
Implementasi *Framework Laravel* pada Aplikasi Pemesanan Jasa *Photography* menggunakan Metode *Rapid Application Development*

Amirul^{1*}, Jaroji², Muhammad Asep Subandri³

Program Studi Sarjana Terapan Rekayasa Perangkat Lunak, Jurusan Teknik Informatika
Politeknik Negeri Bengkalis, Indonesia

amirulputra71350@gmail.com^{1*}, jaroji@polbeng.ac.id², msubandri@polbeng.ac.id³

Alamat: F552+G9C, Sungai Alam, Kec. Bengkalis, Kabupaten Bengkalis, Riau 28714

Korespondensi penulis: amirulputra71350@gmail.com

Abstract: *The advancement of technology in the field of photography has become an increasingly growing trend over time. In the context of Indonesia, the history of photography began in 1840, in line with the discovery of photography by Juriaan Munich, a Dutch media officer. Photography has been a part of people's lifestyle for centuries, ranging from the use of simple digital cameras to sophisticated devices that require specialized skills. Lensa Photography, as a photography studio in Bengkalis Regency, also faces challenges in the booking process. Manual processes through telephone or direct visits often result in scheduling errors. To address this issue, this study has successfully developed a website-based Lensa Photography booking system using the Laravel framework and the Rapid Application Development (RAD) development method. This booking system has successfully met its objectives by providing the following features, More efficient schedule management, Improved scheduling accuracy, Time and cost savings, Increased customer satisfaction, This booking system has been adopted by Lensa Photography and has provided significant benefits for the company, including, More organized and organized booking schedules, Reduced scheduling errors, More efficient booking time and costs, Increased customer satisfaction*

Keywords: *Booking system, Photography, RAD, Website*

Abstrak: Kemajuan teknologi dalam dunia seni fotografi telah menjadi tren yang semakin berkembang seiring berjalannya waktu. Dalam konteks Indonesia, sejarah fotografi dimulai pada tahun 1840, sejalan dengan penemuan fotografi oleh Juriaan Munich, seorang petugas media asal Belanda. Fotografi telah menjadi bagian dari gaya hidup masyarakat selama berabad-abad, mulai dari penggunaan kamera digital sederhana hingga perangkat canggih yang membutuhkan keahlian khusus. Lensa *Photography*, sebagai studio foto di Kabupaten Bengkalis, juga menghadapi tantangan dalam proses pemesanan. Proses manual melalui telepon atau kunjungan langsung sering menghasilkan kesalahan penjadwalan. Untuk mengatasi permasalahan ini, penelitian ini telah berhasil mengembangkan sistem pemesanan Lensa *Photography* berbasis website dengan menggunakan kerangka kerja *Laravel* dan metode pengembangan *Rapid Application Development*. Sistem pemesanan ini telah berhasil memenuhi tujuannya dengan menyediakan fitur-fitur berikut, Kelolaan jadwal yang lebih efisien, Peningkatan akurasi penjadwalan, Penghematan waktu dan biaya, serta Peningkatan kepuasan pelanggan. Sistem pemesanan ini telah diadopsi oleh Lensa *Photography* dan telah memberikan manfaat yang signifikan bagi perusahaan, antara lain, jadwal pemesanan yang lebih rapi dan terorganisir, Kesalahan penjadwalan berkurang, Waktu dan biaya pemesanan lebih efisien, serta Kepuasan pelanggan meningkat.

Kata Kunci: Sistem pemesanan, Pemesanan, *photography*, *RAD*, *Website*.

1. PENDAHULUAN

Di zaman yang semakin berkembang ini, kemajuan di segala bidang teknologi semakin beragam, termasuk dunia seni fotografi. Fotografi mulai masuk di Indonesia pada era 1840 setahun setelah fotografi di temukan seseorang petugas media, Juriaan Munich yang berasal dari Belanda. Selama berabad-abad, fotografi telah dikenal menjadi salah satu kegemaran masyarakat luas. Mulai dari fotografi sederhana dengan kamera digital biasa atau handphone, sampai fotografi profesional dengan menggunakan kamera canggih yang membutuhkan teknik dan keahlian khusus oleh pemakai. (Pratiwi & Minarni, 2020)

Lensa *Photography* adalah sebuah usaha foto studio yang berlokasi di Kabupaten Bengkalis. Lensa *Photography* menyediakan jasa foto untuk pernikahan, keluarga, kehamilan, bayi, dan kelulusan. Proses pemesanan pada Lensa *Photography* dilakukan melalui telepon atau kunjungan langsung, namun, proses ini sering mengakibatkan kesalahan dalam penjadwalan. Pihak Lensa *Photography* ingin adanya sebuah sistem untuk membantu mengorganisir penjadwalan sehingga pelanggan bisa melihat status pesanan dan pihak Lensa *Photography* dapat menghindari jadwal ganda.

