



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Dalam melakukan penelitian ini, terdapat langkah-langkah penelitian yang digunakan sebagai berikut.

1. Studi Literatur

Tahap pertama dalam penelitian ini adalah melakukan studi literatur mengenai sistem pendukung keputusan, *data mining*, dan algoritma C4.5 menggunakan referensi buku, jurnal, artikel, skripsi dan berbagai sumber untuk mendukung perancangan dan pembangunan sistem. Literatur tersebut dijadikan pedoman dalam pembuatan skripsi.

2. Pengumpulan Data

Data diperoleh dengan cara meminta data dari Divisi TI Universitas Multimedia Nusantara (UMN) dan pihak psikolog UMN. Data yang didapat berupa nilai raport mahasiswa semester 1 dan 2 kelas 12 pada saat SMA, program studi, minat, jurusan, dan nomor induk mahasiswa (NIM) dari data mahasiswa angkatan 2015/2016. Sebelum data tersebut dapat digunakan sebagai data *training*, terdapat beberapa proses dalam melakukan pengolahan data agar data tersebut dapat digunakan sesuai dengan target penelitian. Proses yang pertama adalah melakukan pemilihan data, data yang digunakan adalah data mahasiswa yang menggunakan jalur masuk universitas melalui jalur akademik. Proses yang kedua adalah menjumlahkan data nilai semester 1 dan semester 2 untuk mendapatkan nilai rata-rata. Setelah didapatkan nilai rata-rata, nilai tersebut dikonversikan dengan konversi nilai sebagai berikut. Nilai diatas 85 digolongkan

sebagai A, nilai antara 80 dan 85 digolongkan sebagai A-, nilai antara 75 dan 80 digolongkan sebagai B+, nilai diantara 70 dan 75 digolongkan sebagai B, dan nilai di bawah 70 digolongkan sebagai C. Jika terdapat data yang kosong atau *missing values*, dalam kasus ini adalah tidak terdapatnya nilai Fisika bagi penjurusan SMA IPS, maka dilakukan pengisian data dengan cara memilih data yang terendah dari seluruh nilai Fisika yang ada. Setelah semua proses tersebut selesai, maka data tersebut dapat digunakan sebagai data *training*.

3. Perancangan Desain Sistem

Setelah mendapatkan semua data yang dibutuhkan, dilakukan perancangan desain sistem yang akan digunakan untuk melakukan pendukung keputusan bagi para calon mahasiswa dengan menggunakan algoritma C4.5.

4. Pembangunan Sistem dan Penerapan Algoritma

Melakukan pembangunan sistem berbasis *desktop* dengan menggunakan bahasa pemrograman berbasis C#. Penerapan algoritma terlebih dahulu dilakukan dengan cara melakukan perhitungan secara manual dan kemudian diterapkan ke dalam sistem. Setelah berhasil diterapkan terhadap sistem, maka algoritma perhitungan manual dan perhitungan melalui sistem akan dibandingkan. Jika perbandingan tersebut menghasilkan hasil yang sama, maka dapat dikatakan penerapan algoritma berhasil.

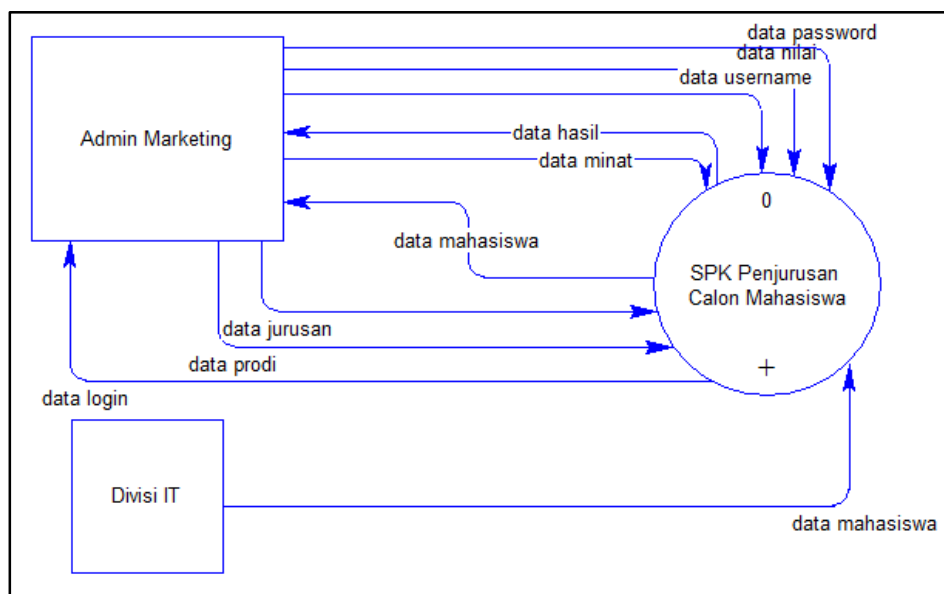
5. Uji Coba

Sistem yang telah selesai dibangun digunakan untuk mengolah data *training* maupun data *testing*. Metode yang digunakan untuk uji coba keakuratan sistem yang telah dibangun adalah dengan menggunakan metode *Confusion Matrix* dan *Cross Validation*. Dari hasil pengujian berdasarkan metode-metode tersebut maka akan didapatkan hasil tingkat keakuratan sistem dalam bentuk persentase.

3.2 Perancangan Sistem

3.2.1 Data Flow Diagram (DFD)

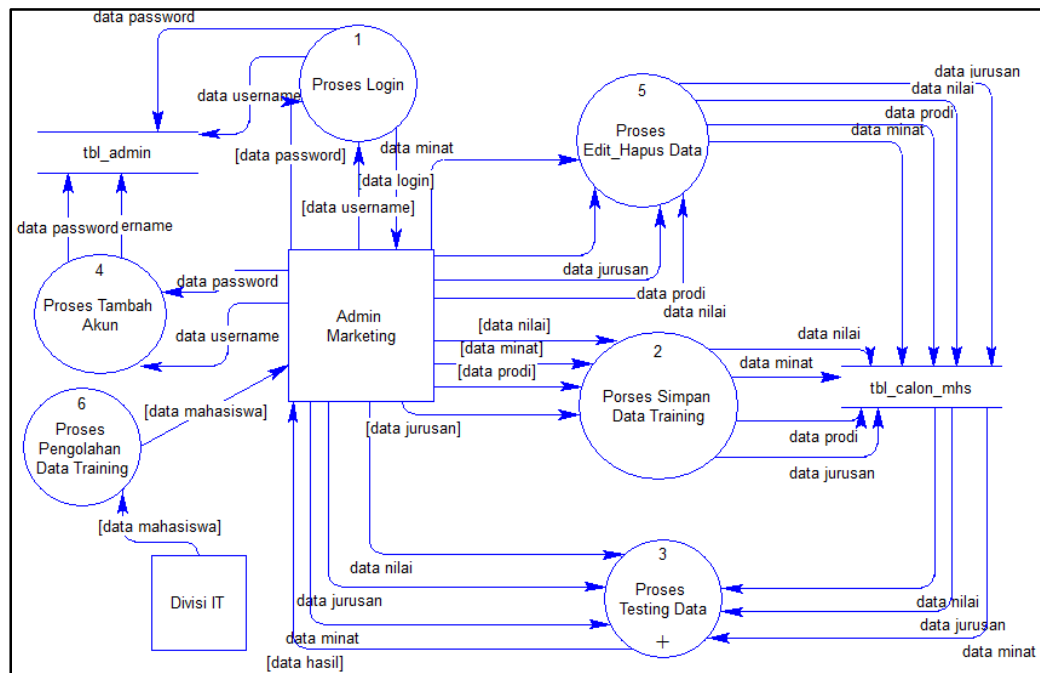
Data Flow Diagram (DFD) merupakan penjelasan dari alur data yang terjadi pada sistem pendukung penjurusan program studi. Alur data tersebut dapat dilihat pada beberapa gambar berikut ini.



