# BAB 4
# IMPLEMENTASI DAN PENGUJIAN

Bab ini merupakan tahap menerjemahankan kebutuhan pembangunan aplikasi ke dalam representasi perangkat lunak sesuai dengan hasil analisis yang telah dilakukan. Setelah implementasi maka dilakukan pengujian sistem yang baru di mana akan dilihat kekurangan-kekurangan pada aplikasi untuk selanjutnya dievaluasi.

## 4.1 Implementasi Sistem

Setelah sistem dianalisis dan didesain secara rinci, maka akan menuju tahap implementasi. Implementasi sistem merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat memberi masukan kepada pengembangan sistem.

Setelah melakukan tahap perancangan sistem dan implementasi perangkat lunak, maka tindakan selanjutnya yang dilakukan adalah penerapan hasil perangkat lunak tersebut.

### 4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk membangun Sistem Perbandingan Metode *Adaline* dan *Backpropagation* untuk Memprediksi Jumlah Pencari Kerja di Jawa Barat seperti pada tabel 4.1.

Tabel 4.1 Perangkat Keras yang Dibutuhkan

| No | Jenis Hardware | Spesifikasi |
|---|---|---|
| 1. | Prosessor | AMD Turion (tm) X2 Dual-Core Mobile Rm-70 2.0 GHz |
| 2. | VGA Card | On-Board128 Mb |
| 3 | Memory | DDR 3 Gb |
| 4 | Optical Drive | DVD-RW |
| 5 | Mouse | Standar |

### 4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun Sistem Perbandingan Metode Adaline dan Backpropagation untuk Memprediksi Jumlah Pencari Kerja di Jawa Barat sebagai berikut :

Tabel 4.2 Perangkat Lunak yang Dibutuhkan

| Sistem Operasi | Microsoft Windows 7 Professional |
|---|---|
| Bahasa Pemrograman | C# |
| Editor Bahasa Pemrograman | *Microsoft Visual Studio 2010* |
| Database | MySQL |
| Tool Database | Xampp 1.8.1 dan SQL Yog V9.10 Beta1 |
| Library | ZedGraph 5.5.1 |
| Connector | MySQL Connector/NET 6.6.5 |

### 4.1.3 Implementasi Database

Pembuatan *database* dilakukan dengan menggunakan aplikasi *DBMS MySQL*. Implementasi *database* dalam bahasa *SQL* seperti pada tabel 4.3

Tabel 4.3 Implementasi *Database*

| | |
|---|---|
| Nama Tabel | metode |
| Deskripsi | Menyimpan nama metode JST yang digunakan |
| *Script SQL* | `CREATE TABLE IF NOT EXISTS `metode` ( `id` varchar(1) NOT NULL, `metode` varchar(15) NOT NULL, PRIMARY KEY (`id`) ) ENGINE=InnoDB DEFAULT CHARSET=latin1;` |
| Nama Tabel | data_master |
| Deskripsi | Menyimpan data master untuk pelatihan |
| *Script SQL* | `CREATE TABLE IF NOT EXISTS `data_master` ( `id` int(11) unsigned NOT NULL AUTO_INCREMENT, `tahun` int(11) unsigned NOT NULL, `jumlah_penduduk` int(11) unsigned NOT NULL, `jumlah_lulusan` int(11) unsigned NOT NULL,` |

| | |
|---|---|
| | `jumlah_angkatan_kerja` int(11) unsigned NOT NULL,`jumlah_lowongan_kerja` int(11) unsigned NOT NULL,`jumlah_pengangguran_terbuka` int(11) unsigned NOT NULL,`target` int(11) unsigned NOT NULL,PRIMARY KEY (`id`)<br>) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ; |
| Nama Tabel | bobot |
| Deskripsi | Menyimpan data bobot hasil pelatihan |
| *Script SQL* | CREATE TABLE IF NOT EXISTS `bobot` (<br>`id` int(11) unsigned NOT NULL AUTO_INCREMENT,<br>`bobot` varchar(10) NOT NULL,<br>`nilai` double NOT NULL,<br>`metode_id` varchar(1) NOT NULL,<br>PRIMARY KEY (`id`)<br>) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1; |
| Nama Tabel | prediksi |
| Deskripsi | Menyimpan data hasil prediksi |
| *Script SQL* | CREATE TABLE IF NOT EXISTS `prediksi` (<br>`id` int(11) unsigned NOT NULL AUTO_INCREMENT,<br>`tahun` int(11) unsigned NOT NULL,<br>`target` int(11) unsigned NOT NULL,<br>`prediksi` int(11) NOT NULL,<br>`metode_id` varchar(1) NOT NULL,<br>PRIMARY KEY (`id`)<br>) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ; |

Untuk mengelolah data-data pada struktur tabel tersebut, digunakan objek database *store procedure* yang merupakan konsep *database* lanjutan. Penerapan *store procedure* untuk mengurangi koding pada level aplikasi, meningkatkan *performance*, menyembunyikan *query* di level aplikasi sekaligus mengimplementasikan database tingkat lanjut. Pada dokumentasi ini, format penamaan *store procedure* adalah sp_aksi_nama_tabel sedangkan penamaan *trigger* dengan format tg_aksi_nama_tabel. *Store procedure* digunakan untuk entri, delete, update dan view di table data_master, bobot dan hasil prediksi di table prediksi. Implementasi *store procedure* diperlihatkan di tabel 4.4.

Tabel 4.4 *Store Procedure*

| | |
|---|---|
| Nama *Store Procedure* | sp_insert_data_master |
| Deskripsi | *Store procedure* untuk insert data ke tabel data master. |
| *Script SQL* | CREATE PROCEDURE sp_insert_data_master(<br>tahun INT,<br>jumlah_penduduk INT, |

| | |
|---|---|
| | jumlah_lulusan INT,<br>jumlah_angkatan_kerja INT,<br>jumlah_lowongan_kerja INT,<br>jumlah_pengangguran_terbuka INT,<br>target INT<br>)<br>BEGIN<br>INSERT INTO data_master<br>(<br>tahun INT, jumlah_penduduk INT,<br>jumlah_lulusan INT,<br>jumlah_angkatan_kerja INT,<br>jumlah_lowongan_kerja INT,<br>jumlah_pengangguran_terbuka INT,<br>target INT<br>)<br>VALUES<br>(<br>tahun,jumlah_penduduk,jumlah_lulusan,<br>jumlah_angkatanKerja,jumlah_lowonganKerja,<br>jumlah_pengangguran_terbuka,target<br>); |
| Nama *Store Procedure* | sp_update_data_master |
| Deskripsi | *Store procedure* untuk update data di tabel data master. |
| *Script SQL* | CREATE PROCEDURE sp_update_data_master(<br>id INT,<br>tahun INT,<br>jumlah_penduduk INT,<br>jumlah_lulusan INT,<br>jumlah_angkatan_kerja INT,<br>jumlah_lowongan_kerja INT,<br>jumlah_pengangguran_terbuka INT,<br>target INT<br>)<br>BEGIN<br>UPDATE data_master<br>SET tahun =tahun,<br>jumlah_penduduk = jumlah_penduduk,<br>jumlah_lulusan = jumlah_lulusan,<br>jumlah_angkatan_kerja=jumlah_angkatan_kerja,<br>jumlah_lowongan_kerja=jumlah_lowongan_kerja,<br>jumlah_pengangguran_terbuka=<br>jumlah_pengangguran_terbuka,<br>target=target<br>WHERE id=id;<br>END; |
| Nama *Store Procedure* | sp_delete_data_master |
| Deskripsi | *Store procedure* untuk delete data di tabel data master. |
| *Script SQL* | CREATE PROCEDURE<br>sp_delete_data_master(prediksi_id INT)<br>BEGIN<br>DELETE FROM data_master WHERE id=<br>prediksi_id;<br>END; |