Untuk mengatasi masalah tersebut, penelitian ini dilakukan dengan tujuan untuk membangun sistem pemesanan Lensa *Photography* online berbasis *website* yang mana dengan adanya *platform* ini bisa mengorganisir informasi Lensa *Photography* dengan lebih efisien dan efektif. Pada penelitian ini peneliti menggunakan *framework web Laravel* dan Metode pengembangan *Rapid Application Development* dalam membangun pemesanan Lensa *Photography* online berbasis *website*. *Laravel* dipilih karena menggunakan arsitektur *Model-View-Controller* (MVC) yang membantu dalam membuat kode yang rapi dan terorganisir agar memudahkan dalam pengembangan aplikasi sistem pemesanan Lensa *Photography* online berbasis *website* dan metode *Rapid Application Development* dipilih karena pengembangan yang cepat, fleksibel, dan memenuhi kebutuhan pengguna.

Dengan adanya penelitian ini diharapkan pemilik studio Lensa *Photography* dapat mengelola jadwal lebih efisien, dan peneliti dapat menggunakan hasil penelitian untuk pengembangan pemesanan selanjutnya.

2. TINJAUAN PUSTAKA

Kajian Terdahulu

Penelitian pertama dengan judul menurut Wistika, dkk (2023) yaitu Sistem Informasi Pemesanan Jasa Fotografi pada Julian *Photography* Menggunakan *Framework Laravel*, membahas tentang Julian *Photography* merupakan salah satu usaha jasa fotografi yang berlokasi di Kabupaten Jembrana, Bali. Penelitian kedua menurut Endra, dkk (2022) yaitu Analisis Perbandingan Bahasa Pemrograman *PHP Laravel* dengan *PHP Native* pada Pengembangan *Website* Untuk merancang sebuah *website* banyak *developer website* menggunakan bahasa pemrograman *PHP*.

Penelitian ketiga menurut Nurwardani, dkk (2022) yaitu Implementasi Metode *Rapid Application Development (RAD)* Terhadap Sistem Pemesanan *Furniture* Berbasis *Web* Pada Maldini Perabot Dalam keadaan perekonomian yang semakin sulit ini banyak terjadi persaingan diberbagai bidang kehidupan, termasuk didalamnya persaingan dalam dunia bisnis. Penelitian keempat menurut Dedi, dkk (2021) yaitu Perancangan Sistem Informasi Promosi Jasa Foto dan Studio Musik Pada M2N *Studio Production*, membahas tentang M2N *Studio Production* merupakan salah satu bisnis di bidang pelayanan jasa foto dan studio musik. Penelitian kelima menurut Saputra & Borman (2020) yaitu sistem informasi pelayanan jasa foto berbasis Android (studi kasus: ace *photography way kanan*), membahas tentang ACE *Photography Way Kanan* adalah bisnis yang berjalan dibidang pelayanan jasa foto.

Framework Laravel

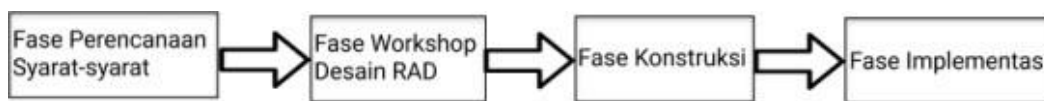
Laravel adalah sebuah *framework web* berbasis *PHP* yang *open-source* dan tidak berbayar, diciptakan oleh Taylor Otwell dan diperuntukkan untuk pengembangan penjabaran *web* yang menggunakan pola *MVC*. Struktur pola *MVC* pada *laravel* sedikit berbeda pada struktur pola *MVC* ambang umumnya. Di *laravel* siap *routing* yang menjembatani renggangan *request* berpokok *user* dan *controller*. Jadi *controller* tidak langsung mengikuti *request* tersebut.