Gambar 3.1 DFD Level 0

Gambar 3.1 menggambarkan proses alur data dari Sistem Pendukung Keputusan (SPK) Calon Mahasiswa yang terdiri atas satu proses, yaitu proses SPK Penjurusan Calon Mahasiswa. Pada DFD Level 0 ini terdapat dua buah entitas,

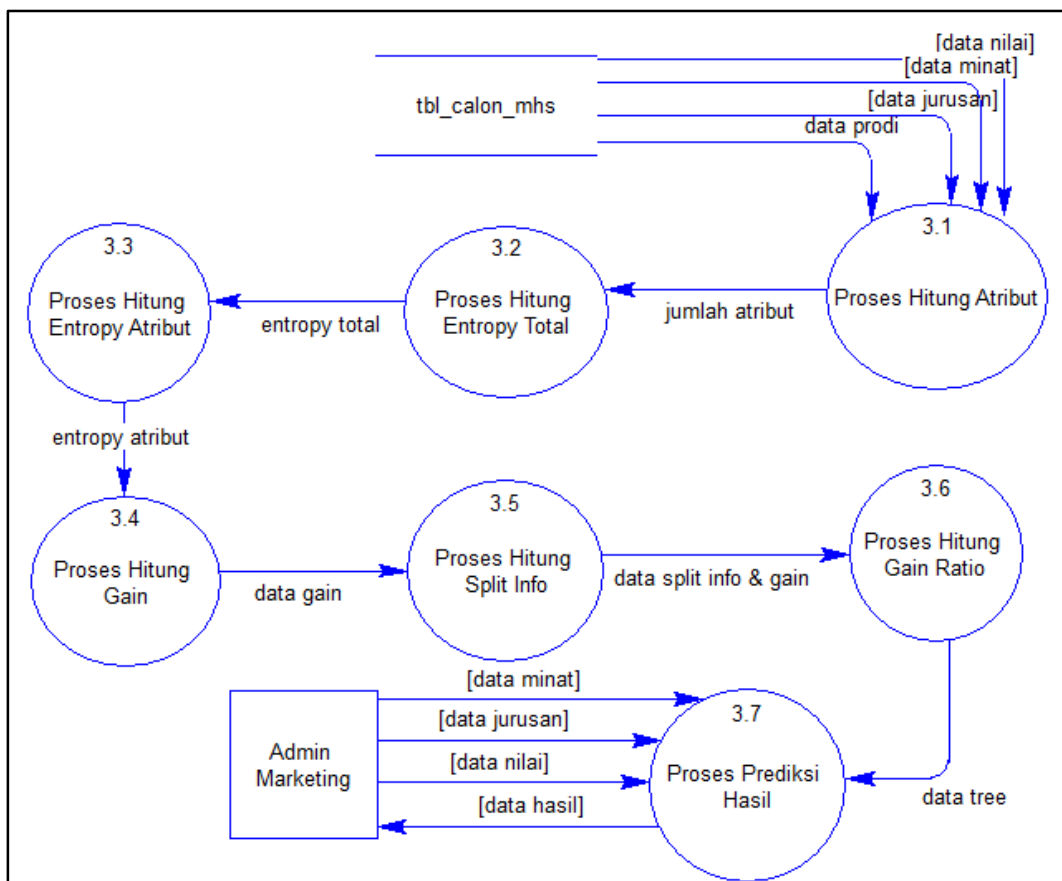
yaitu Admin Marketing dan Divisi IT. Admin Marketing mengalirkan data berupa data *password*, data nilai, data *username*, data minat, data jurusan, dan data prodi. Setelah sistem pendukung keputusan telah selesai, maka sistem mengirimkan data berupa data hasil ke pada Admin Marketing, sedangkan Divisi IT memberikan data mahasiswa untuk dijadikan data *training*.



Gambar 3.2 DFD Level 1

Gambar 3.2 menunjukkan DFD Level 1 yang menggambarkan aliran data pada proses *login*, simpan data *training*, *testing* data, tambah akun, *edit* dan hapus *database*. Pada proses *login*, data yang dikirim berupa *username* dan *password* yang akan dikirim ke dalam *database* dan kemudian akan mengembalikan data berupa data *login* yang bertujuan untuk memberikan izin kepada Admin Marketing untuk dapat masuk ke dalam aplikasi lebih lanjut atau tidak. Proses pengolahan data *training* merupakan proses pembersihan data seperti menghilangkan *missing values* atau menghapus data yang tidak sesuai agar dapat digunakan sebagai data *training*. Proses simpan data merupakan proses yang terdapat data yang akan dimasukkan ke

dalam *database*, data tersebut berupa data *training*. Proses *testing* data merupakan proses yang didalamnya terdapat aliran data dari *database* yang berisikan data *training* dan terdapat data *testing* yang dimasukkan oleh Admin Marketing. Di dalam proses ini akan menghasilkan suatu pendukung keputusan yang akan diterima oleh Admin Marketing. Proses *edit database* dan hapus data merupakan proses yang bertujuan untuk melakukan modifikasi data terhadap *database* berupa *edit* data dan hapus data yang didapat dari data yang dikirimkan oleh Admin Marketing, dan proses tambah akun merupakan proses yang didalamnya terdapat data yang dikirim oleh Admin Marketing untuk melakukan penambahan akun yang selanjutnya akan disimpan ke dalam *database*.

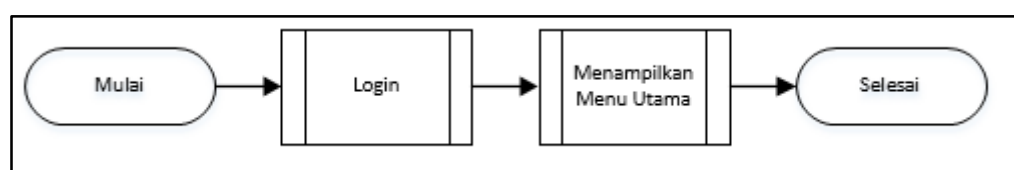


Gambar 3.3 DFD Level 2

Gambar 3.3 menunjukkan DFD Level 2 yang berisikan proses hitung atribut, hitung *entropy* total, hitung *entropy* atribut, hitung *gain*, hitung *split info*, hitung *gain ratio*, dan prediksi kelulusan. Dalam DFD level 2 ini terdapat satu buah entitas, yaitu Admin Marketing. Pada proses hitung jumlah atribut, data berasal dari tabel *tbl_calon_mhs*. Data yang dikirimkan berupa data *training* yang berisikan nilai, minat, jurusan, dan program studi. Kemudian data yang sudah didapat dari proses hitung jumlah atribut akan dilanjutkan ke proses hitung *entropy* total untuk menghasilkan *entropy* total. Setelah mendapatkan *entropy* total, proses dilanjutkan kepada proses hitung *entropy*. Dari proses ini didapatkan *entropy* dari tiap atribut yang digunakan untuk proses selanjutnya, yaitu proses hitung *gain*. Setelah mendapatkan *gain*, maka data dikirimkan kepada proses hitung *split info*. Dari proses *split info*, data kembali dilanjutkan kepada proses hitung *gain ratio* yang akan menghasilkan sebuah *tree* atau pohon keputusan. Setelah pohon keputusan telah siap, Admin Marketing memasukkan data *testing* kepada proses prediksi kelulusan berupa minat, jurusan, dan nilai. Dari proses prediksi hasil maka akan didapatkan keluaran berupa program studi.