| | |
|---|---|
| Nama *Store Procedure* | sp_view_data_master |
| Deskripsi | *Store procedure* untuk menampilkan data master |
| Script SQL | CREATE sp_view_data_master()<br>BEGIN<br>SELECT data_master.id AS 'ID#',<br>data_master.tahun AS TAHUN,<br>jumlah_penduduk AS JP, jumlah_lulusan AS<br>JL,Jumlah_angkatan_kerja AS JAK,<br>jumlah_lowongan_kerja AS JLK,<br>jumlah_pengangguran_terbuka AS JPNG,<br>target AS TARGET<br>FROM data_master;<br>ORDER BY TAHUN;<br>END; |
| Nama *Store Procedure* | sp_view_prediksi |
| Deskripsi | *Store procedure* untuk menampilkan data prediksi |
| *Script SQL* | CREATE sp_view_prediksi(id_metode IN)<br>BEGIN<br>IF id_metode=NULL OR id_metode='' THEN<br>SELECT id AS ID, tahun AS TAHUN,<br>target AS TARGET, prediksi AS<br>PREDIKSI,IF(target <>0,ROUND((100-<br>100*ABS(prediksi-target)/target),2),'NO<br>TARGET') AS 'AKURASI(%)',<br>IF(metode_id='A','ADALINE','BACKPROPAGATI<br>ON') AS 'METODE',IF((ROUND((100-<br>100*ABS(prediksi-<br>target)/target),2)>=10.00),'BAIK',IF(targ<br>et=0,'NO TARGET','JELEK')) AS KETERANGAN<br>FROM prediksi ORDER BY<br>metode_id ASC, tahun ASC;<br>ELSE<br>SELECT id AS ID, tahun AS TAHUN,<br>target AS TARGET, prediksi AS<br>PREDIKSI,IF(target <>0,ROUND((100-<br>100*ABS(prediksi-target)/target),2),'NO<br>TARGET') AS 'AKURASI(%)',<br>IF(metode_id='A','ADALINE','BACKPROPAGATI<br>ON') AS 'METODE',IF((ROUND((100-<br>100*ABS(prediksi-<br>target)/target),2)>=10.00 AND<br>target<>0),'BAIK',IF(target=0,'NO<br>TARGET','JELEK')) AS KETERANGAN<br>FROM prediksi WHERE<br>metode_id=id_metode ORDER BY metode_id<br>ASC, tahun ASC;<br>END IF;<br>END; |

### 4.1.4 Implementasi GUI

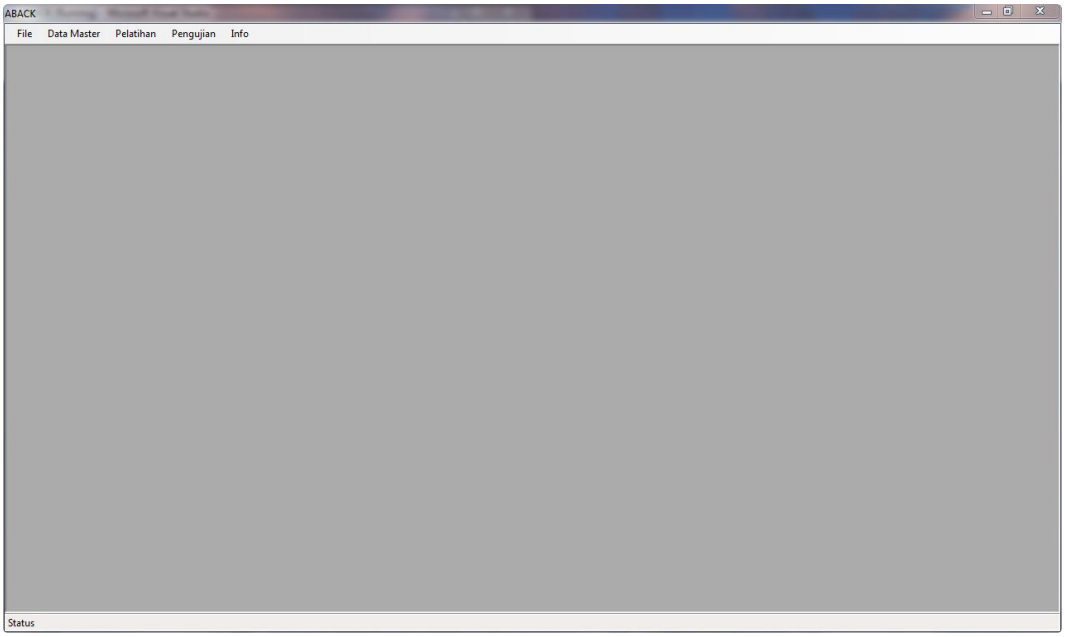
Implementasi *form* dilakukan untuk mengetahui setiap halaman program yang dibuat dan pengkodeannya dalam bentuk *file* program. Implementasi *form* yang terdapat pada perangkat lunak yang ditujukan untuk pakar dapat dilihat pada tabel 4.5

Tabel 4.5 *Implementasi GUI*

| Nama Menu | Deskripsi | Nama File |
|---|---|---|
| Halaman Utama | Tampilan awal GUI | MDIABack.cs |
| File | Keluar aplikasi | - |
| Data Master | GUI data master | DataMasterGUI.cs |
| Pelatihan | Menu yang memiliki submenu *Adaline* dan *Backpropagation* | - |
| Adaline | GUI pengujian *Adaline* yang merupakan submenu dari menu Pelatihan | AdalineGUI.cs |
| Backpropagation | GUI pengujian *Backpropagation* yang merupakan submenu dari menu Pelatihan | BackpropagationGUI.cs |
| Pengujian | GUI pengujian | PengujianGUI.cs |
| Info | GUI info | InfoGUI.cs |

a. Implementasi GUI Menu Utama

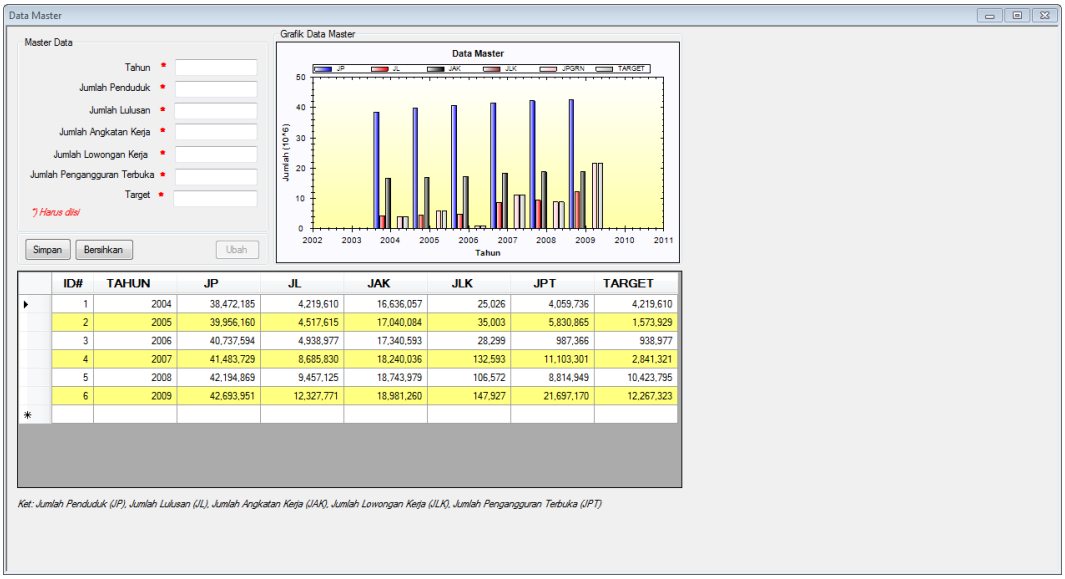
GUI ini ditampilkan ketika aplikasi pertama kali dijalankan. Pada menu utama terdapat menu File, Data Master, Pelatihan, Pengujian dan Info. Gambar 4.1 memperlihatkan implementasi GUI menu utama.