Sebuah *framework* aplikasi *web* biasanya melaksanakan tuangan formasi yang disebut *Model, View dan Controller* atau yang khalayak sebut seperti *MVC*. Modul model memuat kelas-kelas yang mewakili tabel pada *database* yang memegang instances yang digunakan menjelang memalsukan *database*. Model biasanya digunakan seperti penghubung antar modul *controller* dengan *database*. Modul *controller* adalah kelas-kelas yang dibuat oleh *programmer* untuk menangani logika program dan *user events*. Pada aplikasi yang menggunakan pola *MVC controller* bertindak sebagai otak dari sistem, menjembatani hubungan antara *model dan view*. Modul *view* berfungsi untuk menerima dan menampilkan data yang dikirim oleh *controller*.

View dalam aplikasi berbasis *website* biasanya berbentuk kumpulan halaman *HTML* (Yudanto, dkk 2017).

Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah model proses perkembangan *software* sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Tahapan-tahapan dalam RAD terdiri dari 4 yaitu *Requirements Planning*, *Design Workshop*, dan *Implementation*.



Gambar 1 Metode RAD

sumber: (Kendall, 2006)

a. *Requirements Planning* (Perencanaan Syarat-syarat)

Dalam tahap ini diketahui apa saja yang menjadi kebutuhan sistem yaitu dengan mengidentifikasi kebutuhan informasi dan masalah yang dihadapi untuk menentukan tujuan, batasan-batasan sistem, kendala dan juga alternatif pemecahan masalah. Analisis digunakan untuk mengetahui perilaku sistem dan juga untuk mengetahui aktivitas apa saja yang ada dalam sistem tersebut.

b. *Desain Workshop*

Mengidentifikasi solusi alternatif dan solusi yang terbaik. Kemudian membuat desain bisnis, dan desain pemrograman untuk data yang telah didapatkan dan dimodelkan dalam arsitek sistem *tool* yang biasanya digunakan dalam pemodelan sistem biasanya menggunakan *Unified Modelling Language* (UML).

c. *Konstruksi*

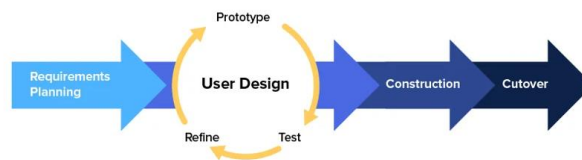
Fase konstruksi merupakan fase eksekusi dalam bentuk pembuatan *script* program dan merupakan kelanjutan dari fase kedua. Pada fase ini juga menunjukkan platform, *hardware*, *software* yang digunakan. Setiap desain yang dibuat pada fase sebelumnya, akan ditingkatkan dengan menggunakan perangkat RAD.

d. *Implementation* (Implementasi)

Setelah desain dilakukan selanjutnya sistem diimplementasikan(*coding*) bahasa yang mudah dimengerti oleh mesin yang diwujudkan dalam bentuk program (Mandasari & Kaban, 2020).

3. METODOLOGI PENELITIAN

Peneliti menggunakan metode *Rapid Application Development* (RAD). Metode ini berfokus pada pengembangan aplikasi melalui iterasi yang cukup sering dengan mengikutsertakan pengguna selama pengembangan yang membuat aplikasi dapat dibangun tidak begitu lama. Terdapat beberapa tahapan yang dilakukan, yaitu Analisa Kebutuhan, *User Design*, konstruksi, dan diakhiri dengan *Cutover*..



Gambar 2. Metode *Rapid Application Development* (RAD)

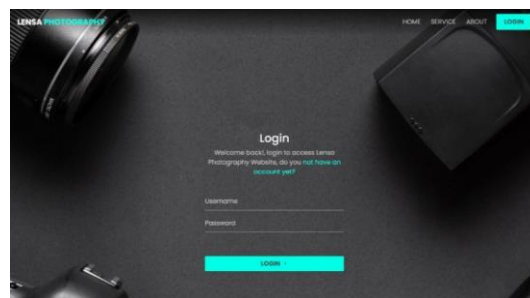
4. HASIL DAN PEMBAHASAN

Hasil

Hasil dari penelitian ini merupakan sebuah aplikasi pemesanan jasa *photography* menggunakan metode *rapid application development*. Berikut adalah hasil tampilan aplikasi antara lain:

a. *Login*

Pada menu *login* berguna untuk pengguna masuk kedalam aplikasi dan hanya pengguna yang terdaftar yang bisa mengakses aplikasi. Berikut potongan *source code* dan hasil tampilan *login*