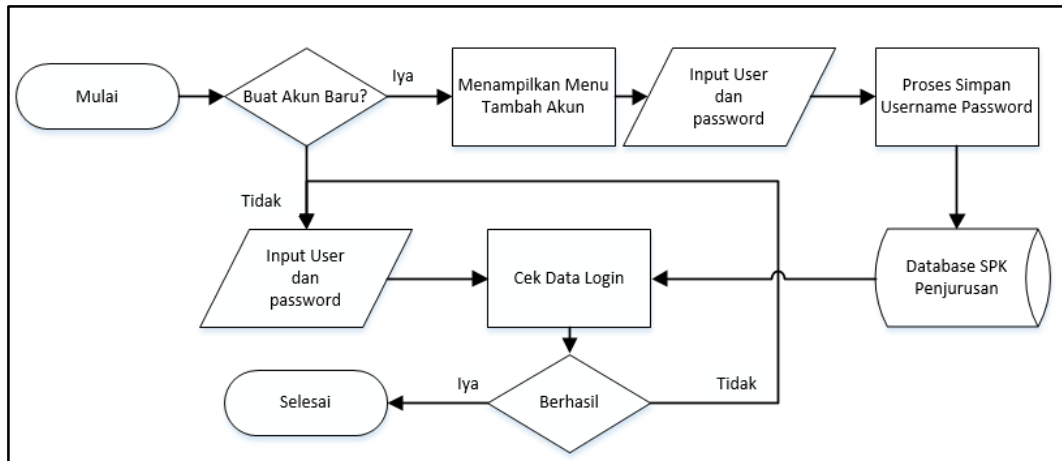
3.2.2 Flowchart Diagram

Flowchart diagram merupakan diagram yang bertujuan untuk memberikan gambaran mengenai urutan proses-proses yang terdapat dalam suatu sistem. Berikut rancangan *flowchart* dari SPK Pemilihan Jurusan Universitas.



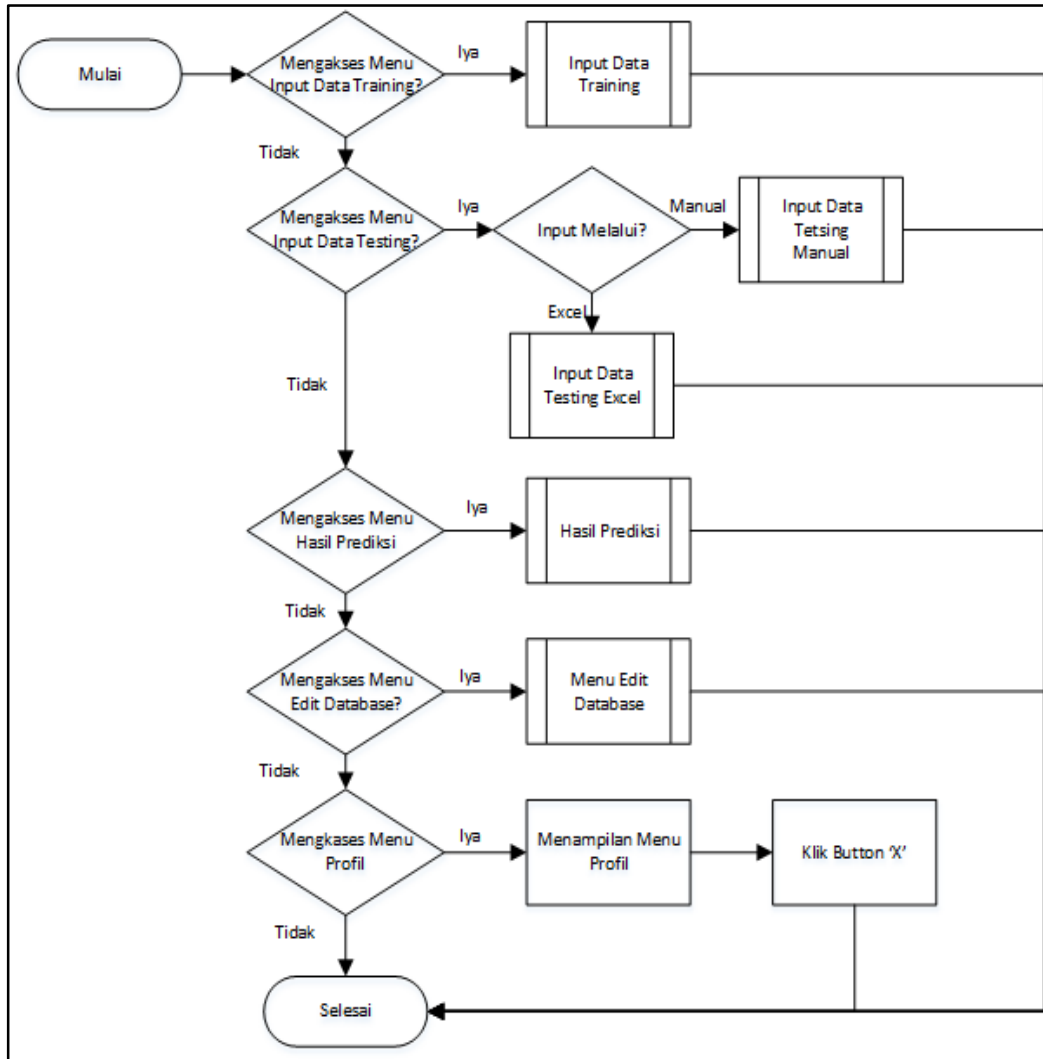
Gambar 3.4 *Flowchart* Sistem Pendukung Keputusan Program Studi

Gambar 3.4 menggambarkan *flowchart* Sistem Pendukung Keputusan Program Studi yang dimulai dengan proses *login*. Setelah proses *login* berhasil akan dilanjutkan dengan proses menampilkan menu utama.



Gambar 3.5 *Flowchart Login*

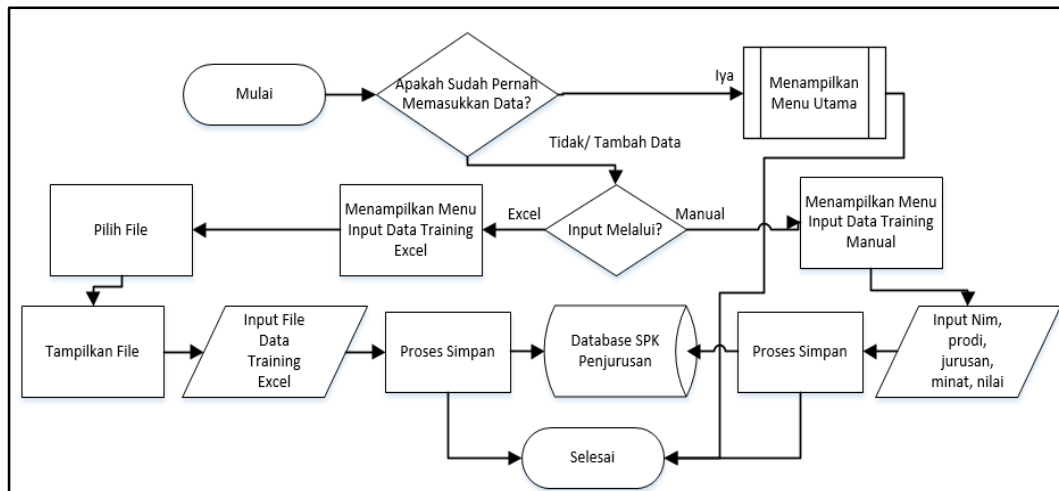
Gambar 3.5 menggambarkan *flowchart login* Sistem Pendukung Keputusan Program Studi. Di dalam *flowchart* ini terdapat pilihan untuk membuat akun baru. Jika *user* ingin menambahkan akun baru, maka *user* akan dipindahkan ke dalam menu Tambah Akun yang bertujuan untuk mengisi data akun baru. Setelah *user* selesai mengisi data, data akan diproses ke dalam *database* untuk disimpan. Jika *user* tidak ingin menambahkan akun, maka *user* dapat melakukan proses *login* dan jika berhasil maka akan dilanjutkan ke proses selanjutnya.



