3. Lowest Level

Merupakan pemecahan dari data *flow* yang ada pada *middle level*.Pemecahan tersebut masih tetap mempunyai fungsi yang sama dari *level* sebelumnya. Untuk *Lowest Level*, pemberian nomor diagram terdiri dari bagian *middle level.*

2.4.2 Kamus Data (*Data Dictionary*)

Kamus data *(Data Dictionary)* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data dapat mendefinisikan data yang mengalir pada sistem dengan lengkap. Kamus data dapat digunakan pada tahap analisa dan perancangan sistem. Pada tahap perancangan sistem, kamus data digunakan untuk merancang masukan *(input),* merancang laporan–laporan dan *database* .

Dengan adanya kamus data, didapat definisi-definisi dari bentuk–bentuk yang tidak dimengerti dalam DFD yaitu aliran data, *file*, proses dan elemen-elemen data. Arus data pada DFD bersifat global, hanya ditunjukan nama arus

datanya saja. Keterangan lebih lanjut tentang struktur dari arus data , secara lebih lengkap dapat dilihat di kamus data.

2.5 Pengertian Basis Data (*Database*)

Basis data adalah kumpulan *file-file* yang mempunyai kaitan antara satu *file* lain dengan *file* lain sehingga membentuk suatu bangunan data untuk menginformasikan suatu perusahaan/instansi dalam batasan tertentu basis data merupakan salah satu komponen penting dalam sistem informasi karena basis data adalah dasar untuk menyediakan informasi bagi para pemakai.

2.5.1 Tujuan Basis Data

Tujuan dari basis data erat hubungannya dengan masalah-masalah yang timbul dalam *file* basis data, yaitu:

1. Penyediaan akses yang *fleksibel*, yaitu untuk memberikan kemudahan dalam menampilkan semua informasi yang diperlukan
2. Pemeliharaan integritas data, basis data berisi *file* yang saling berkaitan yaitu dengan adanya f*ield* kunci yang menghubungkan kedua *file* tersebut
3. Keamanan data, basis data dapat mendefinisikan prosedur otoritas untuk memastikan bahwa data hanya dapat diakses oleh *user* yang berhak dan dapat mengijinkan *user* yang berbeda untuk melakukan tiap akses yang berbeda pada tahap data yang sama.
4. Menghilangkan ketergantungan data, Program yang berinteraksi dengan *DBMS (Database Management System)* relatif mandiri terhadap data

aktual dalam basis data. Hanya perubahan yang mungkin dilakukan terhadap struktur data tanpa membutuhkan perubahan terhadap program aplikasi yang telah ada

5. Mengurangi kerangkapan data
6. Penggunaan data bersama-sama, data yang sama dapat diakses atau digunakan oleh beberapa *user* pada saat bersamaan.Hal ini menggunakan sistem basis data, tiap aplikasi mempunyai *file* tersendiri, sehingga suatu data tunggal dalam basis data dapat digunakan untuk beberapa kegunaan
7. Standarisasi data, yaitu memberikan fasilitas-fasilitas kamus data untuk mendefiniskan nam-nama data secara rinci

2.5.2 Perancangan Basis Data

Pada perancangan model konseptual, penekanan tinjauan dilakukan pada struktur data dan relasi antar *file*. Pendekatan yang dilakukan pada perancangan model konseptual adalah menggunakan model data *relational*.
Terdapat dua buah teknik perancangan basis data, yaitu

1. Teknik Normalisasi
2. Teknik *Entity Relationship*

2.5.3 Teknik Normalisasi

Proses normalisasi merupakan pengelompokan data elemen menjadi table-tabel yang menunjukan *entity* dan relasinya. Pada proses normalisasinya

selalu diuji pada beberapa kondisi. Apabila terdapat kesulitan pada saat menambah/*insert*. menghapus/*delete*, mengubah/*update*, membaca/*retrieve* pada suatu *database* maka relasi tersebut dipecahkan pada beberapa tabel lagi.

Bentuk–bentuk Normalisasi :

1. Bentuk tidak normal *(unnormalizd form)*

Pada bentuk tidak normal, tidak ada keharusan mengikuti suatu format tertentu, data mungkin tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan kedatangannya.

2. Bentuk Normal kesatu (*1NF / first Normal Form)*

Bentuk normal kesatu mempunyai ciri-ciri sebagai berikut :

(a) Data dibentuk dalam *flat file (file* datar/rata)

(b) Data dibentuk dalam satu *record* demi satu *record* dan nilai dari *field-field* berupa *atomic value*

(c) *Set attribute* tidak boleh berulang-ulang atau bernilai ganda *(Multi value)*

3. Bentuk Normal Kedua (*2NF / Second Normal Form)*

Bentuk normal kedua mempunyai syarat yaitu bentuk data telah memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci harus bergantung secara fungsi pada kunci utama / *primary key*. Sehingga untuk membentuk normal kedua harus

ditentukan kunci-kunci *field*. Kunci *field* tesebut harus unik dan dapat mewakili *attribute* lain yang menjadi anggotanya.

