BAB IV

IMPLEMENTASI

Pada bab ini akan dijelaskan mengenai proses implementasi program. Implementasi program didapat melalui desain sistem program yang dibuat. Beberapa hal yang akan dibahas antara lain proses login user, proses insert data, proses update data, proses delete data, proses search data, dan proses pembuatan laporan.

4.1 Proses Login

Pada bagian ini akan dijelaskan mengenai proses login user pada Sistem Manajemen Online Shop “Maju Jaya”. Untuk melaukan login, user memerlukan email dan password. Email diperoleh dari email yang sudah didaftarkan. Begitu juga dengan password, hanya saja password akan dienkripsi atau diubah menjadi suatu kode tertentu pada database sehingga password user benar-benar sangat aman. Potongan program untuk proses login dapat dilihat pada segmen program 5.1 dibawah ini.

Segmen Program 4.1 Login

1. public function login()
2. {
3. if($this->input->post("login"))
4. {
5. $email = $this->input->post('email');
6. $pass = $this->input->post('password');
7. $temp = $this->Model->selectAllUser();
8. $tampung = "";
9. $cek = false;
10. if($email == "administrator" && $pass == "administrator")
11. {
12. redirect(site\_url("Admin/barang"));
13. }
14. else
15. {
16. foreach ($temp as $t)
17. {
18. if($email == $t->Email)

**Segmen Program 4.1 Login (lanjutan)**

1. {
2. if(password\_verify($pass,$t->Password)
3. {
4. if($this->Model->isUserVerified($email)
5. {
6. $cek = true;
7. $tampung = $t->Nama\_user;
8. }
9. }
10. }
11. }
12. if($cek == true)
13. {
14. echo "<script>alert('Sukses Login!')</script>";
15. $data['nama'] = $tampung;
16. $this->session->set\_userdata('login',$data['nama']);
17. redirect('Cont/index');
18. }
19. else
20. {
21. echo "<script>alert('Gagal Login!')</script>";
22. $this->load->view('login');
23. }
24. }
25. }
26. else
27. {
28. $this->load->view("login");
29. }
30. }

Baris ke-5 dan baris ke-6 digunakan untuk mengambil email dan password yang diinputkan oleh user. Baris ke-10 sampai baris ke-13 digunakan untuk mengecek email dan password yang dimasukkan pada saat login. Jika email dan password yang diisikan adalah administrator maka akan dimasukan ke dalam halaman admin. Baris ke-16 sampai ke-29 digunakan untuk mencocokkan email dan password dari user apakah benar dan ada di database dan sudah terverifikasi. Apabila email user sudah terverifikasi maka variabel cek akan diisikan dengan nilai true.. Baris ke-30 sampai ke-41 akan berguna membawa user ke halaman index jika berhasil login(variabel cek beriisikan nilai true) dan akan mengembalikan user ke halaman login jika gagal melakukan login.

4.2 Insert Barang

Bagian ini akan menjelaskan mengenai insert data. Proses insert ini merupakan salah satu fungsi yang berperan besar dalam Sistem Online shop ini. Berikut adalah contoh proses insert barang pada segmen program 4.2 dibawah ini.

Segmen Program 4.2 Tambah Barang

1. if($this->input->post('insert')) {
2. $post = $this->input->post();
3. $tempid='';
4. $id='';
5. if(isset($post["insert"])) {
6. $namabarang = $this->input->post("namabrg");
7. $stok = $this->input->post("stok");
8. $harga = $this->input->post("harga");
9. $deskripsi = $this->input->post("deskripsi");
10. $kategori = $this->input->post("kategori");
11. $merk = substr($namabarang,0,strpos($namabarang," "));
12. $tempmerk = $this->Model->getmerk($merk);
13. $idmerk='';
14. foreach($tempmerk as $row) {
15. $idmerk = $row->id\_merk;
16. }
17. $tempid = $this->Model->getLastIdBarang();
18. foreach($tempid as $row) {
19. $id = $row->id;
20. }
21. $id = $id+1;
22. $kode = 'B000'.$id;
23. $config['upload\_path'] = './resource/';
24. $config['allowed\_types'] = 'png';
25. $config['overwrite'] = TRUE;
26. $config['file\_name'] = $namabarang;
27. $this->load->library('upload', $config);
28. if ( ! $this->upload->do\_upload("foto"))
29. {
30. $data["error"] = $this->upload
31. ->display\_errors();
32. }
33. else
34. {
35. $info = $this->upload->data();
36. $foto = $info["file\_name"];
37. if($this->Model->insertbarang($kode,$namabarang,$stok,
38. $harga,$deskripsi,$idmerk,$kategori,$foto)>0) {
39. echo"<script>alert('Berhasil
40. Upload');</script>";
41. site\_url('Admin/barang');
42. }
43. }
44. }