Gambar 3.6 Flowchart Menampilkan Menu Utama

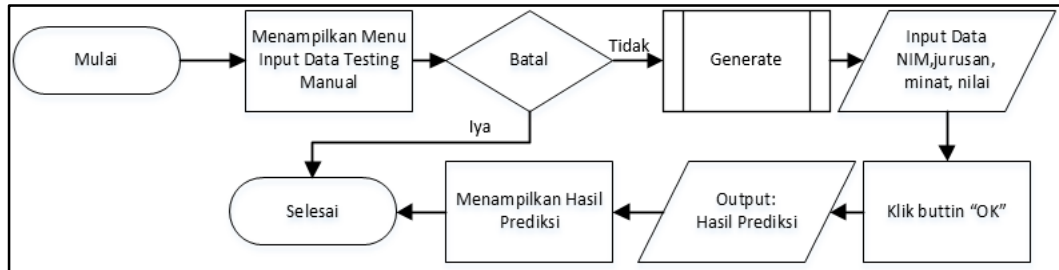
Gambar 3.6 menggambarkan *flowchart* Menampilkan Menu Utama pada sistem pendukung penjurusan program studi untuk menampilkan menu utama pada sistem. Jika *user* ingin mengakses menu *input data training*, maka *user* akan diarahkan ke proses *input data training*. Jika *user* ingin mengakses menu *input data testing* maka *user* akan dihadapkan dengan pilihan *input data testing* melalui *excel* atau *input data testing* melalui *input manual*. Jika *user* ingin mengakses menu hasil prediksi maka *user* akan diarahkan ke *form* baru hasil prediksi. Jika *user* ingin mengakses menu *edit database* maka *user* akan diarahkan ke *form* baru *edit*

database. Jika *user* ingin mengakses menu profil maka *user* akan diarahkan ke *form* baru profil.



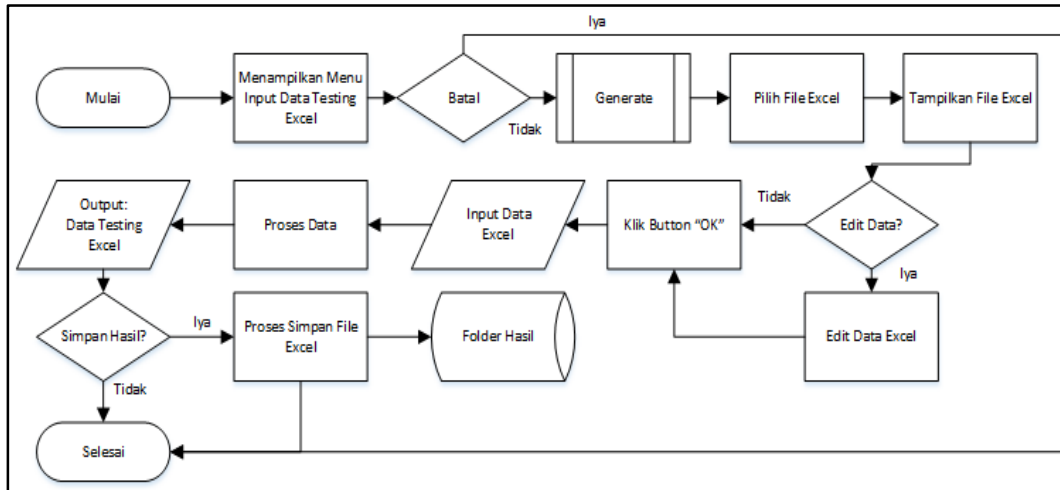
Gambar 3.7 Flowchart Input Data Training

Gambar 3.7 menggambarkan *flowchart* *Input Data Training* untuk menampilkan proses memasukkan data *training* pada sistem. Sebelum menjalankan proses memasukkan data, sistem akan mengeluarkan pesan berupa pertanyaan “Apakah Sudah Pernah Memasukkan Data?”. Jika *user* menjawab dengan pilihan “Iya”, maka *user* akan diarahkan kembali ke menu utama. Jika *user* menjawab dengan pilihan “Tidak/Tambah Data”, maka sistem akan mengeluarkan pesan berupa pertanyaan “Input Data Melalui?”. Jika *user* menjawab dengan pilihan “Manual”, maka *user* akan diarahkan ke *form* baru untuk memasukkan data *training* secara manual dan akan dimasukkan ke dalam *database*. Jika *user* menjawab pilihan dengan “Excel”, maka *user* akan diarahkan ke *form* baru untuk memasukkan data *training* dengan menggunakan file berformat *excel* melalui proses pilih *file*, dan tampilkan *file*. *File* yang dimasukkan akan disimpan ke dalam *database*.



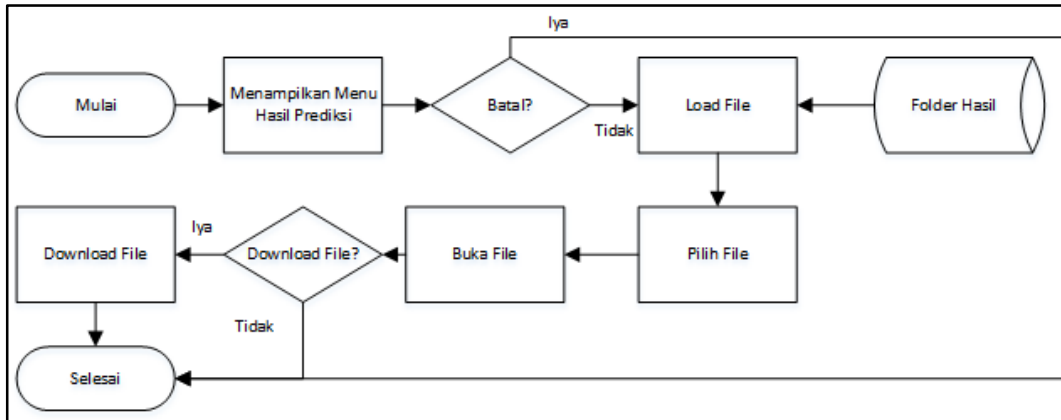
Gambar 3.8 *Flowchart Input Data Testing Manual*

Gambar 3.8 menggambarkan *flowchart Input Data Testing Manual* untuk menampilkan proses memasukkan data *testing* secara manual. Sebelum menjalankan proses memasukkan data, terdapat proses menampilkan menu input data *testing* manual yang berupa *form* baru. Selanjutnya, jika *user* ingin membatalkan aksi memasukkan data *testing* maka proses akan selesai. Jika *user* tidak ingin membatalkan aksi, maka *user* diharuskan untuk melakukan proses *generate* data terlebih dahulu. Setelah proses *generate* data selesai, maka *user* dapat memasukkan data dan setelah selesai memasukkan data, terdapat proses klik button “OK”, untuk mengeluarkan hasil prediksi.