Gambar 3. Tampilan *Login*

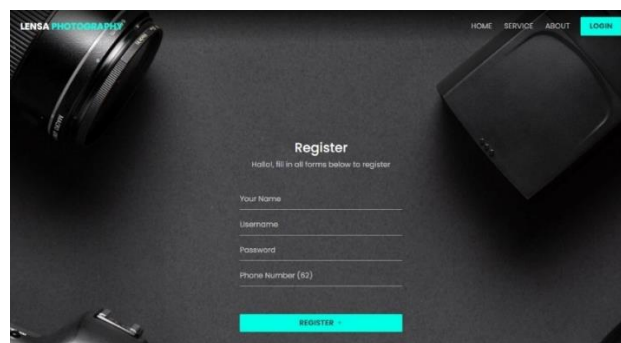
```
public function login(Request $request)
{
    $validasi = $request->validate([
        'username' => 'required',
        'password' => 'required',
    ]);

    if (Auth::attempt($validasi)) {
        $request->session()->regenerate();

        $getRole = User::where('username', $request->username)-
        >first();
        if ($getRole->role == 'Admin') {
            return redirect()->intended('/dashboard');
        } else {
            return redirect()->intended('/service');
        }
    } else {
        return back()->with('error', 'Username atau password salah!');
    }
}
```

b. Register

Pada menu *register* berguna untuk pengguna mendaftarkan sebelum masuk kedalam aplikasi dan hanya pengguna yang terdaftar yang bisa mengakses aplikasi. Berikut potongan *source code* dan hasil tampilan *register* :



Gambar 4. Tampilan Register

```
public function registerPage()
{
    return view('content.client.register');
}

public function register(Request $request)
{
```

```

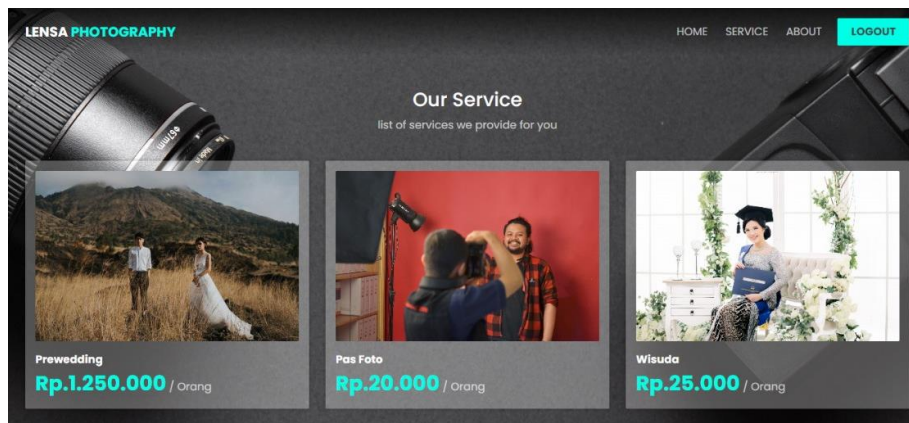
$request['role'] = 'Penyewa';
$validasi = $request->validate([
    'name' => 'required|alpha',
    'username' => 'required|unique:users',
    'password' => 'required|min:8',
    'phone_number' => 'required|in:62',
    'role' => 'required',
]);

$validasi['password'] = Hash::make($validasi['password']);
User::create($validasi);
return redirect('/login/page');
}

```

c. Service

Pada menu beranda pengguna dapat melihat jasa yang ditampilkan oleh lensa *photography* pada aplikasi. Berikut potongan *source code* dan hasil service:



