BAB IV

IMPLEMENTASI

Pada bab yang keempat ini akan membahas mengenai implementasi desain dari software Alternatech. Implementasi / code dibuat melalui software Microsoft Visual Studio. Software ini menggunakan .NET Framework 4.6.1 serta menggunakan WPF Web Application. Software Alternatech saat ini memiliki 7 form atau window, dan setiap role akan memiliki akses pada window-window yang berbeda. Aplikasi ini sendiri terbagi menjadi 1 halaman admin, 4 window customer, dan juga 1 window login. Software juga membutuhkan koneki dengan server MySql.

4.1 Koneksi Server

Segmen Program 4.1 berfungsi untuk membuka koneksi pada server Mysql. Koneksi ini membutuhkan data source, user ID dan juga password untuk dapat terhubung pada server. Selain itu, terdapat juga prosedur *Connect* yang berfungsi untuk membuka koneksi.

**Segment program 4.1 Connection**

1. class Connection
2. {
3. public MySqlConnection conn;
4. public void Connect()
5. {
6. conn= new MySqlConnection("server=localhost;user=root; database=sdp\_db;port=3306;password=");
7. conn.Open();
8. conn.Close();
9. }
10. }

4.2 App Page

App page ini berfungsi untuk login (berdasarkan role). Dari window App, user dapat beralih ke window admin, window homepage(user).Window ini memiliki 1 button serta 1 link yang digunakan untuk register , gambar dibawah ini digunakan Ketika user role login sebagai admin atau sebagain user dan link route untuk menuju register

**Segment program 4.2 Login di App Page**

1. if (email.Length > 0 && password.Length > 0)
2. {
3. if (email.ToLower() == "admin" && password.ToLower() == "admin")

**Lanjutan Segment program** **4.2 Login di App Page**

1. {
2. var admin = new Admin();
3. this.NavigationService.Navigate(admin);
4. }
5. else
6. {
7. conn.conn.Open();
8. cmd = new MySqlCommand();
9. cmd = new MySqlCommand($"SELECT \* FROM CUSTOMER WHERE EMAIL = '{email}'", conn.conn);
10. MySqlDataReader reader = cmd.ExecuteReader();
11. bool isFound = false;
12. while (reader.Read())
13. {
14. if (reader[3].ToString() == email)
15. {
16. isFound = true;
17. if (reader[5].ToString() == password)
18. {
19. Int id= Convert.ToInt32 (reader[0].ToString());
20. var menu = new MainMenu(id);
21. this.NavigationService.Navigate(menu);
22. break;
23. }
24. else
25. {
26. MessageBox.Show("password salah");
27. }
28. }
29. }
30. conn.conn.Close();
31. if (!isFound)
32. {
33. MessageBox.Show("User tidak ditemukan");
34. }
35. }
36. }
37. else
38. {
39. MessageBox.Show("data tidak boleh kosong");
40. }

**Segment program 4.2 Route Register di App Page**

1. private void lbregister\_PreviewMouseDown(object sender, MouseButtonEventArgs e)
2. {
3. var reg = new register();
4. this.NavigationService.Navigate(reg);
5. }

4.3 Register Page

Register page ini berfungsi untuk mendaftarkan akun user.page ini memiliki 1 button untuk register dan 1 page link untuk menuju login.

**Segment program 4.3 Register**

1. if (email.Length > 0 && username.Length > 0 && pass.Length > 0 && passconf.Length > 0 && nomor.Length > 0)
2. {
3. if(pass == passconf)
4. {
5. try
6. {
7. conn.conn.Open();
8. MySqlCommand cmd = new MySqlCommand($"INSERT INTO CUSTOMER(USERNAME, NAMA\_CUST, EMAIL,NO\_TELP,PASSWORD, SALDO) VALUES('{username}', '{name}', '{email}', '{nomor}', '{pass}', 0)", conn.conn);
10. cmd.ExecuteNonQuery();
11. conn.conn.Close();
12. }
13. catch(Exception ex)
14. {
15. MessageBox.Show(ex.Message);
16. conn.conn.Close();
17. }
18. }
19. else
20. {
21. MessageBox.Show("Konfirmasi password tidak sesuai");
22. }
23. }
24. else
25. {
26. MessageBox.Show("Data tidak boleh kosong");
27. }