Gambar 4.1 Implementasi GUI Menu Utama

b. Implementasi GUI Data Master

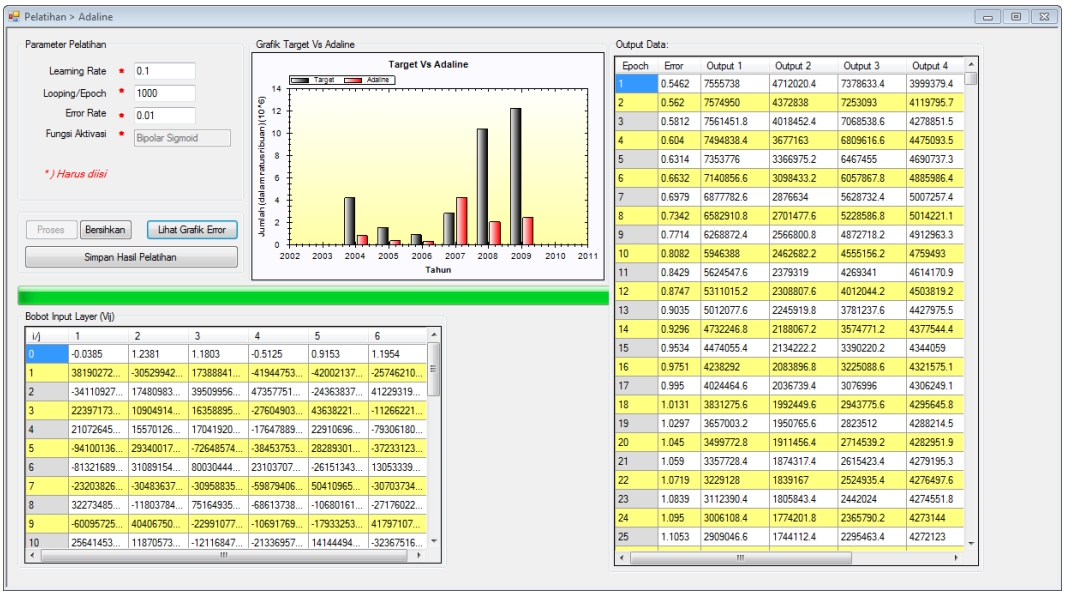
GUI untuk mengelolah data master seperti tambah, ubah, dan hapus. Gambar 4.2 memperlihatkan implementasi GUI data master.




Gambar 4.2 Implementasi GUI Data Master

c. Implementasi GUI Pelatihan *Adaline*

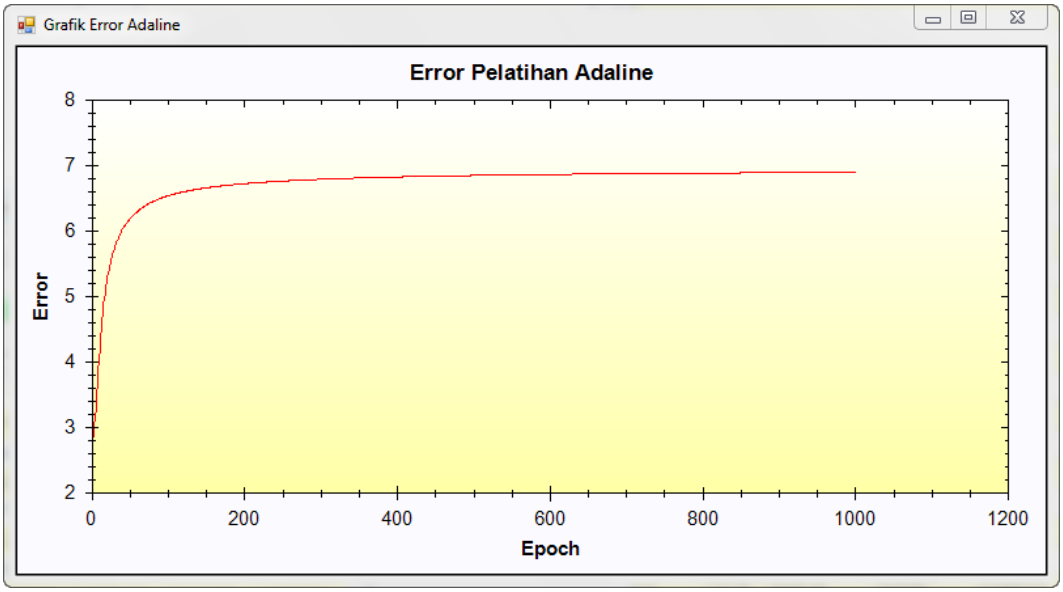
GUI ini untuk melakukan pelatihan *Adaline*. Pada GUI ini terdapat parameter pelatihan, grafik hasil pelatihan, grafik *error* pelatihan, dan data bobot. Gambar 4.3 memperlihatkan implementasi GUI pelatihan dan pengujian *Adaline*.




Gambar 4.3 Implementasi GUI Pelatihan *Adaline*

d. Implementasi GUI Grafik *Error* Pelatihan *Adaline*

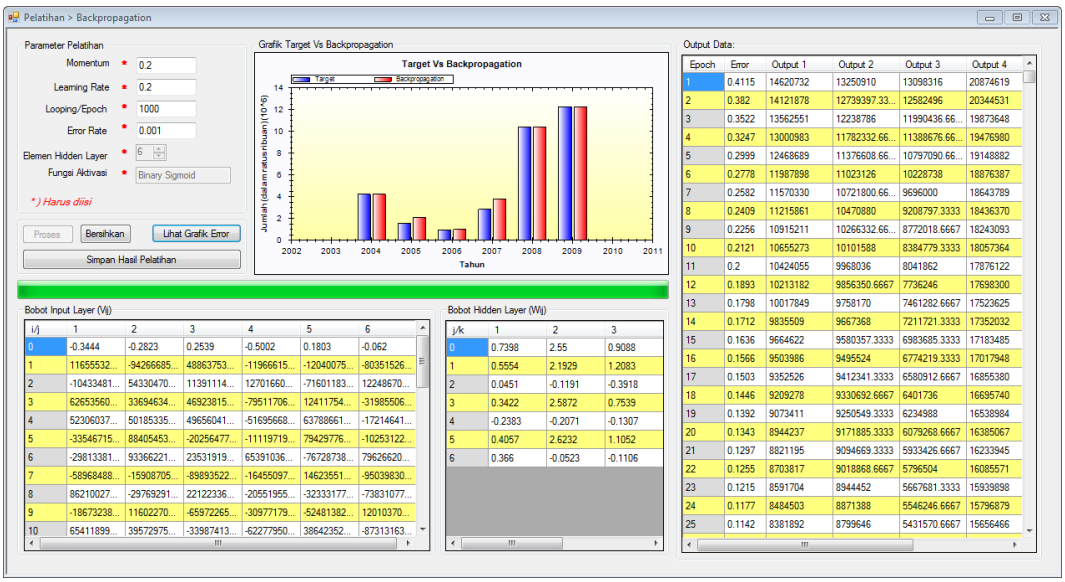
GUI ini untuk menampilkan grafik error pelatihan *Adaline*. Gambar 4.4 memperlihatkan implementasi GUI *grafik error* pelatihan *Adaline*.