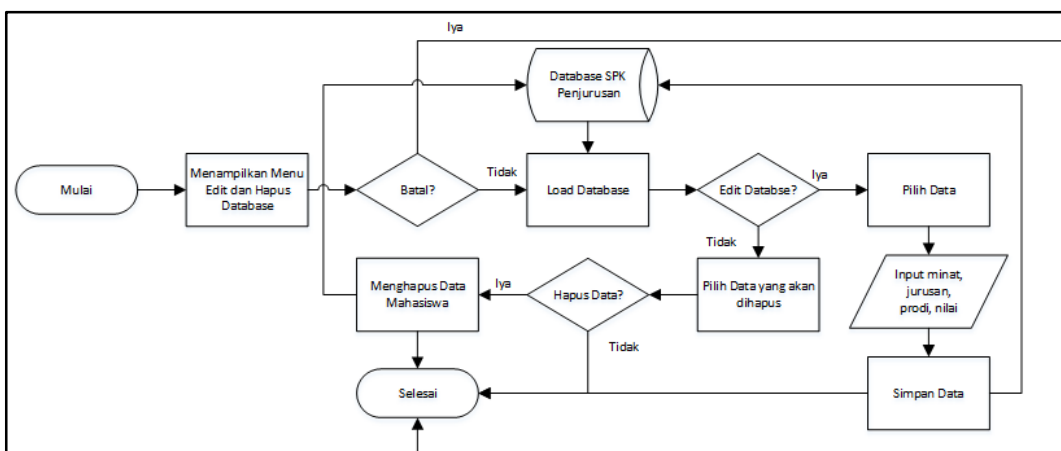
Gambar 3.9 Flowchart Input Data Testing Excel

Gambar 3.9 menggambarkan *flowchart* Input Data Testing Excel yang bertujuan untuk memasukkan data *testing* dengan *file* berformat *excel*. Sebelum melakukan proses memasukkan data *testing*, terdapat proses menampilkan menu input data *testing excel* berupa *form* baru. Jika *user* ingin membatalkan aksi memasukkan data, maka proses akan selesai. Jika tidak *user* diharuskan untuk melakukan proses *generate* data terlebih dahulu. Ketika proses *generate* data telah selesai, dilanjutkan dengan proses pilih *file* dan tampilkan *file*. Terdapat pilihan untuk melakukan perubahan terhadap isi *excel*. Jika ingin melakukan perubahan terhadap isi *file*, maka akan dilakukan proses *edit* data *excel*, jika tidak maka terdapat proses klik button “OK”, untuk memasukkan *file* ke dalam sistem. Setelah *file* dimasukkan, maka akan berjalan proses untuk melakukan proses terhadap data yang tersedia dan akan mengeluarkan hasil prediksi. Terdapat pilihan untuk menyimpan hasil, jika *user* memilih untuk menyimpan hasil, maka data tersebut akan disimpan ke dalam *folder* hasil, jika tidak ingin menyimpan hasil maka proses akan selesai.



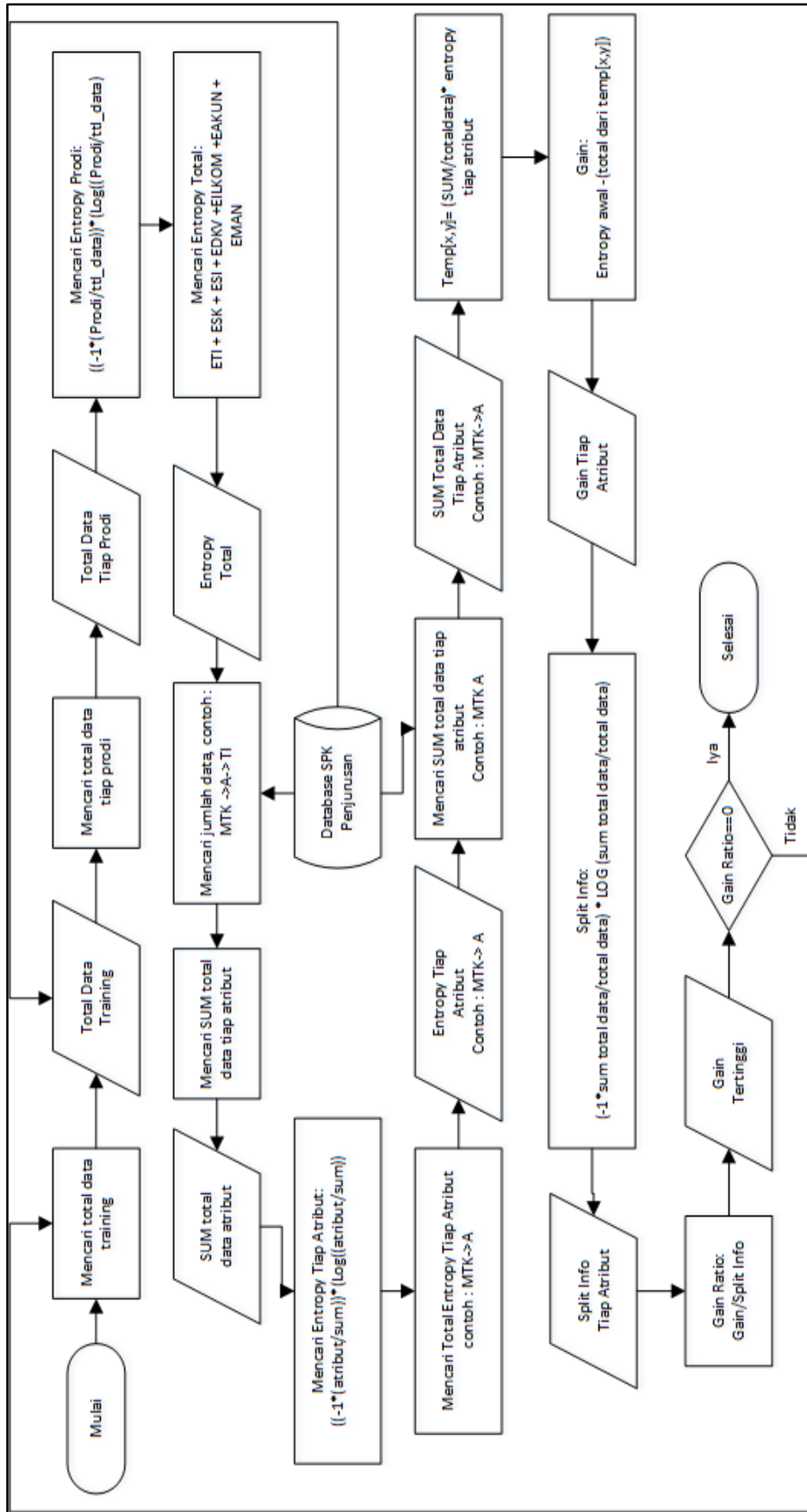
Gambar 3.10 *Flowchart* Hasil Prediksi

Gambar 3.10 menggambarkan *flowchart* Hasil Prediksi yang dimulai dengan menampilkan *file* hasil prediksi yang telah disimpan pada *folder* hasil. Sebelum melakukan proses lihat hasil prediksi, sistem akan melakukan proses menampilkan menu hasil prediksi yang berupa *form* baru. Jika *user* ingin membatalkan aksi maka proses akan selesai. Jika *user* ingin melanjutkan aksi maka akan terdapat proses *load file* yang bertujuan untuk menampilkan *file* yang telah disimpan pada *folder* hasil. Setelah melakukan proses pilih *file*, dilanjutkan dengan melakukan proses buka *file*. Terdapat pilihan untuk melakukan *download file*, jika *user* menginginkan *file* untuk di-*download* maka *file* akan ter-*download*, jika tidak maka proses akan selesai.