**Segment program 4.3 Route Login**

1. Private void lblogin\_PreviewMouseDown(objectsender, MouseButtonEventArgs e)
2. {
3. var page1 = new Page1();
4. this.NavigationService.Navigate(page1);
5. }

4.4 Admin Page

Admin page ini berfungsi untuk CRUD barang , update request saldo user serta dapat meliah report user dan report transaksi. Admin page ini memiliki 5 button dan 1 button untuk logout, button barang digunakan untuk memunculkan data list barang, lalu button user digunakan untuk memunculkan data list user , lalu button laporan transaksi digunakan untuk memunculkan report laporan transaksi, lalu button konfirmasi top up untuk memunculkan list top up yang masih pending ,lalu button load digunakan untuk memunculkan user yang dimana terdapat combo box yang digunakan untuk memunculkan sesuai user yang dipilih Combo Box serta bisa di filter sesuai kegunaannya dengan memilih 4 button yaitu Barang, User, Laporan Transaksi, Konfirmasi Top up.

**Segment program 4.4 Button Load Barang**

1. conn.conn.Open();
2. cmd = new MySqlCommand();
3. cmd = new MySqlCommand($"SELECT BARANG.ID ,BARANG.NAMA\_BARANG AS \"NAMA BARANG\",MERK.NAMA\_MERK ,KATEGORI.NAMA\_KAT,STOK,HARGA FROM BARANG,MERK,KATEGORI WHERE BARANG.MERK = MERK.KODE\_MERK AND BARANG.KATEGORI = KATEGORI.KODE\_KAT ORDER BY id ASC", conn.conn);
4. MySqlDataAdapter sda = new MySqlDataAdapter(cmd);
5. dt = new DataTable();
6. sda.Fill(dt);
7. dgvUser.ItemsSource = dt.DefaultView;
8. conn.conn.Close();

**Segment program 4.4 Button Load User**

1. conn.conn.Open();
2. cmd = new MySqlCommand();
3. cmd = new MySqlCommand($"SELECT CUSTOMER.ID ,CUSTOMER.NAMA\_CUST AS \"NAMA CUSTOMER\",CUSTOMER.SALDO AS \"SALDO\" FROM CUSTOMER ORDER BY id ASC", conn.conn);
4. MySqlDataAdapter sda = new MySqlDataAdapter(cmd);
5. dt = new DataTable();
6. sda.Fill(dt);
7. dgvcustomer.ItemsSource = dt.DefaultView;
8. conn.conn.Close();

**Segment program 4.4 Button Load Laporan Transaksi**

1. int sub = 0;
2. conn.conn.Open();
3. cmd = new MySqlCommand();
4. cmd = new MySqlCommand($"SELECT BARANG.NAMA\_BARANG AS \"NAMA BARANG\",H\_BELI.NOMOR\_NOTA AS \"NOMOR NOTA\",D\_BELI.jumlah,D\_BELI.SUBTOTAL as \"Subtotal\" from Barang,H\_BELI,D\_BELI,Customer where Customer.id={getiduser} and H\_BELI.ID\_CUSTOMER=CUSTOMER.ID and D\_BELI.NOMOR\_NOTA=H\_BELI.NOMOR\_NOTA and BARANG.ID=D\_BELI.ID\_BARANG ORDER BY BARANG.NAMA\_BARANG", conn.conn);
6. MySqlDataAdapter sda = new MySqlDataAdapter(cmd);
7. dt = new DataTable();
8. sda.Fill(dt);
9. dgvLaporan.ItemsSource = dt.DefaultView;
10. conn.conn.Close();
11. conn.conn.Open();

**Lanjutan Segment program 4.4 Button Load Laporan Transaksi**

1. cmd = new MySqlCommand($"SELECT D\_BELI.SUBTOTAL from Barang,H\_BELI,D\_BELI,Customer where Customer.id= {getiduser} and H\_BELI.ID\_CUSTOMER=CUSTOMER.ID and D\_BELI.NOMOR\_NOTA=H\_BELI.NOMOR\_NOTA and BARANG.ID=D\_BELI.ID\_BARANG ORDER BY BARANG.NAMA\_BARANG", conn.conn);
2. MySqlDataReader reader = cmd.ExecuteReader();
3. while(reader.Read())
4. {
5. sub += Convert.ToInt32(reader[0].ToString());
6. }
7. customerLb.Content = "Name: " + (cblistcustomer.SelectedItem.ToString().Split('-'))[1];
8. totalLb.Content = "Total: " + sub.ToString();
9. conn.conn.Close();