Gambar 4.4 Implementasi GUI Grafik *Error* Pelatihan *Adaline*

e. Implementasi GUI Pelatihan *Backpropagation*

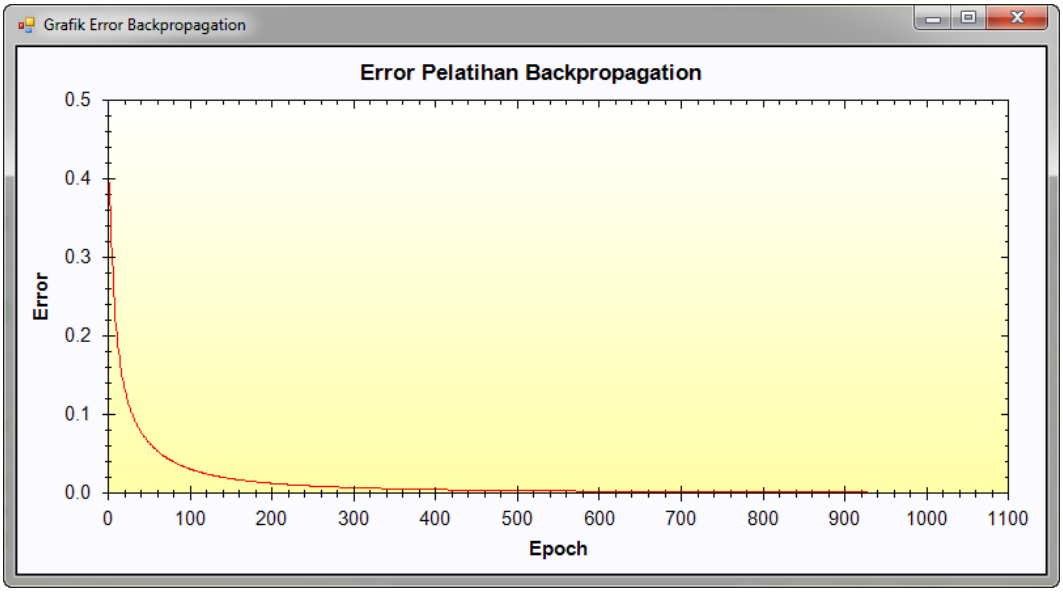
GUI ini untuk melakukan pelatihan metode *Backpropagation.* Pada GUI ini terdapat parameter pelatihan, grafik hasil pelatihan, grafik *error* pelatihan, dan data bobot. Gambar 4.5 memperlihatkan implementasi GUI pelatihan dan Pengujian *Backpropagation*.




Gambar 4.5 Implementasi GUI Pelatihan *Backpropagation*

f. Implementasi GUI Grafik *Error* Pelatihan *Backpropagation*

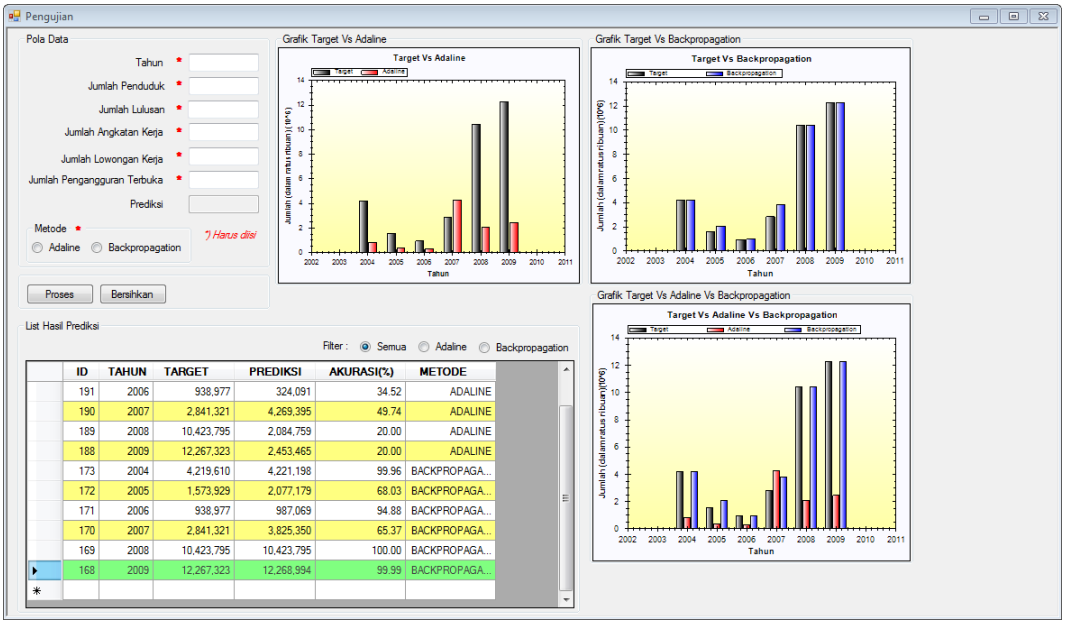
GUI ini untuk menampilkan grafik *error* pelatihan *Adaline*. Gambar 4.6 memperlihatkan implementasi GUI *grafik error* pelatihan *Adaline*.




Gambar 4.6 Implementasi GUI Grafik *Error* Pelatihan *Backpropagation*

g. Implementasi GUI Pengujian *Adaline* dan *Backpropagation*

GUI ini untuk melakukan pengujian tunggal metode *Adaline* dan *Backpropagation.* Pada GUI ini terdapat inputan pola data dan pemilihan metode pengujian, grafik *error* data target, grafik data hasil prediksi *Adaline* dan target, grafik hasil prediksi *Backpropagation* dan target, dan list data hasil prediksi dan target. Pada data list hasil prediksi dan target terdapat kolom akurasi dan dapat di *sort* berdasarkan metode. Gambar 4.7 memperlihatkan implementasi GUI pelatihan dan Pengujian *Backpropagation*.




