

DAFTAR PUSTAKA

- Dhika, H., & Fitriansyah, A. (2016). Perancangan Sistem Informasi Jasa Pengiriman Barang Berbasis Web. *Jurnal SIMETRIS*, 7(1).
- Febrianti, N., D. (2021). Implementasi *Black Box Testing* Pada Sistem Informasi Manajemen Dosen. *Jurnal Teknologi Dan Komputer*. (Vol. 2, No. 3).
- Ferdika, M., Kuswara, H. (2017). Sistem Informasi Penjualan Berbasis Web Pada PT Era Makmur Cahaya Damai Bekasi. *Information System For Educators And Professionals*, 1(2): 175.
- Firman, A., Wowor, H. F., & Najoran, X. (2016). Sistem Informasi Perpustakaan Online Berbasis Web. *Jurnal Teknik Elektro Dan Komputer*, 5(2).
- Hidayat, dkk. (2017). Perancangan Sistem Informasi Penjualan Barang Handmade Berbasis Website Dengan Metode *Waterfall*. *Simposium Nasional Ilmu Pengetahuan dan Teknologi (SIMNASIPTEK)*, 175-183.
- Junaedi, D., & Salistia, F. (2020). Dampak Pandemi *COVID-19* Terhadap Pertumbuhan Ekonomi Negara-Negara Terdampak. *SIMPOSIUM Nasional Keuangan Negara*, 995-1012.
- Laksono, H., Hariyanti, S., & Romzi, M. (2018). Membangun Website Rumah Sakit Pratama Lubai Ulu Kabupaten Muara Enim Menggunakan Php & Mysql. *Jtim: Jurnal Teknik Informatika Mahakarya*, 1(1).
- Laugi, Saidah. (2018). Sistem Informasi berbasis Web dalam Penyelenggaraan Lembaga Pendidikan. *Shautut Tarbiyah*, 109-126.
- Manurung, I. H. G. (2019). Sistem Informasi Lembaga Kursus Dan Pelatihan (Lkp) City Com Berbasis Web Menggunakan Php Dan Mysql. *Jurnal Mahajana Informasi*, 4(1).
- Praba, Ardian. D. (2018). Implementasi Model View Controller Dengan framework CodeIgniter Pada Perpustakaa. *Indonesia Journal on Software Engineering*. (Vol. 4, No. 1).

- Pujantoko, Yoga. (2009). *Pembuatan Website SMA Negeri 1 Pracimantoro Menggunakan PHP & MySQL*. Universitas Sebelas Maret
- Rais, M. (2019). Penerapan Konsep Object Oriented Programming Untuk Aplikasi Pembuat Surat. *Jurnal PROtek*, 6(2).
- Sahi, Ahmad. (2020). Aplikasi Test Potensi Akademin Seleksi Saringan Musik LP31 Berbasis Web Online Menggunakan Framework Codeigniter. *Jurnal Teknologi Informasi Dan Komunikasi*, 7(1).
- Setiawan Wawan. (2017). Era Digital dan Tantangannya. *Jurnal Repository UMMI*.
- Sitohang, dkk. (2018). Sistem Informasi Pengagendaan Surat Berbasis Web Pada Pengadilan Tinggi Medan. *Jurnal Informatic Pelita Nusantara*, 3(1).
- Sijintak, dkk. (2020). Analisis Perancangan Sistem Informasi Administrasi Kursus Bahasa Inggris Pada Intensive English Course Di Ciledug Tangerang. *Jurnal IPSIKOM*, 8(1).
- Sucipto. (2017). Perancangan Active Database System pada Sistem Informasi Pelayanan Harga Pasar. *Jurnal INTENSIF*, 1(1).
- Suprayogi, B., & Rahmanesa, A. (2019). Penerapan Framework Bootstrap Dalam Sistem Informasi Pendidikan Sma Negeri 1 Pacet Cianjur Jawa Barat. *Tematik*, 6(2), 23–30. <https://doi.org/10.38204/tematik.v6i2.244>
- Susilo, M., & Kurniati, R. (2018). Rancang Bangun Website Toko Online Menggunakan Metode Waterfall. *Jurnal Nasional Informatika dan Teknologi Jaringan*. 2(2).
- Syarif, Abdul & Wijayanto, Heru. (2017). Analisis Pengaruh Kemenarikan Desain Website, Kepercayaan Dan Profesionalisme Terhadap Keputusan Pembelian Online Jasa Aqiqah. *Ikraith-Informatika*, 1(2).
- Vidya, Vensy. (2009). Pemanfaatan Website Sebagai Media Promosi pada Fakultas Teknologi Informasi dan Komunikasi Universitas Semarang (Studi

Kasus Pada Fakultas Teknologi Informasi dan Komunikasi – USM). *Jurnal The Messenger*, 1(1).