Baris ke-6 sampai baris ke-12 digunakan untuk menampung data barang yang dimasukkan oleh user. Baris ke-21 dan baris ke-22 digunakan untuk menghasillan id barang. Baris ke-22 sampai baris ke-27 digunakan untuk mengkonfigurasi file upload gambar. Baris ke-28 sampai ke-41 berfungsi untuk melakukan pengecekan apakah file upload telah berhasil dilakukan. Jika tidak maka akan keluar tulisan error dan jika berhasil akan keluar tulisan “Berhasil Upload”.

4.3 Update Barang

Bagian ini akan menjelaskan mengenai update data atau memperbarui data. Proses update ini merupakan salah satu fungsi yang berperan besar dalam Sistem Online Shop ini. Proses update pada Sistem Online Shop terletak pada halaman yang sama dengan proses insert dan diberi nama edit. Sebagai contoh akan ditunjukkan proses update atau edit barang tanpa melakukan perubahan pada gambar barang pada segmen program 4.3 dibawah ini.

Segmen Program 4.3 Update Barang

1. if($\_FILES['foto']['size'] == 0) {
2. $id = $this->input->post("id");
3. $namabarang = $this->input->post("namabrg");
4. $stok = $this->input->post("stok");
5. $harga = $this->input->post("harga");
6. $deskripsi = $this->input->post("deskripsi");
7. $kategori = $this->input->post("kategori");
8. $merk = substr($namabarang,0,strpos($namabarang," "));
9. $tempmerk = $this->Model->getmerk($merk);
10. $idmerk='';
11. foreach($tempmerk as $row) {
12. $idmerk = $row->id\_merk;
13. }
14. $this->Model->updatetanpafoto($id,$namabarang,
15. $stok,$harga,$deskripsi,$kategori,$idmerk);
16. redirect('Admin/barang');
17. }
18. public function updatetanpafoto($id,$namabarang,
19. $stok,$harga,$deskripsi,$kategori,$merk)
20. {
21. $data = array(
22. 'barang\_nama' => $namabarang,
23. 'barang\_stok' => $stok,
24. 'harga\_satuan' => $harga,
25. 'deskripsi' => $deskripsi,
26. 'id\_merk' => $merk,
27. 'id\_kategori' => $id\_kategori
28. );
29. $this->db->where('barang\_id',$id);
30. $this->db->update('barang',$data);
31. }

Baris ke-2 sampai baris ke-9 digunakan untuk mengambil data yang dimasukkan oleh user dan menyimpannya kedalam variabel. Baris ke-21 sampai ke-30 digunakan untuk melakukan proses update barang pada database sesuai dari input user. Baris ke-16 digunakan untuk mengembalikan user ke halaman admin barang.

4.4 Delete Barang

Pada bagian ini akan dijelaskan mengenai delete data atau menghapus data. Proses delete ini merupakan salah satu fungsi yang berperan besar dalam Sistem Online Shop ini. Proses delete pada Sistem Online Shop terletak pada halaman master barang. Sebagai contoh akan ditunjukkan proses delete barang pada segmen program 4.4 dibawah ini.

Segmen Program 4.4 Delete Barang

1. public function getdelete()
2. {
3. $id = $this->input->post("id");
4. $this->session->set\_userdata("id",$id);
5. redirect('Admin/deletebarang');
6. }
7. public function deletebarang() {
8. $id = $this->session->userdata("id");
9. $this->Model->deletebarang($id);
10. $this->session->unset\_userdata('id');
11. redirect('Admin/barang');
12. }
13. public function deletebarang($id)
14. {
15. $this->db->where('barang\_id',$id);
16. $this->db->delete('barang');
17. }

Baris pertama sampai baris keenam digunakan untuk menyimpan id barang yang akan dihapus oleh user. Baris ke-8 digunakan untuk menampung id barang yang disimpan sebelumnya tadi. Baris ke-15 dan ke-16 digunakan untuk menghapus data barang sesuai dengan yang memiliki id barang sesuai dari id yang ditampung dari database. Jika berhasil user akan dikembalikan ke halaman barang.

4.5 Search Barang

Pada bagian ini akan dijelaskan mengenai search data atau mencari data barang. Proses search ini terdapat pada halaman master barang. Proses search dilakukan dengan menginputkan nama barang yang hendak dicari. Sebagai contoh akan ditunjukkan search data barang menggunakan alamat pada segmen program 4.5 dibawah ini.