4. Bentuk Normal Ketiga (*3 NF / Thrid Normal Form)*

Untuk membentuk normal ketiga maka relasi harus dalam bentuk normal kedua dan semua *attribute* bukan primer harus bergantung hanya pada *primary key* secara menyeluruh.

5. *Boyce –Codd Normal Form* (*BCNF*)

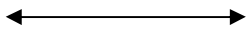
Pada *Boyce – Codde Normal Form*, relasi harus dalam bentuk normal kesatu dan setiap *attribute* harus bergantung fungsi pada *attribute superkey*.

2.5.4 *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram dibuat untuk merancang suatu basis data dengan memperlihatkan hubungan antara *entity* atau objek yang terlibat beserta atributenya. Untuk membantu gambaran relasi secara lengkap ada tiga macam relasi dalam hubungan *attribute* dalam suatu *file* :

1. *One to One Relationship*

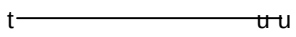
Hubungan antara *file* kesatu dengan *file* kedua adalah satu berbanding satu. Dengan simbol :



Gambar 2.6 One to one relationship

2. *One to many Relationship*

Hubungan antara *file* pertama dengan *file* kedua adalah satu berbanding banyak atau dapat pula dibalik banyak berbanding satu. Dengan simbol :

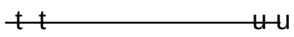


Gambar 2.7 One to many relationship

3. *Many to many Relationship*

Hubungan antara *file* pertama dan *file* kedua adalah banyak ke banyak.

Dengan simbol :



Gambar 2.8 Many to many relationship

2.6 Sekilas Mengenai *Microsoft Access*

Bila kita mempunyai daftar berisi nama dan alamat mahasiswa berarti kita telah mempunyai *database*. Demikian pula juga kita memiliki buku telepon di rumah, buku itu pun sebuah *database*. Ada banyak hal yang tidak mungkin atau sukar sekali dilakukan secara manual. Disinilah perlunya program komputer seperti *Microsoft Access* untuk mengelola *database*.

2.6.1 Jenis Data di dalam *Microsoft Access XP*

Jenis data yang dimiliki *Microsoft Access 97* adalah sebagai berikut :

1. *Text*

   Jenis data *text* (teks) digunakan untuk menyimpan data seperti nama, alamat, atau bilangan yang tidak membutuhkan perhitungan, seperti nomor telepon, kode pos, atau nomor serial produk.

2. *Memo*

   Jenis data *memo* digunakan untuk menyimpan data teks yang panjangnya lebih dari 225 karakter.

3. *Number*

   Jenis data *number* (bilangan) digunakan untuk menyimpan nilai numerik untuk perhitungan matematis, kecuali perhitungan yang menyangkut uang.

4. *Date/Time*

   Digunakan bagi *field* yang menampung data tanggal atau jam.

5. *Currency*

   Digunakan untuk menampung bilangan. Kelebihan *field* jenis ini dari *field number* adalah bahwa bilangan yang disimpan dalam *field* ini tidak akan kena pembulatan pada saat perhitungan.

6. *AutoNumber*

   Digunakan untuk membuat *field* yang secara otomatis akan memasukkan bilangan pada saat kita menambahkan sebuah *record*.

7. *Yes/No*

Membuat *field* untuk menampung dua jenis keadaan , ya atau tidak. *Field* ini menyimpan bilangan -1 untuk entri *Yes* dan 0 untuk entri *No*.

8. *OLE Object*

   Membuat *field* untuk menyimpan objek *OLE*, file dokumen, gambar, dan file lainnya yang dibuat dalam program lain.

9. *Hiperlink*

   Teks atau gabungan teks dan bilangan yang disimpan sebagai teks dan digunakan sebagai alamat *hiperlink*.

10. *Lookup Wizard*

    Membuat *field* yang menampilkan salah satu dari dua macam daftar yang memudahkan pemasukan data. Kedua macam daftar itu adalah :

    a. Daftar *Lookup* yang menampilakn nilai-nilai yang dirujuk dari sebuah tabel atau *query* yang telah ada.
    b. Daftar nilai yang menampilkan kumpulan nilai yang tidak bisa diubah yang kita masukkan saat kita membuat *field* ini.