Wahid, A. A. (2020). Analisis Metode *Waterfall* Untuk Pengembangan Sistem Informasi. *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK*, November, 1–5.

Walim, & Suhardi. (2020). Rancang Bangun Sistem Informasi E-Commerce Dalam Penjualan Hardware Komputer Berbasis *Website*. *Jurnal Penelitian*, 4(2).

**L
A
M
P
I
R
A
N**

Kode controllers registrasi

```

public function registrasi_pengguna()
{
    $this->form_validation->set_rules('nama', 'Nama',
    'required|trim', array('required' => '%s harus diisi'));
    $this->form_validation->set_rules('pass', 'Password',
    'required|trim|min_length[6]|matches[konfirPass]', [
        'required' => '%s harus diisi',
        'matches' => 'Password Salah',
        'min_length' => 'Password Terlalu Pendek'
    ]);
    $this->form_validation->set_rules('konfirPass',
    'Password', 'required|trim|matches[pass]', [
        'required' => '%s harus diisi'
    ]);
    $this->form_validation->set_rules('telepon', 'Nomor HP',
    'required|trim|is_unique[user.telepon]', [
        'is_unique' => 'Nomor Anda Sudah Pernah Teraftar',
        'required' => '%s harus diisi'
    ]);
    $this->form_validation->set_rules('email', 'Email',
    'required|trim|valid_email|is_unique[user.email]', [
        'is_unique' => '%s anda sudah pernah terdaftar',
        'required' => '%s harus diisi',
        'valid_email' => '%s anda tidak valid'
    ]);

    if ($this->form_validation->run() == false) {
        $this->load->view('auth/header');
        $this->load->view('auth/registrasi_pengguna');
        $this->load->view('auth/footer');
    } else {
        //Ambil Data yang Diinput di Registrasi
        $email = $this->input->post('email', true);
        $data = [
            'nama' => htmlspecialchars($this->input-
            >post('nama', true)),
            'password' => password_hash($this->input-
            >post('pass'), PASSWORD_DEFAULT),
            'telepon' => htmlspecialchars($this->input-
            >post('telepon', true)),
            'email' => htmlspecialchars($email),
            'role_id' => 2,
            'is_active' => 0
        ];
    }
}

```

```

        //siapkan token
        $token = base64_encode(random_bytes(32));
        $user_token = [
            'email' => $email,
            'token' => $token,
            'date' => time()
        ];
        //Masukkan kedalam Database
        $this->db->insert('user', $data);
        $this->db->insert('user_token', $user_token);

        //kirim email ke email user yang telah registrasi
        $this->_sendEmail($token, 'verify');

        //Memberikan Pesan Sebelum Redirect
        $this->session->set_flashdata('message', '<div
class="alert alert-success" role="alert">
        Akun Anda Sudah Terdaftar! Silahkan Aktivasi Akun
Anda </div>');
        redirect('auth/login');
    }
}

```

Kode controllers login

```

public function login()
{
    //cek_login_pelanggan();

    $this->form_validation->set_rules('email', 'Email',
'required|trim', [
        'required' => '%s harus diisi'
    ]);
    $this->form_validation->set_rules('pass', 'Password',
'required|trim', [
        'required' => '%s harus diisi'
    ]);

    if ($this->form_validation->run() == false) {
        $this->load->view('auth/header');
        $this->load->view('auth/login_pengguna');
        $this->load->view('auth/footer');
    } else {
        // validasinya sukses
        // _login = method pt
    }
}