Segmen Program 4.5 Search Barang Dengan Nama Barang

1. public function search()
2. {
3. $index = $this->input->post('jumlah');
4. $this->session->set\_userdata('jumlah',$index);
5. $keyword = $this->input->post('keyword');
6. $this->session->set\_userdata('cari',$keyword);
7. redirect('Admin/hasilsearch');
8. }
9. public function hasilsearch()
10. {
11. $cari = $this->session->userdata('cari');
12. if($this->session->userdata('jumlah') == false){
13. $jumlah = 5;
14. }
15. else{
16. $jumlah = $this->session->userdata('jumlah');
17. }
18. $config = array();
19. $config['base\_url'] = base\_url()."index.php/Admin/hasilsearch";
20. $config['total\_rows'] = $this->Model->jumlahsearch($cari);
21. $config['per\_page'] = $jumlah;
22. $config['uri\_segment'] = 3;
23. $config['first\_link'] = '<|';
24. $config['last\_link'] = '|>';
25. $config['next\_link'] = '>';
26. $config['prev\_link'] = '<';
27. $config['num\_links'] = 1;
28. $this->pagination->initialize($config);
29. if($this->uri->segment(3) > 1){
30. $data[‘data’] = $this->Model->search($cari,$config[‘per\_page’],$this-
31. >uri->segment(3));
32. }
33. else if($this->uri->segment(3) < 1){
34. $data[‘data’] = $this->Model->search($cari,$config[‘per\_page’],0);
35. }
36. $data['barang'] = $this->Model->selectbarang();
37. $data['links'] = $this->pagination->create\_links();
38. $data['halaman'] = $this->pagination->cur\_page;
39. $this->session->set\_userdata('def',$jumlah);
40. $this->load->view('masterbarang',$data);
41. }

Baris ke-3 sampai baris ke-6 terdapat merupakan bagian program yang mengambil input dari user berupa jumlah dan nama barang yang hendak dicari. Baris ke-16 samapi baris ke-28 berisikan konfigurasi dari pagination yang hendak ditampilkan. Baris ke-30 dan ke-34 berisikan bagian program yang mencari barang yang hendak dicari oleh user. Baris ke-36 sampai ke-38 merupakan bagian menampung hasil pencarian dan konfigurasi pagination yang hendak ditampilkan.

4.6 Laporan

Pada bagian ini akan dijelaskan mengenai proses pembuatan laporan dari data yang ada pada Sistem Online shop ini. Proses pembuatan laporan ini menggunakan library Chart.js untuk menampilkan laporan berupa grafik. Laporan yang ada pada Sistem Online Shop ini adalah laporan pemasuka. Sebagai contoh akan ditunjukkan pembuatan laporan pemasukan bulanan. Pada segmen program 4.6 dibawah ini akan ditunjukkan cara menampilkan laporan.

Segmen Program 4.6 Cara Menampilkan Laporan

1. <script src="https://cdnjs.cloudflare.com/ajax/libs
2. /Chart.js/2.1.4/Chart.min.js"></script>
3. public function halamanreport()
4. {
5. $data['report'] = $this->Model->
6. getReportPemasukanBulanan();
7. $this->load->view('report',$data);
8. }
9. public function getReportPemasukanBulanan()
10. {
11. $query = $this->db->query("SELECT DISTINCT
12. substring(DATE\_FORMAT(Tanggal,'%Y-%m-%d'),6,2)
13. as bulan,SUM(TOTAL) AS total FROM HTRANS
14. WHERE YEAR(CURRENT\_DATE) = 2018 GROUP BY substring
15. (DATE\_FORMAT(Tanggal,'%Y-%m-%d'),6,2)");
16. return $query->result();
17. }
18. var myChart = new Chart(ctx, {
19. type: 'line',
20. data: {
21. labels: [
22. <?php
23. foreach($report as $row)
24. {
25. echo "'".$row->bulan."',";
26. }
27. ?>
28. ],
29. datasets: [{
30. label: "Pemasukan",
31. data: [
32. <?php
33. foreach($report as $row)
34. {
35. echo $row->total.",";
36. }
37. ?>
38. ],
39. backgroundColor: "rgba(153,255,51,0.4)"
40. }]
41. }
42. });

Baris pertama dan kedua digunakan untuk memanggil library Chart.js. Baris ke-5 sampai baris ke-7 digunakan untuk mengambil laporan dari database dan memasukan data laporan ke halaman report. Baris ke-11 sampai baris ke-16 berisi program secara detail bagaimana pengambilan data laporan dari database. Baris ke-18 sampai baris ke-42 digunakan untuk menampilkan laporan pada halaman report berdasarkan dari data yang diambil dari database.