Gambar 5. Tampilan Service

```

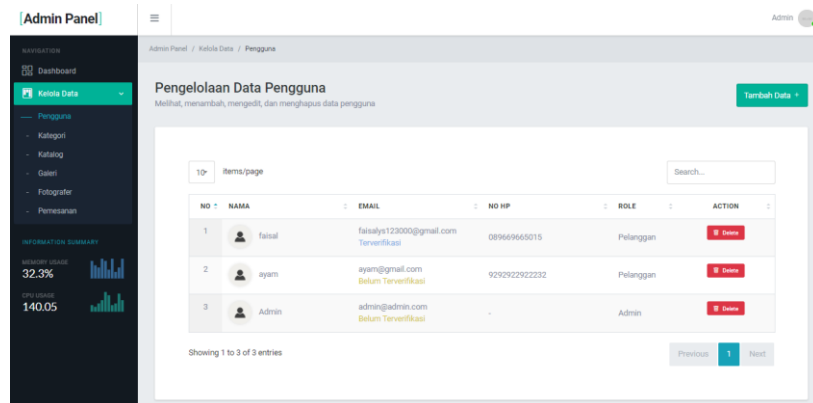
<div class="container mt-5">
  <div class="box-content">
    <h3 class="tagline">Our Service</h3>
    <p class="subline">list of services we provide for you</p>
    <div class="row">
      @foreach ($categories as $category)
        <div class="col-md-6 col-lg-4 mb-4">
          <a href="" class="text-decoration-none">
            <div class="box-service">
              
              <b class="type">{{ $category->type }}</b><br>
              <span class="box-harga"><b class="harga">{{
                $category->formatRupiah('price') }}</b> / Orang</span>
            </div >
          </a>
        </div>
      @endforeach
    </div>

```

```
</div>  
</div>  
</div>
```

d. Kelola Data Customer

Pada menu ini *admin* dapat mengelola data *customer* yang telah mendaftar pada aplikasi. Berikut potongan *source code* dan data *customer*:

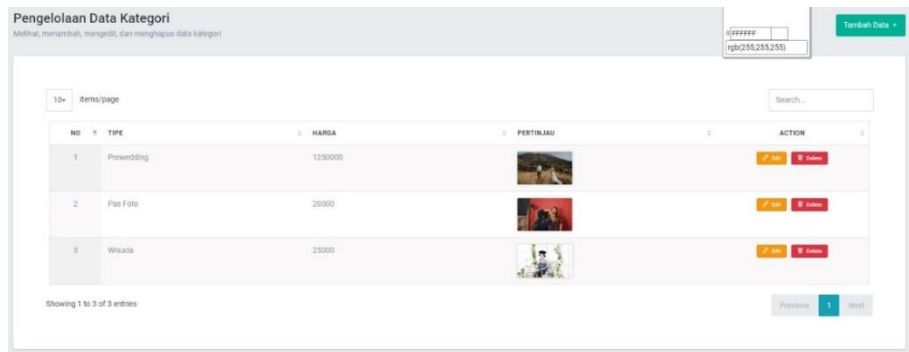


Gambar 6. Tampilan Kelola Data Customer

```
public function store(Request $request)  
{  
    $validasi = $request->validate([  
        'name' => 'required',  
        'username' => 'required',  
        'password' => 'required',  
        'phone_number' => 'required',  
        'role' => 'required',  
        'profile_img' => 'nullable|mimes:jpeg,jpg,png|max:5120',  
    ]);  
  
    $validasi['password'] = Hash::make($validasi['password']);  
  
    if ($request->file('profile_img')) {  
        $validasi['profile_img'] = $request->file('profile_img')->  
>store('berkas_profile_img');  
    }  
  
    User::create($validasi);  
  
    return redirect('/manage/user')->with('success', 'Data pengguna  
berhasil ditambah!');
```

e. Kelola Data Kategori

Pada menu ini *admin* dapat mengelola data kategori jasa yang telah pada aplikasi. Berikut potongan *source code* dan hasil kategori:



Gambar 7. Tampilan Kelola Data Kategori

```

public function store(Request $request)
{
    $validasi = $request->validate([
        'type' => 'required',
        'price' => 'required',
        'category_img' => 'required|mimes:jpeg,jpg,png|max:5120',
    ]);

    if ($request->file('category_img')) {
        $validasi['category_img'] = $request->file('category_img')-
        >store('berkas_category_img');
    }

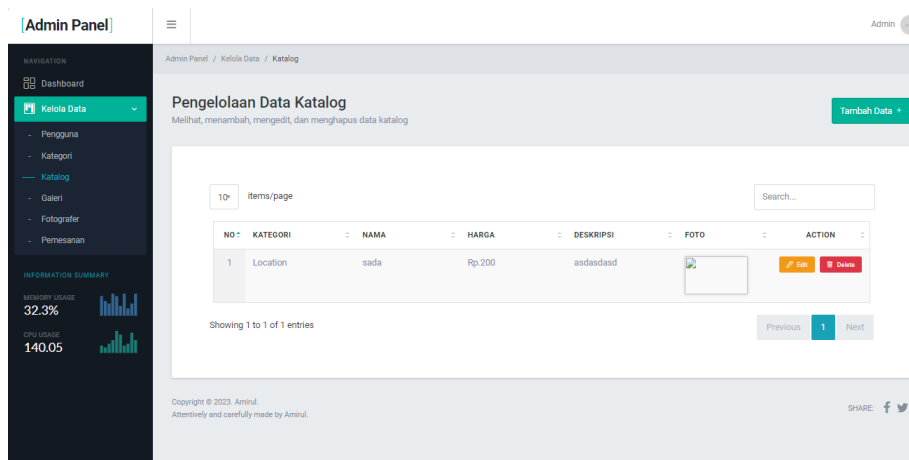
    Category::create($validasi);

    return redirect('/manage/category')->with('success', 'Data
    kategori berhasil ditambah!');
}