Gambar 3.11 *Flowchart* Menu Edit Database

Gambar 3.11 menggambarkan *flowchart* Menu *Edit Database* yang bertujuan untuk melakukan perubahan dan penghapusan data terhadap *database*. Sebelum melakukan perubahan atau penghapusan data, sistem akan melakukan proses menampilkan menu *edit* dan hapus *database* yang berupa *form* baru. Jika *user* ingin membatalkan aksi, maka proses akan selesai, jika tidak terdapat proses *load database* yang bertujuan untuk menampilkan data dari *database*. Setelah itu terdapat pilihan *edit database*, jika *user* memilih untuk melakukan aksi *edit database*, maka terdapat proses pemilihan data yang akan dilakukan perubahan dan setelah selesai data yang diubah akan dimasukkan ke dalam *database*. Jika *user* tidak ingin melakukan perubahan data, terdapat pilihan untuk melakukan hapus data yang terdapat pada *database*. Jika sudah memilih data yang akan dihapus, sistem akan memberikan pilihan “Hapus Data?”. Jika menjawab pertanyaan dengan pilihan “Iya”, maka data akan terhapus dari *database*, jika tidak maka sistem akan menggagalkan aksi penghapusan data.

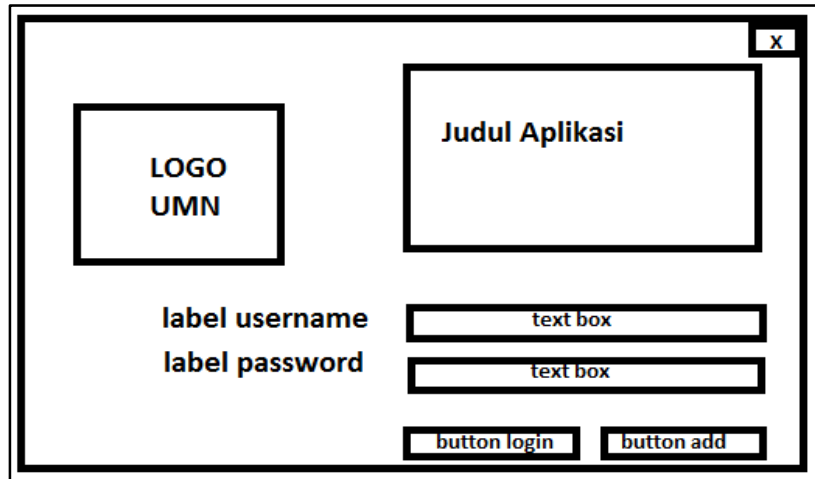


Gambar 3.12 Flowchart Generate

Gambar 3.12 menggambarkan *flowchart Generate* yang bertujuan untuk membentuk suatu pohon keputusan dengan menggunakan algoritma C4.5. Sebelum pohon keputusan terbentuk terdapat proses mencari total data *training* yang didapat dari *database*, setelah mendapatkan total data *training*, proses dilanjutkan dengan mencari total data tiap program studi, setelah itu dilanjutkan dengan proses menghitung *entropy* tiap atribut dan proses menghitung *entropy* total. Proses dilanjutkan dengan mencari total jumlah data tiap atribut dilanjutkan dengan mencari *sum* dari total data tiap atribut sebelumnya. Dengan menggunakan data yang didapat dari proses sebelumnya, dilakukan proses perhitungan mencari *entropy* tiap atribut dan perhitungan total atribut untuk mencari nilai *gain*. Proses dilanjutkan dengan melakukan proses perhitungan untuk mencari nilai *split info*. Proses perhitungan *gain ratio* didapat dari melakukan pembagian antara *gain* dan *split info*. Jika *gain* bernilai nol, maka cabang pohon akan selesai, jika tidak maka akan dilakukan pengulangan dari proses menghitung jumlah data *training* berdasarkan *gain* tertinggi.

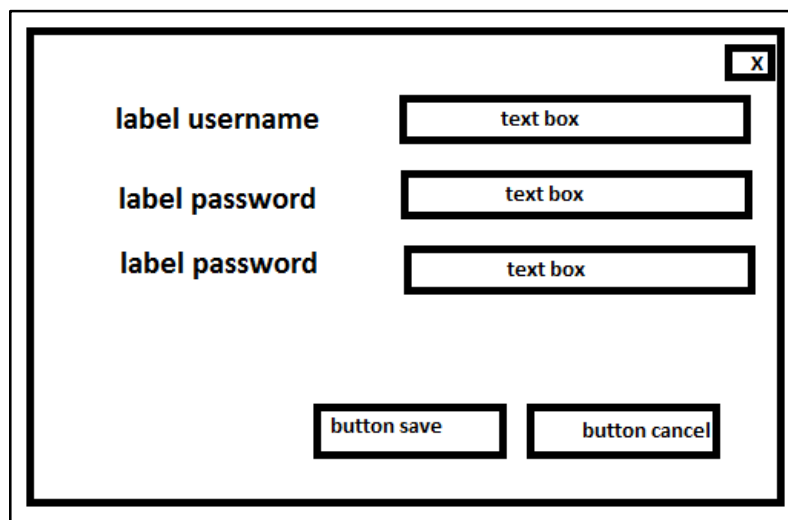
3.2.3 Rancangan Antarmuka Sistem

Rancangan antarmuka aplikasi SPK Penjurusan Universitas merupakan rancangan tampilan aplikasi yang akan ditampilkan kepada pengguna aplikasi. Tampilan aplikasi diawali dengan sebuah *form login* seperti pada Gambar 3.13.



Gambar 3.13 Rancangan Antarmuka *Form Login*

Gambar 3.13 merupakan rancangan antarmuka dari *form login* SPK Program Studi. *Form* terdiri dari *button* 'logo UMN', 'login', dan 'add' yang bertujuan untuk melakukan *login* terhadap sistem. *Form* ini merupakan *form* yang akan muncul pada saat sistem pertama kali dijalankan.



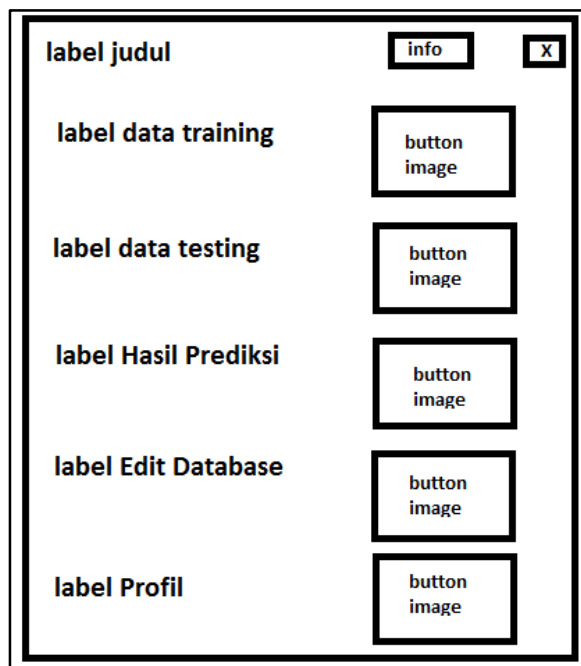
Gambar 3.14 Rancangan Antarmuka *Form Tambah Akun*

Gambar 3.14 merupakan rancangan antarmuka dari *form* tambah akun. Di dalam *form* ini terdapat komponen-komponen yang digunakan untuk melakukan proses penambahan akun, seperti terdapatnya *label*, *textbox*, dan *button save* atau *cancel*.