Gambar 4.7 Implementasi GUI Pengujian *Adaline* dan *Backpropagation*

h. Implementasi GUI Info

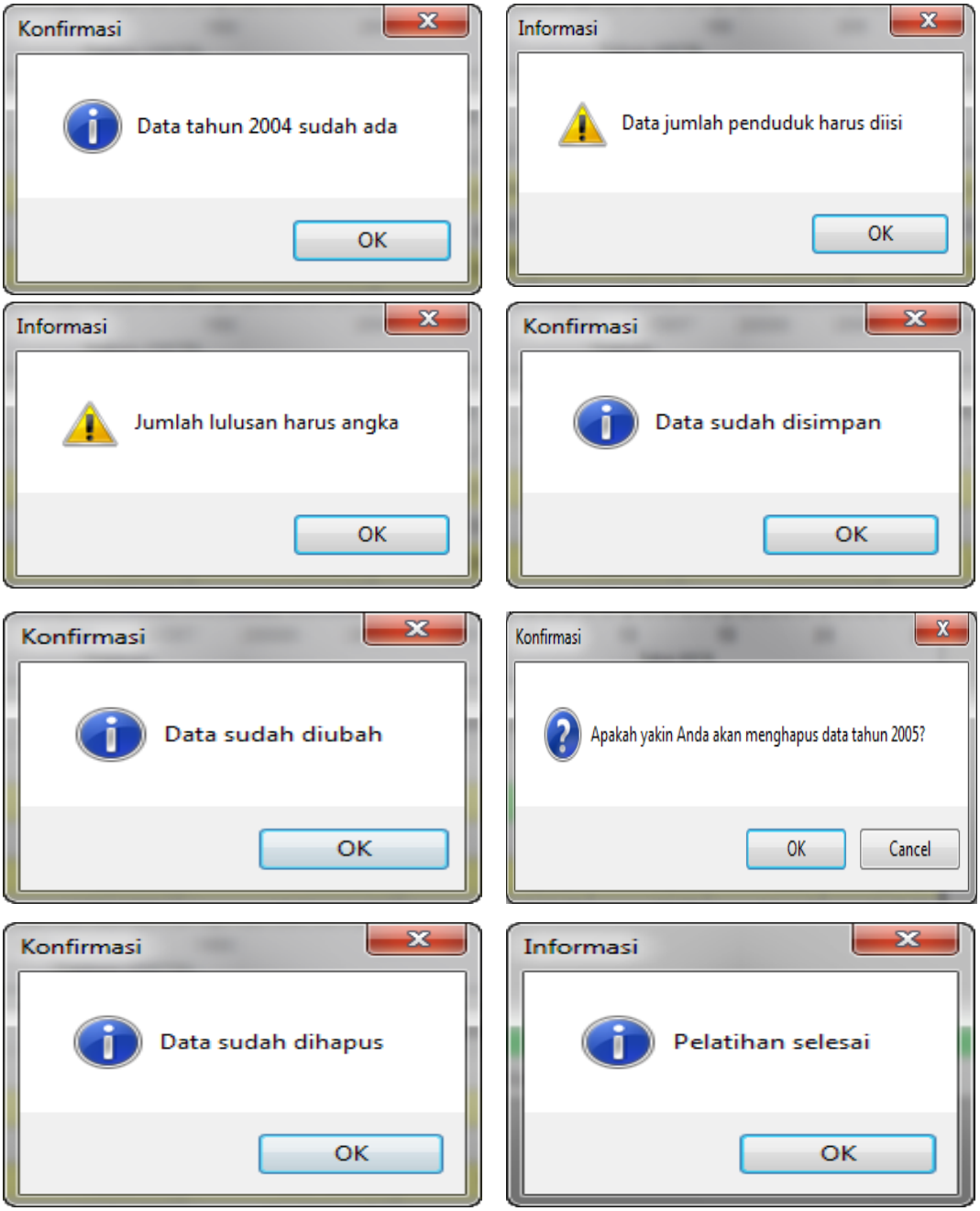
GUI ini untuk menampilkan informasi nama aplikasi, *programmer*, tahun pembuatan, versi aplikasi dan kampus. Gambar 4.8 memperlihatkan implementasi GUI info.




Gambar 4.8 Implementasi GUI Info

### 4.1.5 Implementasi GUI Pesan

Berikut ini implementasi GUI pesan aplikasi perbandingan metode *Adaline* dan *Backpropagation*:




Gambar 4.9 Implementasi GUI Pesan

## 4.2 Pengujian Sistem

Setelah perangkat lunak dibuat dengan menggunakan bahasa pemograman, perangkat lunak itu perlu diuji dan diaplikasikan untuk menunjukkan kesesuaian sistem dengan persoalan perangkat lunak tersebut dapat diuji dengan cara *black box test* dan *white box test* dan dilakukan simulasi.

.

### 4.2.1 Pengujian *Black Box*

Pengujian *black box* untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Tabel 4.6 berisi rencana pengujian *black box* yang akan dilakukan.

Tabel 4.6 Rencana Pengujian *Black Box*

| Kelas Uji | Butir Uji | Jenis Pengujian |
|---|---|---|
| Data Master | Tambah data master | *black box* |
| | Ubah data master | *black box* |
| | Hapus data master | *black box* |
| Pelatihan *Adaline* | Entri parameter pelatihan *Adaline* | *black box* |
| | Menjalankan pelatihan *Adaline* | *black box* |
| | Melihat grafik *error* pelatihan *Adaline* | *black box* |
| | Menyimpan hasil pelatihan *Adaline* | *black box* |
| Pelatihan *Backpropagation* | Entri parameter pelatihan *Backpropagation* | *black box* |
| | Menjalankan pelatihan *Backpropagation* | *black box* |
| | Melihat grafik *error* pelatihan *Backpropagation* | *black box* |
| | Menyimpan hasil pelatihan *Backpropagation* | *black box* |
| Pengujian | Masukan data pola pelatihan | *black box* |
| | Memilih metode pelatihan | *black box* |
| | Menjalankan pelatihan | *black box* |

Berdasarkan rencana pengujian, maka dapat dilakukan pengujian pada Sistem Perbandingan Metode *Adaline* dan *Backpropagation* sebagai berikut:

Tabel 4.7 Pengujian Data Master

| Tambah Data Master (Data Normal) | | | |
|---|---|---|---|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Tahun, jumlah penduduk, jumlah lulusan, jumlah angkatan kerja jumlah lowongan kerja, dan jumlah pengangguran terbuka | Data *tahun* dicari di tabel data master, jika ada tahun yang sama, data tidak akan disimpan. Jika format terpenuhi, data disimpan | Tombol **Simpan** berfungsi dengan baik, pesan jika tahun sudah terdaftar muncul, dan dapat menyimpan data | [√] Diterima<br>[ ] Tidak Diterima |

| Tambah Data Master (Data Salah) | | | |
|---|---|---|---|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Salah satu data berikut tidak diisi : Tahun, jumlah penduduk, jumlah lulusan, jumlah angkatan kerja jumlah lowongan kerja, dan jumlah pengangguran | Menampilkan pesan data harus diisi dan sistem tidak menyimpan data jika ada yang tidak lengkap | Menampilkan pesan data harus diisi sesuai dengan data yang dikosongkan dan sistem tidak menyimpan data | [√] Diterima<br>[ ] Tidak Diterima |
| Salah satu data berikut diisi bukan angka : Tahun, jumlah penduduk, jumlah lulusan, jumlah angkatan kerja jumlah lowongan kerja, dan jumlah pengangguran | Menampilkan pesan data harus angka dan sistem tidak menyimpan data jika ada yang tidak lengkap | Menampilkan pesan data harus angka sesuai dengan data yang diisi bukan dengan angka dan sistem tidak menyimpan data | [√] Diterima<br>[ ] Tidak Diterima |
| Ubah Data Master (Data Normal) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Salah satu data berikut tidak diisi : Tahun, jumlah penduduk, jumlah lulusan, jumlah angkatan kerja jumlah lowongan kerja, dan jumlah pengangguran | Data *tahun* dicari di tabel data master, jika ada tahun yang sama, data tidak akan disimpan. Jika format terpenuhi, data disimpan | Tombol **Ubah** berfungsi dengan baik, pesan jika tahun sudah terdaftar muncul, dan dapat menyimpan data | [√] Diterima<br>[ ] Tidak Diterima |
| Ubah Data Master (Data Salah) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Salah satu data berikut tidak diisi : Tahun, jumlah penduduk, jumlah lulusan, jumlah angkatan kerja jumlah lowongan kerja, dan jumlah pengangguran | Menampilkan pesan data harus diisi dan sistem tidak menyimpan data | Menampilkan pesan data harus diisi sesuai dengan data yang dikosongkan dan sistem tidak menyimpan data | [√] Diterima<br>[ ] Tidak Diterima |
| Salah satu data berikut diisi bukan angka : Tahun, jumlah penduduk, jumlah lulusan, jumlah angkatan kerja jumlah lowongan kerja, dan jumlah pengangguran | Menampilkan pesan data harus angka dan sistem tidak menyimpan data | Menampilkan pesan data harus angka sesuai dengan data yang diisi bukan dengan angka dan sistem tidak menyimpan data | [√] Diterima<br>[ ] Tidak Diterima |
| Hapus Data Master | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Klik dua kali pada data yang ingin dihapus | Menampilkan pesan konfirmasi penghapusan data. Jika menekan tombol **Yes**, data dihapus. | Menampilkan pesan konfirmasi penghapusan data. Jika menekan tombol **Yes**, data dihapus. | [√] Diterima<br>[ ] Tidak Diterima |

| | | | |
|---|---|---|---|
| | Jika menekan tombol **No**, data tidak jadi dihapus. | Jika menekan tombol **No**, data tidak jadi dihapus. | |