**Segment program 4.4 Button Load Konfirmasi Top Up**

1. conn.conn.Open();
2. cmd = new MySqlCommand();
3. cmd = new MySqlCommand($"SELECT REQ\_SALDO.ID AS \"ID\",CUSTOMER.NAMA\_CUST AS \"NAMA CUSTOMER\",REQ\_SALDO.saldoreq AS \"SALDO\" FROM REQ\_SALDO,CUSTOMER WHERE CUSTOMER.ID = REQ\_SALDO.ID\_CUST AND REQ\_SALDO.KONFIRMASI = 'pending' ORDER BY REQ\_SALDO.id ASC", conn.conn);
4. MySqlDataAdapter sda = new MySqlDataAdapter(cmd);
5. dt = new DataTable();
6. sda.Fill(dt);
7. dgvkonfirsaldo.ItemsSource = dt.DefaultView;
8. conn.conn.Close();

lalu Ketika admin ingin melakukan penambahan data, admin harus memasukkan nama barang, kategori, merek, harga barang dan stock barang dan untuk mengafifkan mode ini user harus menekan tombol load barang terlebih dahulu . Ketika penambahan berhasil dimasukkan, akan muncul notifikasi berhasil. Ketika penambahan gagal, akan terdapat notifikasi juga yang muncul memberitahukan bahwa proses penambahan barang tersebut gagal.

**Segment program 4.4 Insert Barang**

1. if (text\_nmbarang.Text.Length != 0 || text\_harga.Text.Length != 0 || text\_stok.Text.Length != 0)
2. {
3. if (combo\_kategori.SelectedIndex != -1 || combo\_merk.SelectedIndex != -1)
4. {
5. getkodekat();
6. getkodemerk();
7. conn.conn.Open();

**Lanjutan Segment program 4.4 Insert Barang**

1. using (MySqlTransaction trans = conn.conn.BeginTransaction())
2. {
3. try
4. {
5. MySqlCommand cmd = new MySqlCommand($"insert into BARANG(NAMA\_BARANG, MERK, KATEGORI,STOK,HARGA) values('{text\_nmbarang.Text}','{kodemerk}','{kodekat}',{Convert.ToInt32(text\_stok.Text)},{Convert.ToInt32(text\_harga.Text)})", conn.conn);
6. cmd.ExecuteNonQuery();
7. trans.Commit();
8. conn.conn.Close();
9. MessageBox.Show("Insert berhasil");
10. }
11. catch (Exception ex)
12. {
13. trans.Rollback();
14. conn.conn.Close();
15. MessageBox.Show(ex.Message);
16. }
17. }
18. conn.conn.Close();
19. loadBarang();
20. normalmode();
21. }
22. else
23. {
24. MessageBox.Show("Pilih combobox terlebih dahulu");
25. }
26. }
27. else
28. {
29. MessageBox.Show("isi textbox terlebih dahulu");
30. }

Lalu Ketika admin ini menghapus salah satu barang harus menekan salah satu dari datagrid terlebih dahulu lalu data tersebut akan tertampil di textbox lalu admin bisa menghapus barang tersebut.

**Segment program 4.4 Delete Barang**

1. conn.conn.Open();
2. using(MySqlTransaction trans = conn.conn.BeginTransaction())
3. {
4. try
5. {
6. MySqlCommand cmd = new MySqlCommand($"delete from barang where ID = {index+1}", conn.conn);
7. cmd.ExecuteNonQuery();
8. trans.Commit();
9. conn.conn.Close();

**Lanjutan Segment program 4.4 Delete Barang**

1. MessageBox.Show("delete berhasil");
2. }
3. catch (Exception ex)
4. {
5. trans.Rollback();
6. conn.conn.Close();
7. MessageBox.Show(ex.Message);
8. }
9. }
10. conn.conn.Close();
11. loadBarang();
12. normalmode();

lalu ketika admin ingin memperbarui isi data barang sama dengan cara untuk menghapus barang yang terdapat pada list , hanya menekan barang yang dipilih serta lalu isi 5 combo box yang mau diubah yang terdiri dari nama barang, stock, harga , kategori, merek.