2.6.2 Pembuatan *Database* baru dalam *Microsoft Access XP*

Di jendela *database*, ada enam tab yang bisa diplih, yaitu *Tables, Queries, Forms, Reports, Pages, Macros* dan *Modules*.

a. Tables

Untuk menyimpan data yang kita buat.

b. *Queries*

Select Query adalah jenis *query* yang paling umum. Dengan *select query* bisa didapatkan data dari satu atau beberapa tabel dan menampilkan hasilnya dalam sebuah *datasheet* dimana dapat juga memperbaharui *record-record* dengan beberapa pembatas.

c. *Forms*

Sebuah *form* membutuhkan sumber data yang berasal dari tabel atau *query*.

d. *Report*

Report adalah tampilan untuk melihat data pada tabel atau *query*. Berbeda dengan *form* yang biasa menerima masukan data, *report* hanya menampilkan data saja.

e. *Pages*

Pages adalah tampilan untuk membuka data *access pages* di dalam *microsort access database* atau *microsoft access project* yang digunakan untuk membuka atau mengubah *page* yang dipilih atau membuat *page* baru.

f. *Macros*

Makros bisa megotomasi beberapa tindakan seperti membuka *form* atau membuka *report*.

g. *Modules*

Moduls adalah tampilan untuk di dalam *microsoft access database* atau *microsoft access project* yang digunakan untuk mengubah dengan memilih modul atau membuat modul baru.

2.7 Sekilas Mengenai *Visual Basic 6.0*

Perangkat lunak yang digunakan untuk membuat implementasi dari sistem pada tugas akhir ini adalah Bahasa Pemograman *Visual Basic 6.0*. Bahasa Pemograman *Visual Basic 6.0* mrupakan Bahasa Pemograman yang digunakan untuk membuat aplikasi *Windows* yang berisi grafis (*GUI – Graphical User Interface*), atau pemograman yang menggunakan tampilan grafis sebagai alat komunikasi dengan pemakainya.

Sejarah pekembangan *Visual Basic* adalah sebagai berikut :

a. *Visual Basic* pertama kali diperkenelkan pada tahun 1991 yaitu program *basic* untuk *Dos* dan *Windows*
b. *Visual Basic 3.0* dirilis tahun 1993
c. *Visual Basic 4.0* dirilis pada akhir tahun 1995
d. *Visual Basic* yang digunakan untuk membuat aplikasi adalah versi 6.0 yang dirilis pada akhir tahun 1998.

Pada *Visual Basic* , langkah – langkah untuk mengembangkan aplikasi :

1. Membuat *user interface*/tampilan
2. Mengatur properti
3. Menulis kode program

Lingkungan pemograman *Visual Basic 6.0* mengandung sarana pemograman yang dapat membantu Anda dalam membuat program *Visual Basic* diantaranya :

a. Baris Menu

   Baris menu menyediakan akses kepada sebagian besar perintah yang mengendalikan lingkungan pemograman ini. Menu dan perintah sama fungsinya seperti pada pemograman – pemograman berbasis *windows* lainnya, Anda dapat mengaksesnya dengan menggunakan *keyboard* atau mouse.

b. Toolbar

   Toolbar yaitu sekumpulan tombol yang berfungsi sebagai tombol cepat untuk menjalankan perintah dan mengendalikan lingkungan pemograman *Visual Basic*.

c. Taskbar windows

   Taskbar Windows digunakan untuk berpindah antara berbagai komponen Visual Basic dan untuk mengaktifkan program Windows lainnya.

d. Toolbox

   Toolbox biasanya terletak di sisi sebelah kiri layar. *Toolbox* mengandung kontrol – kontrol yang biasa digunakan untuk menambah gambar, label, tombol, kotak daftar, baris pengulung, menu, serta bentuk – bentuk geometris antar muka.

e. *Project Windows*

Jendela *Project* biasanya terletak di sisi sebelah kiri atas layar. Jendela *Project* berisi semua file yang digunakan pada proses pemograman dan menyediakan akses ke file tersebut menggunakan dua tombol khusus : *View Code* dan *View Object*.

f. *Propertis Windows*

Jendela Propetis biasanya terletak di sisi kanan tengah layar. Jendela propertis digunakan untuk mengubah karakteristik atau setting propertis dari elemen yang terdapat pada form. Setting propertis adalah kualitas objek – objek yang terdapat pada antarmuka.

g. *Form Layout Windows*

Jendela *Form Layout* biasanya terletak di sisi kanan bawah layar. Jendela *form layout* digunakan untuk melihat jumlah Object dalam layar yang kita tampilkan, juga kita dapat menampilkan garis relasi dengn memilih *Resolution Guides.*