Tabel 4.8 Pengujian Pelatihan *Adaline*

| Pengujian *Adaline* (Data Normal) | | | |
|---|---|---|---|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| *Learning rate*, *looping/epoch*, dan *error rate* | Menjalankan pelatihan *Adaline* | Tombol **Pelatihan** berfungsi dengan baik, pelatihan *Adaline* dijalankan | [√] Diterima<br>[ ] Tidak Diterima |
| Pengujian *Adaline* (Data Salah) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Salah satu data berikut tidak diisi : *Learning rate*, *looping/epoch*, dan *error rate* | Menampilkan pesan data harus diisi dan sistem tidak menjalankan pelatihan *Adaline* | Menampilkan pesan data harus diisi sesuai dengan data yang dikosongkan dan sistem tidak menjalankan pelatihan | [√] Diterima<br>[ ] Tidak Diterima |
| Salah satu data berikut diisi bukan angka : *Learning rate*, *looping/epoch*, dan *error rate* | Menampilkan pesan data harus angka dan dan sistem tidak menjalankan pelatihan *Adaline* | Menampilkan pesan data harus angka dan dan sistem tidak menjalankan pelatihan *Adaline* | [√] Diterima<br>[ ] Tidak Diterima |
| Menampilkan Grafik *Error* Pelatihan *Adaline* | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Klik tombol **Grafik Error** | Menampilkan grafik *error* hasil pelatihan *Adaline* | Menampilkan grafik *error* hasil pelatihan *Adaline* | [√] Diterima<br>[ ] Tidak Diterima |
| Menyimpan Hasil Pelatihan *Adaline* | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Klik tombol **Simpan Hasil Pelatihan** | Menyimpan data hasil pelatihan *Adaline* dan menampilkan konfirmasi data pelatihan *Adaline* sudah disimpan | Menyimpan data hasil pelatihan *Adaline* dan menampilkan konfirmasi data pelatihan *Adaline* sudah disimpan | [√] Diterima<br>[ ] Tidak Diterima |

Tabel 4.9 Pengujian Pelatihan *Backpropagation*

| Pengujian *Backpropagation* (Data Normal) | | | |
|---|---|---|---|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| *Momentum, Learning rate*, *looping/epoch*, *error rate*, dan elemen *hidden layer* | Menjalankan pelatihan *Backpropagation* | Tombol **Pelatihan** berfungsi dengan baik, pelatihan *Backpropagation* dijalankan | [√] Diterima<br>[ ] Tidak Diterima |
| Pengujian *Backpropagation* (Data Salah) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |

| | | | |
|---|---|---|---|
| Salah satu data berikut tidak diisi : *Momentum, Learning rate, looping/epoch, error rate,* dan elemen *hidden layer* | Menampilkan pesan data harus diisi dan sistem tidak menjalankan pelatihan *Backpropagation* | Menampilkan pesan data harus diisi sesuai dengan data yang dikosongkan dan sistem tidak menjalankan pelatihan | [√] Diterima<br>[ ] Tidak Diterima |
| Salah satu data berikut diisi bukan angka : *Momentum, Learning rate, looping/epoch, error rate,* dan elemen *hidden layer* | Menampilkan pesan data harus angka dan dan sistem tidak menjalankan pelatihan *Backpropagation* | Menampilkan pesan data harus angka dan dan sistem tidak menjalankan pelatihan *Backpropagation* | [√] Diterima<br>[ ] Tidak Diterima |
| Menampilkan Grafik *Error* Pelatihan *Backpropagation* | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Klik tombol **Grafik Error** | Menampilkan grafik *error* hasil pelatihan *Backpropagation* | Menampilkan grafik *error* hasil pelatihan *Backpropagation* | [√] Diterima<br>[ ] Tidak Diterima |
| Menyimpan Hasil Pelatihan *Adaline* | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Klik tombol **Simpan Hasil Pelatihan** | Menyimpan data hasil pelatihan *Backpropagation* dan menampilkan konfirmasi data pelatihan *Backpropagation* sudah disimpan | Menyimpan data hasil pelatihan *Backpropagation* dan menampilkan konfirmasi data pelatihan *Backpropagation* sudah disimpan | [√] Diterima<br>[ ] Tidak Diterima |

Berdasarkan hasil pengujian yang telah dilakukan dapat ditarik kesimpulan bahwa aplikasi sudah berjalan sesuai yang diharapkan, tetapi tidak menutup kemungkinan dapat terjadi kesalahan suatu saat pada saat aplikasi digunakan, sehingga membutuhkan proses *maintenance* untuk lebih mengetahui kekurangan dari aplikasi.

### 4.2.2 Pengujian *White Box*

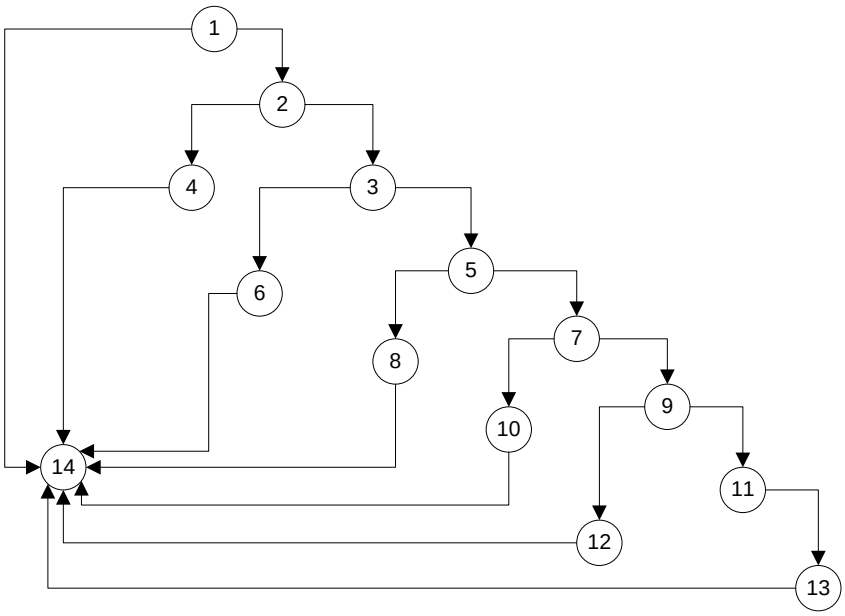
Pengujian *white box* bertujuan meyakinkan semua perintah dan kondisi dieksekusi minimal sekali. Pada dokumentasi ini hanya akan menguji validasi data dan format data harus angka pada data master. Pengujian menu lain tidak didokumentasikan karena memiliki pola yang sama dengan data master ditnjau dari validasi data. Tabel 4.10 berisi rencana pengujian *white box* yang akan dilakukan.