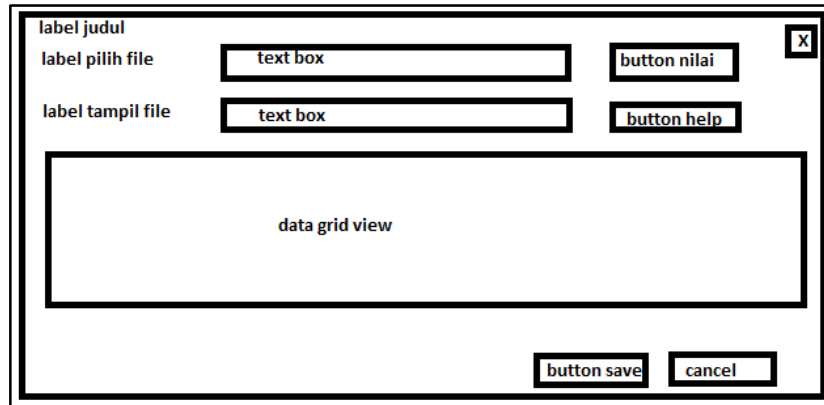
Gambar 3.15 Rancangan Antarmuka *Form* Deskripsi

Gambar 3.15 merupakan *form* rancangan antarmuka dari *form* deskripsi SPK Pemilihan Program Studi yang berisikan komponen-komponen untuk menjelaskan deskripsi mengenai sistem yang terdiri dari *button* dan label.



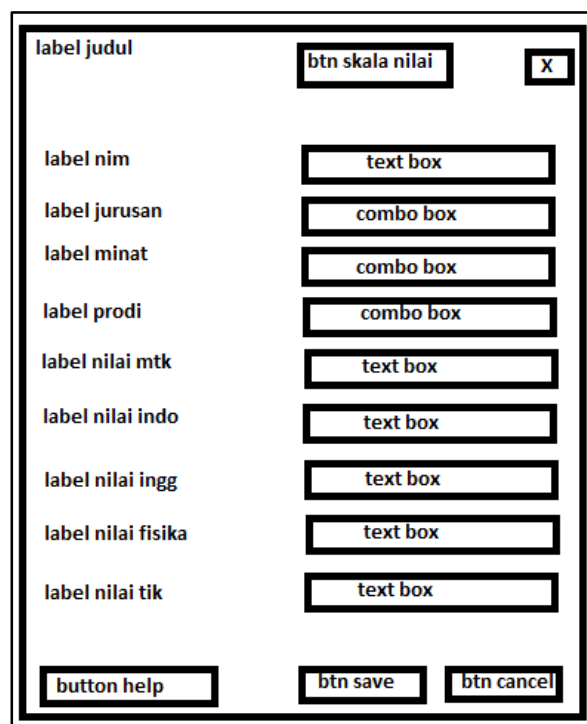
Gambar 3.16 Rancangan Antarmuka *Form* Menu Utama

Gambar 3.16 merupakan rancangan antarmuka dari *form* menu utama yang akan muncul setelah selesai melakukan proses *login*. Didalamnya terdapat *label* dan *button* untuk melanjutkan ke menu selanjutnya.



Gambar 3.17 Rancangan Antarmuka *Form Input Data Training Excel*

Gambar 3.17 merupakan rancangan antarmuka dari *form input data training excel* yang berisikan komponen-komponen untuk melakukan proses memasukkan data *training* ke dalam *database* seperti *label*, *textbox* untuk lokasi *file* yang akan dimasukkan, *data gridview* untuk menampilkan isi *file*, dan *button*.



Gambar 3.18 Rancangan Antarmuka *Form Input Data Training Manual*

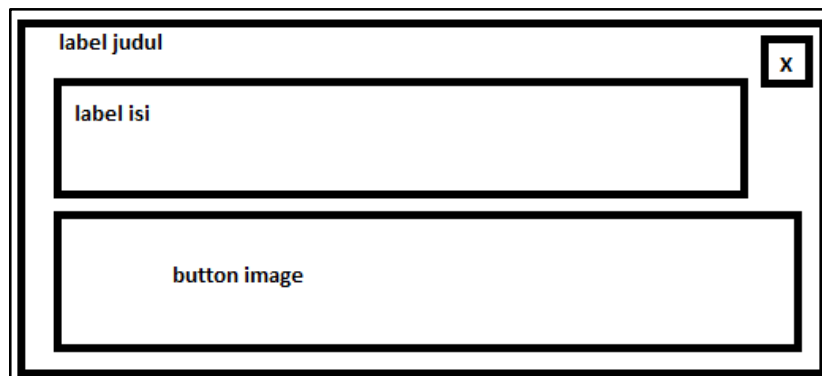
Gambar 3.18 merupakan rancangan antarmuka *form input data training* secara manual yang berisikan komponen-komponen untuk memasukkan data ke

dalam *database* yang berisikan *label*, *textbox* untuk melakukan pengisian data, dan *button* 'save', 'help', 'skala nilai', dan 'cancel'.



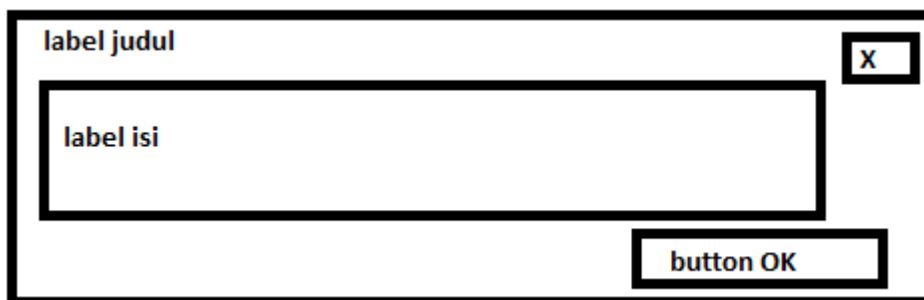
Gambar 3.19 Rancangan Antarmuka *Form Skala Nilai*

Gambar 3.19 merupakan rancangan antarmuka *form* skala nilai yang berisikan komponen-komponen seperti *label* untuk judul dan isi, serta *button* 'X' untuk keluar dari *form*.



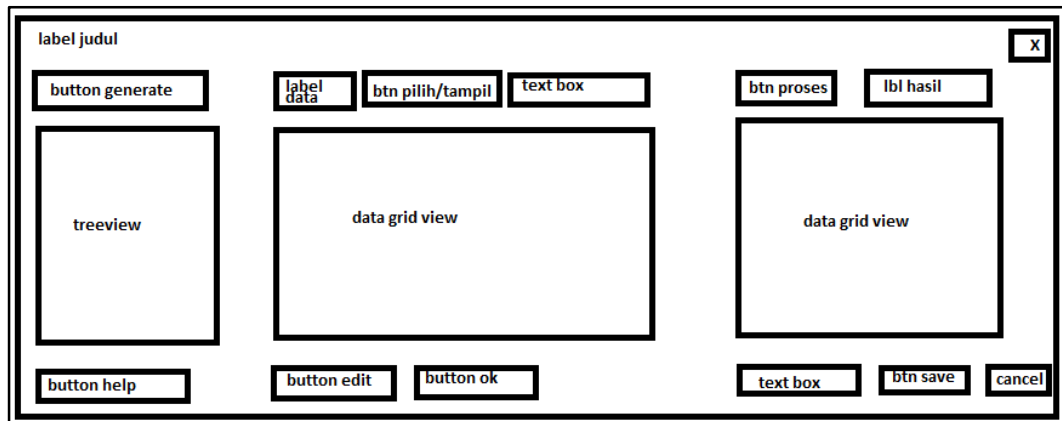
Gambar 3.20 Rancangan Antarmuka *Form Help*

Gambar 3.20 merupakan rancangan antarmuka dari *form help* yang berisikan komponen-komponen seperti *label* untuk judul dan isi, serta *button* 'X' dan *image* untuk menampilkan gambar.