**Segment program 4.4 Update Barang**

1. if (text\_nmbarang.Text.Length != 0 || text\_harga.Text.Length != 0 || text\_stok.Text.Length != 0)
2. {
3. if (combo\_kategori.SelectedIndex != -1 || combo\_merk.SelectedIndex != -1)
4. {
5. getkodekat();
6. getkodemerk();
7. conn.conn.Open();
8. using (MySqlTransaction trans = conn.conn.BeginTransaction())
9. {
10. try
11. {
12. MySqlCommand cmd =new MySqlCommand($"Update BARANG SET NAMA\_BARANG = '{ text\_nmbarang.Text }',MERK = '{ kodemerk }',kategori='{kodekat}',STOK={Convert.ToInt32 (text\_stok.Text)},HARGA = {Convert.ToInt32(text\_harga.Text)} Where id = " + (index+1) ,conn.conn);
13. cmd.ExecuteNonQuery();
14. trans.Commit();
15. conn.conn.Close();
16. MessageBox.Show("Update berhasil");
17. }
18. catch (Exception ex)
19. {
20. trans.Rollback();
21. conn.conn.Close();
22. MessageBox.Show(ex.Message);
23. }
24. }
25. conn.conn.Close();
26. loadBarang();
27. normalmode();
28. }
29. else
30. {

**Lanjutan Segment program 4.4 Delete Barang**

1. MessageBox.Show("Pilih combobox terlebih dahulu");
2. }
3. }
4. else
5. {
6. MessageBox.Show("isi textbox terlebih dahulu");
7. }

Lalu pada fitur konfirmasi saldo, admin dapat menyetujui request saldo atau menolak request saldo user, terlebidahulu admin harus menekan salah satu user yang terdapat pada datagridview, lalu terdapat 2 tombol terima dan tolak.saat admin menekan tombol terima maka saldo akan langsung masuk ke saldo user yang sudah di konfirmasi dan saat admin menekan tombol tolak makan request tersebut dibatalkan.

**Segment program 4.4 menyetejui request saldo**

1. if (tbnamacust.Text.Length != 0 || tbsaldo.Text.Length != 0)
2. {
3. conn.conn.Open();
4. using (MySqlTransaction trans = conn.conn.BeginTransaction())
5. {
6. try
7. {
8. MySqlCommand cmd = new MySqlCommand($"Update CUSTOMER SET SALDO = {Convert.ToInt32(tbsaldo.Text)} WHERE NAMA\_CUST = '{tbnamacust.Text}'" , conn.conn);
9. MySqlCommand cmd2 = new MySqlCommand($"Update REQ\_SALDO SET KONFIRMASI = 'accepted' WHERE ID = {dt.Rows[dgvkonfirsaldo.SelectedIndex][0]}", conn.conn);
10. cmd.ExecuteNonQuery();
11. cmd2.ExecuteNonQuery();
12. trans.Commit();
13. conn.conn.Close();
14. MessageBox.Show("Top Up Berhasil Diterima");
15. LoadSaldo();
16. }
17. catch (Exception ex)
18. {
19. trans.Rollback();
20. conn.conn.Close();
21. MessageBox.Show(ex.Message);
22. }
23. }
24. }
25. else
26. {
27. MessageBox.Show("pilihan tidak boleh kosong");
28. }

**Segment program 4.4 menolak request saldo**

1. if (tbnamacust.Text.Length != 0 || tbsaldo.Text.Length != 0)
2. {
3. conn.conn.Open();
4. using (MySqlTransaction trans = conn.conn.BeginTransaction())
5. {
6. try
7. {
8. MySqlCommand cmd = new MySqlCommand($"Update REQ\_SALDO SET KONFIRMASI = 'rejected' WHERE ID = {dt.Rows[dgvkonfirsaldo.SelectedIndex][0]}", conn.conn);
9. cmd.ExecuteNonQuery();
10. trans.Commit();
11. conn.conn.Close();
12. MessageBox.Show("Top Up Berhasil Ditolak");
13. LoadSaldo();
14. }
15. catch (Exception ex)
16. {
17. trans.Rollback();
18. conn.conn.Close();
19. MessageBox.Show(ex.Message);
20. }
21. tbnamacust.Clear();
22. tbsaldo.Clear();
23. }
24. }
25. else
26. {
27. MessageBox.Show("pilihan tidak boleh kosong");
28. }