```

```

        $this->_login();
    }
}
private function _login()
{
    $email = $this->input->post('email');
    $pass = $this->input->post('pass');

    $user = $this->db->get_where('user', ['email' =>
$email])->row_array();

    if ($user) {
        //jika usernya ada
        if ($user) {

            //Jika usernya aktif
            if ($user['is_active'] == 1) {

                //cek Password
                if (password_verify($pass,
$user['password'])) {
                    $data = [
                        'id_user' => $user['id_user'],
                        'nama' => $user['nama'],
                        'telepon' => $user['telepon'],
                        'email' => $user['email'],
                        'role_id' => $user['role_id'],
                    ];
                    $this->session->set_userdata($data);

                    if ($this->session->userdata('role_id')
== 1) {
                        redirect('admin');
                    } elseif ($this->session-
>userdata('role_id') == 2) {
                        redirect('member/index');
                    } elseif ($this->session-
>userdata('role_id') == 3) {
                        redirect('kasir');
                    }
                } else {
                    $this->session->set_flashdata('message',
'<div class="alert alert-danger" role="alert">
Password Salah </div>');
                    redirect('auth/login');
                }
            } else {

```

```

        $this->session->set_flashdata('message',
        '<div class="alert alert-danger" role="alert">
        Akun anda belum diaktivasi </div>');
        redirect('auth/login');
    }
}
} else {
    $this->session->set_flashdata('message', '<div
class="alert alert-danger" role="alert">
    Email belum terdaftar </div>');
    redirect('auth/login');
}
}

```

Kode *controllers* pemesanan

```

public function pesan(){
    $config['upload_path'] = './assets/desain';
    $config['allowed_types'] = 'pdf';
    $config['max_size']     = 40000;
    //
    $this->load->library('upload');
    $this->upload->initialize($config);
    //
    if( ! $this->upload->do_upload('filedesain')){
        $this->session->set_flashdata('msg','<div class="alert
alert-success text-center">Gagal Upload Desain!</div>');
        redirect('Member/pesanan');
    }else{
        $this->member_model->pesan();
    }
}
}

```

Kode *controllers* input lokasi

```

public function get_lokasi($id){
    $data['title'] = 'Input Lokasi';
    $data['lokasi_pesanan'] = $this->member_model->hrg($id);
    $this->form_validation->set_rules('latitude', 'Latitude',
    'required|trim', [
        'required' => '%s harus diisi'
    ]);
    $this->form_validation->set_rules('longitude', 'Longitude',

```

```

'required|trim', [
    'required' => '%s harus diisi'
]);
$this->form_validation->set_rules('nama_tempat', 'Nama
tempat', 'required|trim', [
    'required' => '%s harus diisi'
]);

if ($this->form_validation->run() == false){
    $this->load->view('member/header_pn', $data);
    $this->load->view('member/lokasi', $data);
    // $this->load->view('member/footer', $data);
} else {
    $data['lokasi'] = $this->member_model->addlokasi();
    $this->session->set_flashdata('msg', '<div class="alert
alert-success text-center">Input lokasi sukses!</div>');
    redirect('member/riwayat_pesan');
}
}

```

Kode *controllers* konfirmasi pembayaran

```

public function bayar($id){
    $data['rekening'] = $this->member_model->rekening();
    $data['hrg'] = $this->member_model->hrg($id);
    $config['upload_path'] = './assets/bukti';
    $config['allowed_types'] = 'jpg|png|jpeg|pdf|svg';
    $config['max_size'] = 30000;
    //
    $this->load->library('upload');
    $this->upload->initialize($config);
    //
    if( ! $this->upload->do_upload('bukti')){
        $this->session->set_flashdata('msg', '<div class="alert
alert-danger text-center">Gagal Upload Bukti Pembayaran!</div>');
        // redirect('Member/pesanan');
        $this->load->view('member/header_pn');
        $this->load->view('member/bayar', $data );
        $this->load->view('member/footer');

    } else {

        $data['pesanan'] = $this->member_model->bayar();
        // $this->member_model->bayar($id);
        // redirect('member/riwayat_pesan');
    }
}