Tabel 4.10 Rencana Pengujian *White Box*

| Kelas Uji | Butir Uji | Jenis Pengujian |
|---|---|---|
| Data Master | Mengecek data tidak boleh null | *white box* |
| | Mengecek format data harus angka | *white box* |

Tabel 4.11 Pembuatan Node Validasi Data Tidak Boleh Kosong

```
if (tbTahun == NULL)then                                              1
  Write("Data tahun harus diisi");                                    2
else
  if (tbJumlahPenduduk==NULL)                                         3
    Write("Data jumlah penduduk harus diisi");                        4
  else
    if (tbJumlahLulusan==NULL)                                        5
       Write("Data jumlah lulusan harus diisi");                      6
    else
       if(tbJumlahAngkatanKerja==NULL)                                7
          Write("Data jumlah angkatan kerja harus diisi";             8
       else
         if(tbJumlahLowonganKerja==NULL)                              9
           Write("Data jumlah lowongan kerja harus diisi");           10
         else
           if(tbJumlahPengangguran==NULL)                             11
              Write("Data jumlah pengangguran harus diisi")           12
           else
              Write("Data jumlah target harus diisi")                 13
           Endif
         Endif
       Endif
    Endif
  Endif
Endif                                                                 14
```




Gambar 4.10 *Graph* Validasi Data Tidak Boleh Kosong

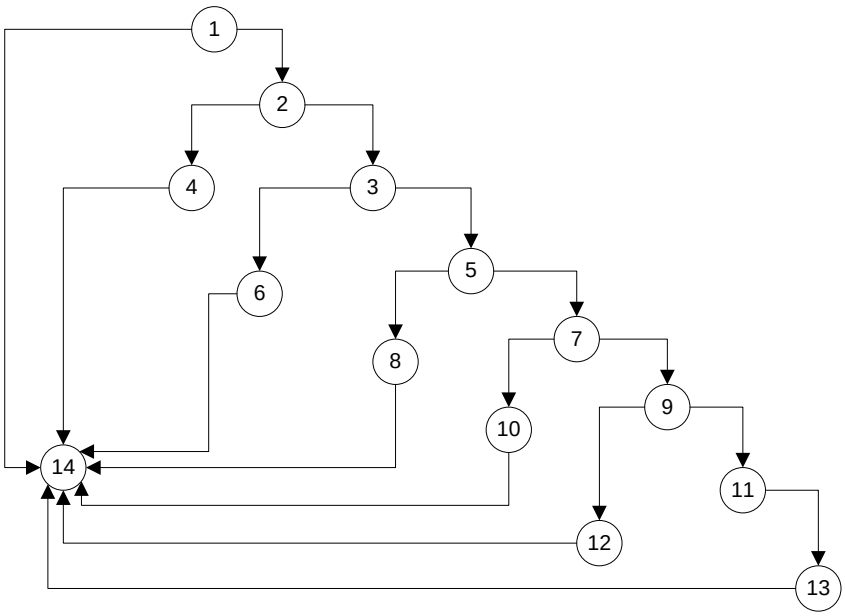
Tabel 4.12 *Path* Validasi Data Tidak Boleh Kosong

| No | Path | No | Path |
|---|---|---|---|
| 1 | 1,14 | 8 | 1,2,3,5,8,14 |
| 2 | 1,2,3,14 | 9 | 1,2,3,5,7,9,12,14 |
| 3 | 1,2,3,6,14 | 10 | 1,2,3,5,7,10,14 |
| 4 | 1,2,14 | 11 | 1,2,3,5,7,9,11,13,14 |
| 5 | 1,2,3,5,8,14 | 12 | 1,2,3,5,7,9,12,14 |
| 6 | 1,2,3,6,14 | 13 | 1,2,3,5,7,9,11,13,14 |
| 7 | 1,2,3,5,7,10,14 | | |

Tabel 4.13 Pembuatan Node Validasi Data Harus Angka

```
if (tbTahun <> angka)then                                          1
  Write("Data tahun harus angka");                                 2
else
  if (tbJumlahPenduduk <> angka)                                   3
    Write("Data jumlah penduduk harus angka");                     4
  else
    if (tbJumlahLulusan <> angka)                                  5
       Write("Data jumlah lulusan harus angka");                   6
    else
       if(tbJumlahAngkatanKerja <> angka)                          7
          Write("Data jumlah angkatan kerja harus angka";          8
       else
         if(tbJumlahLowonganKerja <> angka)                        9
```

```
            Write("Data jumlah lowongan kerja harus angka");   10
          else
            if(tbJumlahPengangguran <> angka)                  11
               Write("Data jumlah pengangguran harus angka")   12
            else
               Write("Data jumlah target harus angka")         13
            Endif
          Endif
        Endif
      Endif
    Endif
Endif                                                          14
```




Gambar 4.11 *Graph* Validasi Data Harus Angka

Tabel 4.14 *Path* Validasi Data Harus Angka

| No | Path | No | Path |
|---|---|---|---|
| 1 | 1,14 | 8 | 1,2,3,5,8,14 |
| 2 | 1,2,3,14 | 9 | 1,2,3,5,7,9,12,14 |
| 3 | 1,2,3,6,14 | 10 | 1,2,3,5,7,10,14 |
| 4 | 1,2,14 | 11 | 1,2,3,5,7,9,11,13,14 |
| 5 | 1,2,3,5,8,14 | 12 | 1,2,3,5,7,9,12,14 |
| 6 | 1,2,3,6,14 | 13 | 1,2,3,5,7,9,11,13,14 |
| 7 | 1,2,3,5,7,10,14 | | |

### 4.2.3 Pengujian Akurasi Peramalan

Bagian ini menyajikan data hasil peramalan menggunkan metode *Adaline* dan *Backpropagation* yang dibagi menjadi dua bagian yaitu pengujian akurasi terhadap data yang sudah ada, perbandingan akurasi, dan pengujian akurasi terhadap data yang belum ada dengan menggunakan learning rate 0.1, 0.2, 0.3, 0.4, dan 0.5.

a. Pengujian Akurasi Peramalan terhadap Data yang Sudah Ada

a) Hasil Prediksi *Adaline*

Tabel 4.15 Hasil Prediksi *Adaline* Tahun 2004

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 845.335 | 20,03 |
| 2 | 0.2 | 843.927 | 20,00 |
| 3 | 0.3 | 843.948 | 20,00 |
| 4 | 0.4 | 843.922 | 20,00 |
| 5 | 0.5 | 843.922 | 20.00 |

Tabel 4.16 menunjukkan hasil pengujian *Adaline* untuk data tahun 2004 dengan akurasi terbaik didapat pada *learning rate* 0.1.

Tabel 4.16 Hasil Prediksi *Adaline* Tahun 2005

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 393.357 | 24,99 |
| 2 | 0.2 | 339.924 | 21,60 |
| 3 | 0.3 | 299.123 | 20.10 |
| 4 | 0.4 | 299.123 | 20.10 |
| 5 | 0.5 | 280.792 | 19.30 |

Tabel 4.16 menunjukkan hasil pengujian *Adaline* untuk data tahun 2005 dengan akurasi terbaik didapat pada *learning rate* 0.1.