```

f. Kelola Data Katalog

Pada menu ini *admin* dapat mengelola data katalog yang telah ada pada aplikasi. Berikut potongan *source code* dan hasil katalog:



Gambar 8. Kelola Data Katalog

```

@include('partials.crud-alert')

<div class="pd-x-20 pd-sm-x-30 pd-t-2 d-flex align-items-center
justify-content-between flex-wrap">
  <div>
    <h4 class="tx-gray-800 mg-b-5">Pengelolaan Data
Katalog</h4>
    <p class="mg-b-0">Melihat, menambah, mengedit, dan
menghapus data katalog</p>
  </div>

  <!-- BASIC MODAL -->
  <a href="" class="btn btn-teal" data-toggle="modal" data-
target="#modalCreateCatalog">Tambah Data <i class="bi bi-
plus"></i></a>
  <div id="modalCreateCatalog" class="modal fade">
    <div class="modal-dialog modal-dialog-vertical-center"
role="document">
      <form action="/manage/catalog" method="POST"
class="modal-content bd-0 tx-14" enctype="multipart/form-data">
        @csrf
        <div class="modal-header pd-y-20 pd-x-25">
          <h6 class="tx-14 mg-b-0 tx-uppercase tx-inverse tx-
bold">Tambah Data Katalog</h6>
          <button type="button" class="close" data-dismiss="modal"
aria-label="Close">
            <span aria-hidden="true">&times;</span>
          </button>
        </div>
        <div class="modal-body pd-25">
          <div class="form-group">
            <label class="form-control-label">Kategori: <span
class="tx-danger">*</span></label>
            <select class="form-control" name="category_id"
required>
              <option>Pilih Kategori</option>
              @foreach ($categories as $category)
                <option value="{{ $category->id }}">{{ $category-
>type }}</option>
              @endforeach
            </select>
            @error('category_id')
              <div class="invalid-feedback">{{ $message }}</div>
            @enderror
          </div>
          <div class="form-group">
            <label class="form-control-label">Nama: <span class="tx-
danger">*</span></label>
            <input class="form-control" type="text" name="name"
placeholder="Masukkan nama katalog" required>

```

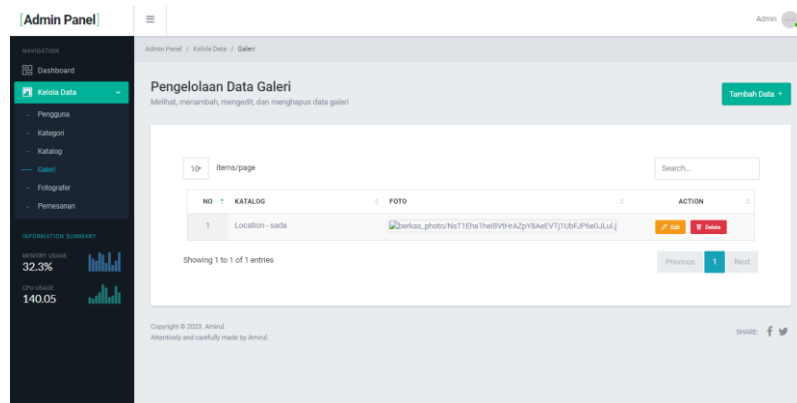
```

    @error('name')
      <div class="invalid-feedback">{{ $message }}</div>
    @enderror
  </div>
  <div class="form-group">
    <label class="form-control-label">Harga: <span
class="tx-danger">*</span></label>
    <input class="form-control" type="text" name="price"
placeholder="Masukkan harga katalog" required>
    @error('price')
      <div class="invalid-feedback">{{ $message }}</div>
    @enderror
  </div>