```

```

    }
}

```

Kode *controllers* manajemen pesanan pada admin

```

public function pesanan_rev(){
    $data ['title'] = 'pesanan';
    $data['pesanan'] = $this->Admin_model-
>get_pesanan('pesanan');
    $data['pesanan_diproses'] = $this->Admin_model-
>pesanan_diproses('pesanan');

    $data['pesanan_dikirim'] = $this->Admin_model-
>pesanan_dikirim('pesanan');
    $data['pesanan_selesai'] = $this->Admin_model-
>pesanan_selesai('pesanan');
    $data['pesanan_dicancel'] = $this->Admin_model-
>pesanan_dicancel('pesanan');
    $this->load->view('admin/header', $data);
    $this->load->view('admin/sidebar', $data);
    $this->load->view('admin/topbar', $data);
    $this->load->view('admin/pesanan_rev', $data);
    $this->load->view('admin/footer');
}

public function proses($id){
    $data = array(
        'id'=> $id,
        'status' => '1',
    );
    $this->Admin_model->update_pesanan($data);
    $this->session->set_flashdata('msg','<div class="alert alert-
success text-center">Pesanan telah Diproses!</div>');
    redirect('admin/pesanan_rev');
}

public function dikirim($id){
    $data = array(
        'id'=> $id,
        'status' => '2',
    );
    $this->Admin_model->update_pesanan($data);
    $this->session->set_flashdata('msg','<div class="alert
alert-success text-center">Pesanan telah Dikirim!</div>');
    redirect('admin/pesanan_rev');
}

public function selesai($id){

```

```

$data = array(
    'id'=> $id,
    'status' => '3',
);
$this->Admin_model->update_pesanan($data);
$this->session->set_flashdata('msg', '<div class="alert
alert-success text-center">Pesanan selesai diantarkan!</div>');
redirect('admin/pesanan_rev');
}

```

Kode *controllers* verifikasi pembayaran pada kasir

```

public function verifikasi_pembayaran($id){
    $data = array(
        'id'=> $id,
        'verif_pembayaran' => 'Pembayaran lunas',
    );
    $this->Admin_model->update_transaksi($data);
    $this->session->set_flashdata('msg', '<div class="alert
alert-success text-center">Validasi Pembayaran Sukses!</div>');
    redirect('kasir/pesanan');
}

```

Kode *model* pemesanan

```

public function pesan(){
    $id_user = $this->session->userdata('id_user');
    $upload_data = $this->upload->data();

    $filedesain= $upload_data['file_name'];
    $nama_pemesan =htmlspecialchars($this->input-
>post('nama_pemesan', TRUE));
    $telepon =htmlspecialchars ($this->input->post('telepon',
TRUE));
    $id_produk =htmlspecialchars ($this->input->post('id_produk',
TRUE));
    $id_user = $this->session->userdata('id_user');
    $jumlah_pesanan = htmlspecialchars($this->input-
>post('jumlah_pesanan', TRUE));
    $ukuran =htmlspecialchars($this->input->post('ukuran',
TRUE));
    $waktu_pengantaran = htmlspecialchars($this->input-
>post('waktu_pengantaran', TRUE));
}

```

```

        $catatan =htmlspecialchars($this->input->post('catatan',
TRUE));
        $desain =htmlspecialchars( $this->input->post('desain',
TRUE));
        $pengambilan =htmlspecialchars ($this->input-
>post('pengambilan', TRUE));

        $pesanan = array(
            'kode_pesanan' => $this->kode(),
            'nama_pemesan' => $nama_pemesan,
            'telepon' => $telepon,
            'id_produk' => $id_produk,
            'id_user' =>$id_user,
            'jumlah_pesanan' =>$jumlah_pesanan,
            'ukuran' => $ukuran,
            'waktu_pengantaran' => $waktu_pengantaran,
            'catatan' => $catatan,
            'filedesain' => $filedesain,
            'pengambilan' => $pengambilan
        );

        // insert ke db pesanan
        $this->db->insert('pesanan' , $pesanan );
        redirect('member/riwayat_pesan');
    }

```

Kode *model* input lokasi

```

public function addlokasi($id){
    $data=[
        'kode_pesanan'=>$this->input->post('kode_pesanan', TRUE),
        'latitude' => $this->input->post('latitude', true),
        'longitude' => $this->input->post('longitude', true) ,
        'nama_tempat' => htmlspecialchars($this->input-
>post('nama_tempat', true)) ,
        'keterangan_tempat' => htmlspecialchars($this->input-
>post('keterangan_tempat', true))
    ];
    $this->db->where('id',$id);
    $this->db->update('pesanan',$data);
}