Tabel 4.17 Hasil Prediksi *Adaline* Tahun 2006

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 324.091 | 34,52 |
| 2 | 0.2 | 256.461 | 27,31 |
| 3 | 0.3 | 257.161 | 27,39 |
| 4 | 0.4 | 203.329 | 21,65 |
| 5 | 0.5 | 197.217 | 21,00 |

Tabel 4.17 menunjukkan hasil pengujian *Adaline* untuk data tahun 2006 dengan akurasi terbaik didapat pada *learning rate* 0.1.

Tabel 4.18 Hasil Prediksi *Adaline* Tahun 2007

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 4.269.395 | 49,74 |
| 2 | 0.2 | 4.269.395 | 49,74 |
| 3 | 0.3 | 4.269.395 | 49,74 |
| 4 | 0.4 | 3.536.279 | 47,57 |
| 5 | 0.5 | 3.536.279 | 47,57 |

Tabel 4.18 menunjukkan hasil pengujian *Adaline* untuk data tahun 2004 dengan akurasi terbaik didapat pada *learning rate* 0.1.

Tabel 4.19 Hasil Prediksi *Adaline* Tahun 2008

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 17.072.558 | 36.22 |
| 2 | 0.2 | 2.084.759 | 20,00 |
| 3 | 0.3 | 2.084.759 | 20,00 |
| 4 | 0.4 | 2,084,759 | 20,00 |
| 5 | 0.5 | 2.084.759 | 20,00 |

Tabel 4.18 menunjukkan hasil pengujian *Adaline* untuk data tahun 2008 dengan akurasi terbaik didapat pada *learning rate* 0.1.

Tabel 4.20 Hasil Prediksi *Adaline* Tahun 2009

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 2.453.465 | 20,00 |
| 2 | 0.2 | 2,453,465 | 20,00 |
| 3 | 0.3 | 2,453,465 | 20,00 |
| 4 | 0.4 | 2,453,465 | 20,00 |
| 5 | 0.5 | 2.453.465 | 20,00 |

Tabel 4.20 menunjukkan hasil pengujian *Adaline* untuk data tahun 2004 dengan akurasi terbaik didapat pada *learning rate* 0.1.

b) Hasil Prediksi *Backpropagation*

Tabel 4.21 Hasil Prediksi *Backpropagation* Tahun 2004

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 4.222.466 | 99,93 |
| 2 | 0.2 | 4.221.224 | 99,96 |
| 3 | 0.3 | 4.200.198 | 99,54 |
| 4 | 0.4 | 4.121.048 | 97,66 |
| 5 | 0.5 | 4.120.117 | 97,64 |

Tabel 4.21 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2004 dengan akurasi terbaik didapat pada *learning rate* 0.2.

Tabel 4.22 Hasil Prediksi *Backpropagation* Tahun 2005

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 2.076.868 | 68.05 |
| 2 | 0.2 | 2.070.079 | 68.48 |
| 3 | 0.3 | 2.077.179 | 68.03 |
| 4 | 0.4 | 2.075.733 | 68.12 |
| 5 | 0.5 | 2.075.750 | 68.12 |

Tabel 4.22 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2005 dengan akurasi terbaik didapat pada *learning rate* 0.2.

Tabel 4.23 Hasil Prediksi *Backpropagation* Tahun 2006

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 1,017,555 | 91.63 |
| 2 | 0.2 | 985,689 | 95.03 |
| 3 | 0.3 | 987,069 | 94.88 |
| 4 | 0.4 | 986,113 | 94.98 |
| 5 | 0.5 | 987,269 | 94.86 |

Tabel 4.23 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2006 dengan akurasi terbaik didapat pada *learning rate* 0.2.

Tabel 4.24 Hasil Prediksi *Backpropagation* Tahun 2007

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 3,815,669 | 65.71 |
| 2 | 0.2 | 3,825,350 | 94.88 |
| 3 | 0.3 | 3,824,785 | 65.39 |
| 4 | 0.4 | 3,823,339 | 65.44 |
| 5 | 0.5 | 3,823,289 | 65.44 |

Tabel 4.24 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2007 dengan akurasi terbaik didapat pada *learning rate* 0.2.

Tabel 4.25 Hasil Prediksi *Backpropagation* Tahun 2008

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 10,423,795 | 100.00 |
| 2 | 0.2 | 10,423,795 | 100.00 |
| 3 | 0.3 | 10,423,795 | 100.00 |
| 4 | 0.4 | 10,423,795 | 100.00 |
| 5 | 0.5 | 10,423,795 | 100.00 |

Tabel 4.25 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2008 dengan akurasi terbaik didapat pada *learning rate* 0.2.

Tabel 4.26 Hasil Prediksi *Backpropagation* Tahun 2009

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 12,268,907 | 99.99 |
| 2 | 0.2 | 12,269,004 | 99.99 |
| 3 | 0.3 | 12,268,994 | 99.99 |
| 4 | 0.4 | 12,268,961 | 99.99 |
| 5 | 0.5 | 12,268,951 | 99.99 |

Tabel 4.26 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2009 dengan akurasi terbaik didapat pada *learning rate* 0.2.

b. Perbandingan Akurasi *Adaline* dan *Backpropagation*

Berdasarkan data hasil pengujian dengan metode *Adaline* dan *Backpropagation* tersebut, rata-rata akurasi terbaik dirangkum di tabel 4.27.

Tabel 4.27 Perbandingan Akurasi *Adaline* dan *Backpropagation*

| Tahun | Adaline | Backpropagation |
|---|---|---|
| 2004 | 20.03 % | 99.96 % |
| 2005 | 24.99 % | 68.48 % |
| 2006 | 34.52 % | 95.03 % |
| 2007 | 49.74 % | 94.88 % |
| 2008 | 36.22 % | 100.00 % |
| 2009 | 20.00 % | 99.99 % |
| Rata-rata | 30.92 % | 93.06 % |

Tabel 4.27 menunjukkan bahwa rata-rata akurasi hasil prediksi dengan metode *Backpropagation* lebih tinggi daripada akurasi hasil prediksi dengan metode *Adaline*.

c. Pengujian Akurasi Peramalan terhadap Data yang Belum Ada

Data pelatihan yang digunakan adalah data tahun 2004-2009. Data pelatihan tersebut untuk menguji data tahun 2010 yang tidak diikutsertakan dalam pelatihan. Data tahun 2010 adalah data yang akan diramalkan. Dari hasil pengujian, prediksi jumlah pencari kerja tahun 2010 diperlihatkan di tabel 4.28.

Tabel 4.28 Hasil Prediksi *Adaline* Tahun 2010

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 2.200.759 | 42,52 |
| 2 | 0.2 | 2.195.126 | 42,41 |
| 3 | 0.3 | 1.974.680 | 38,15 |
| 4 | 0.4 | 2.000.785 | 38,66 |
| 5 | 0.5 | 2.041.996 | 39,45 |

Tabel 4.28 menunjukkan hasil pengujian *Adaline* untuk data tahun 2010 dengan akurasi terbaik didapat pada *learning rate* 0.1.

Tabel 4.29 Hasil Prediksi *Backpropagation* Tahun 2010

| No | Learning Rate (α) | Hasil Prediksi | Akurasi (%) |
|---|---|---|---|
| 1 | 0.1 | 4.317.741 | 83,42 |
| 2 | 0.2 | 4.473.135 | 86,43 |
| 3 | 0.3 | 4.038.696 | 78,03 |
| 4 | 0.4 | 3.798.530 | 73,39 |
| 5 | 0.5 | 3.456.033 | 66,78 |

Tabel 4.29 menunjukkan hasil pengujian *Backpropagation* untuk data tahun 2010 dengan akurasi terbaik didapat pada *learning rate* 0.2.