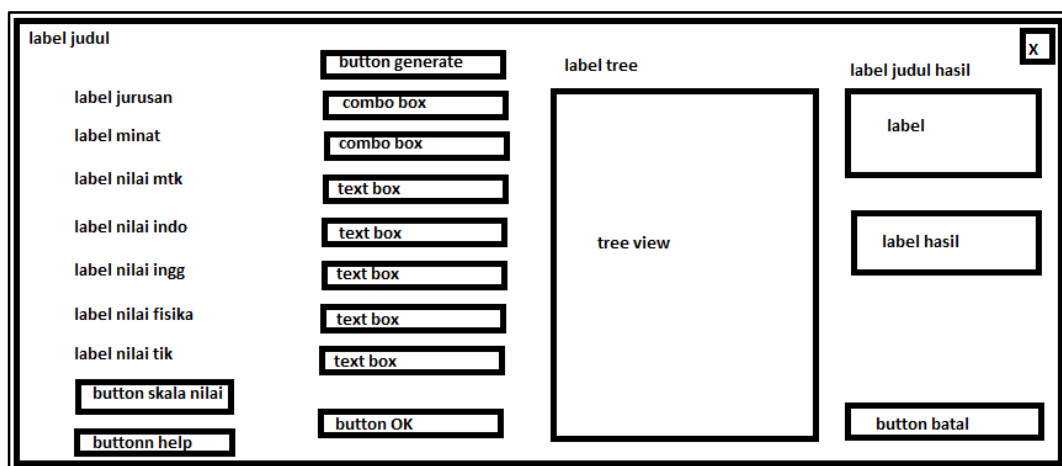
Gambar 3.21 Rancangan Antarmuka *Form Info*

Gambar 3.21 merupakan rancangan antarmuka dari *form* info yang berisikan komponen-komponen seperti *label* judul dan isi, serta *button* 'OK' dan 'X'. *Form* ini bertujuan untuk memberikan informasi-informasi terhadap sistem.



Gambar 3.22 Rancangan Antarmuka *Form Input Data Testing Excel*

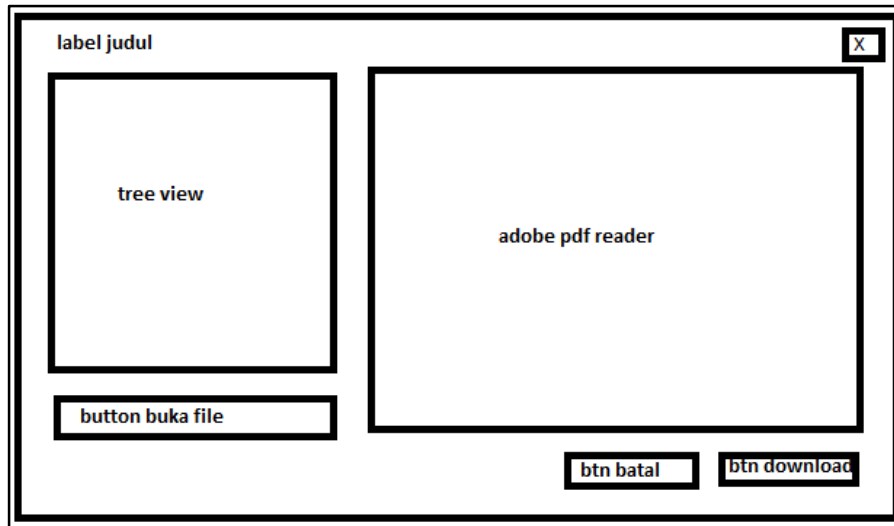
Gambar 3.22 merupakan rancangan antarmuka dari *form input data testing excel* yang berisikan komponen-komponen seperti *label*, *button*, *textbox*, *treeview*, dan *data gridview*. *Form* ini bertujuan untuk melakukan pengujian terhadap sistem dengan menggunakan data *testing* berformat *excel*.



Gambar 3.23 Rancangan Antarmuka *Form Input Data Testing Manual*

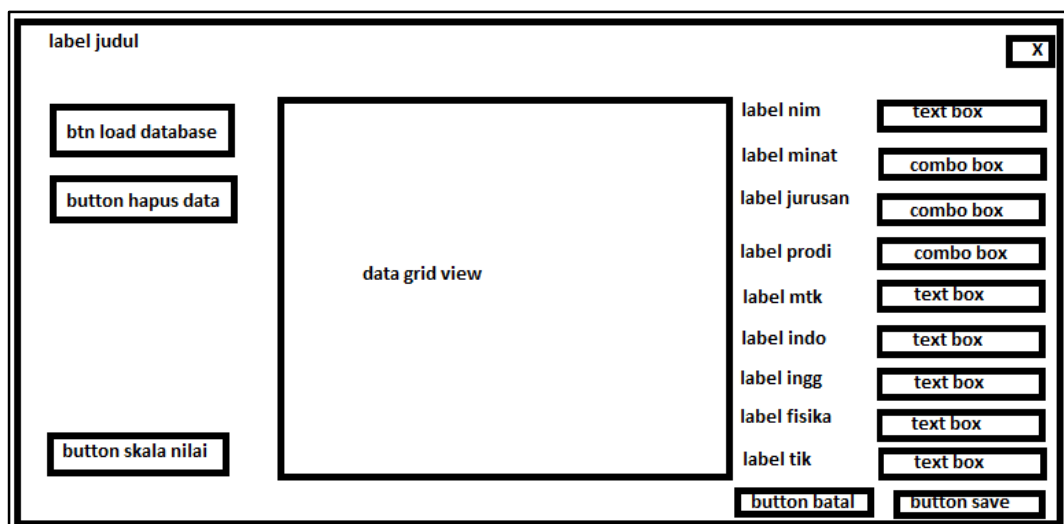
Gambar 3.23 merupakan rancangan antarmuka dari *form input data testing manual* yang berisikan komponen-komponen seperti *label*, *textbox*, *button*, dan

treeview yang bertujuan untuk melakukan pengujian terhadap sistem dengan menggunakan data *testing* secara manual.



Gambar 3.24 Rancangan Antarmuka *Form* Lihat Hasil

Gambar 3.24 merupakan rancangan antarmuka dari *form* lihat hasil yang berisikan komponen-komponen seperti *label*, *treeview*, *button*, dan *adobe pdf reader*. *Form* ini bertujuan untuk menampilkan hasil-hasil yang telah disimpan dari Gambar 3.22.



Gambar 3.25 Rancangan Antarmuka *Form* Edit Database

Gambar 3.25 merupakan rancangan antarmuka dari *form edit database* yang berisikan komponen-komponen seperti *label*, *data gridview*, *button*, dan *textbox*.

Form ini bertujuan untuk melakukan proses *edit database* dan melakukan penghapusan data terhadap data yang ada pada *database*.

3.2.4 Struktur Tabel

Rincian tabel-tabel yang terdapat pada *database* skripsi adalah sebagai berikut.

1. Nama tabel : admin

Fungsi : menyimpan identitas admin yang menggunakan sistem.

Tabel 3. 1 Struktur Tabel admin

Nama Kolom	Tipe	Keterangan
username	Varchar(30)	Username dari Admin
password	Varchar(10)	Password dari Admin

2. Nama Tabel : calon_mhs

Fungsi : menyimpan data *training* berupa data mahasiswa.

Tabel 3. 2 Struktur Tabel calon_mhs

Nama Kolom	Tipe	Keterangan
no	Varchar(20)	NIM mahasiswa
minat	Varchar(30)	Minat mahasiswa
jurusan	Varchar(30)	Jurusan mahasiswa
prodi	Varchar(30)	Prodi mahasiswa
nilai_mtk	Varchar(2)	Nilai matematika mahasiswa
nilai_indo	Varchar(2)	Nilai indonesia mahasiswa
nilai_ingg	Varchar(2)	Nilai inggris mahasiswa
nilai_fis	Varchar(2)	Nilai fisika mahasiswa
nilai_tik	Varchar(2)	Nilai TIK mahasiswa