```

Kode *model* konfirmasi pembayaran pelanggan

```
public function bayar($id){
    $upload_data = $this->upload->data();
    $bukti= $upload_data['file_name'];
    $nama=$this->input->post('nama', TRUE);
    $kode_pesanan=$this->input->post('kode_pesanan', TRUE);
    $bank = $this->input->post('bank', TRUE);
    $no_rek = $this->input->post('no_rek', TRUE);

    $data= array(
        'kode_pesanan' => $kode_pesanan,
        'nama' => $nama,
        'bank' => $bank,
        'no_rek' => $no_rek,
        'bukti' => $bukti,
        'status_bayar' => 'Sudah Bayar'
    );
    $this->db->where('id',$id);
    $this->db->update('pesanan',$data);
    redirect('member/riwayat_pesan');
}
```

Kode *model* manajemen pesanan admin

```
public function get_pesanan($table){
    $this->db->select('*');
    $this->db->join('produk', 'produk.id_produk =
pesanan.id_produk','left');
    $this->db->join('transaksi', 'transaksi.kode_pesanan=
pesanan.kode_pesanan','left');
    $this->db->where('pesanan.status = 0');
    return $this->db->get($table)->result_array();
}

public function pesanan_diproses($table){
    $this->db->select('*');
    $this->db->join('produk', 'produk.id_produk =
pesanan.id_produk', 'left');
    $this->db->join('transaksi', 'transaksi.kode_pesanan=
pesanan.kode_pesanan','left');
    $this->db->where('pesanan.status = 1');
    return $this->db->get($table)->result_array();
}
```

```

public function pesanan_dikirim($table){
    $this->db->select('*');
    $this->db->join('produk', 'produk.id_produk =
pesanan.id_produk', 'left');
    $this->db->join('transaksi', 'transaksi.kode_pesanan=
pesanan.kode_pesanan','left');
    $this->db->where('pesanan.status = 2');
    return $this->db->get($table)->result_array();
}
public function pesanan_selesai($table){
    $this->db->select('*');
    $this->db->join('produk', 'produk.id_produk =
pesanan.id_produk', 'left');
    $this->db->join('transaksi', 'transaksi.kode_pesanan=
pesanan.kode_pesanan','left');
    $this->db->where('pesanan.status = 3');
    return $this->db->get($table)->result_array();
}
public function pesanan_dicancel($table){
    $this->db->select('*');
    $this->db->join('produk', 'produk.id_produk =
pesanan.id_produk', 'left');
    $this->db->join('transaksi', 'transaksi.kode_pesanan=
pesanan.kode_pesanan','left');
    $this->db->where('pesanan.status = 5');
    return $this->db->get($table)->result_array();
}

public function update_pesanan($data){
    $this->db->where('id', $data['id']);
    $this->db->update('pesanan', $data);
}

```

Kode input lokasi

```

<script>
    // you want to get it of the window global
    const providerOSM = new GeoSearch.OpenStreetMapProvider();

    //leaflet map
    var leafletMap = L.map('map', {
        fullscreenControl: true,
        // OR
        fullscreenControl: {
            pseudoFullscreen: false // if true, fullscreen to page

```

```

width and height
    },
    minZoom: 2
  }).setView([-5.133998,119.481538], 15);

  L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {
    attribution: '&copy; <a
href="http://osm.org/copyright">OpenStreetMap</a> contributors'
  }).addTo(leafletMap);
  let theMarker = {};

  //memunculkan mamrker ketika di klik
  leafletMap.on('click',function(e) {
    //ambil koordinat
    let latitude = e.latlng.lat.toString().substring(0,15);
    let longitude = e.latlng.lng.toString().substring(0,15);
    document.getElementById("latitude").value = latitude;
    document.getElementById("longitude").value = longitude;
    let popup = L.popup()
      .setLatLng([latitude,longitude])
      .setContent("Kordinat : " + latitude + " -
"+ longitude )
      .openOn(leafletMap);

    if (theMarker != undefined) {
      leafletMap.removeLayer(theMarker);
    };
    theMarker =
L.marker([latitude,longitude]).addTo(leafletMap);
  });

  const search = new GeoSearch.GeoSearchControl({
    provider: providerOSM,
    style: 'bar',
    searchLabel: 'Cari alamat anda',
    showMarker: true,
  });
  leafletMap.addControl(search);
</script>

```