```

g. Kelola Galeri

Pada menu ini *admin* dapat mengelola data galeri yang telah ada pada aplikasi. Berikut potongan *source code* dan hasil galeri:



Gambar 9. Kelola Galeri

```

<a href="" class="btn btn-teal" data-toggle="modal" data-
target="#modalCreateGallery">Tambah Data <i class="bi bi-
plus"></i></a>
  <div id="modalCreateGallery" class="modal fade">
    <div class="modal-dialog modal-dialog-vertical-center"
role="document">
      <form action="/manage/gallery" method="POST"
class="modal-content bd-0 tx-14" enctype="multipart/form-data">
        @csrf
        <div class="modal-header pd-y-20 pd-x-25">
          <h6 class="tx-14 mg-b-0 tx-uppercase tx-inverse tx-
bold">Tambah Data Galeri</h6>
          <button type="button" class="close" data-dismiss="modal"
aria-label="Close">
            <span aria-hidden="true">&times;</span>
          </button>
        </div>
        <div class="modal-body pd-25">
          <div class="form-group">

```

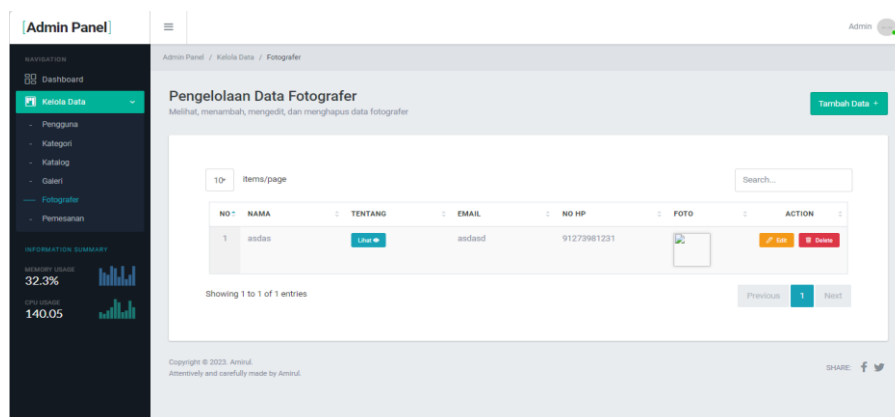
```

<label class="form-control-label">Katalog: <span
class="tx-danger">*</span></label>
<select class="form-control" name="catalog_id"
required>
  <option>Pilih Katalog</option>
  @foreach ($catalogs as $catalog)
    <option value="{{ $catalog->id }}">{{ $catalog-
>category->type }} - {{ $catalog->name }}</option>
  @endforeach
</select>
@error('catalog_id')
  <div class="invalid-feedback">{{ $message }}</div>
@enderror
</div>
<div class="form-group">
  <label class="form-control-label">Foto Pertinjau: <span
class="tx-danger">*</span></label>
  <input class="form-control @error('photo') is-invalid
@enderror" type="file" name="photo" placeholder="Masukkan foto
pertinjau" required>
  @error('photo')
    <div class="invalid-feedback">{{ $message }}</div>
  @enderror
</div>
</div>

```

h. Kelola *photographer*

Pada menu ini *admin* dapat mengelola data *photographer* yang telah ada pada aplikasi. Berikut potongan *source code* dan hasil *photographer*:



Gambar 10. Kelola *Photographer*

```

<div class="br-pagebody">
  <div class="br-section-wrapper">
    {{-- <h6 class="tx-gray-800 tx-uppercase tx-bold tx-14 mg-b-10">Basic Responsive DataTable</h6>
    <p class="mg-b-25 mg-lg-b-50">Searching, ordering and
    paging goodness will be immediately added to the table, as shown in
    this example.</p> --}}

    <div class="table-wrapper">
      <table id="datatable1" class="table display responsive
      nowrap">
        <thead>
          <tr>
            <th class="wd-5p text-center">No</th>
            <th class="wd-20p">Nama</th>
            <th class="wd-20p"> TENTANG</th>
            <th class="wd-20p">Email</th>
            <th class="wd-20p">No HP</th>
            <th class="wd-20p">Foto</th>
            <th class="wd-15p text-center">Action</th>
          </tr>
        </thead>
        <tbody>
          @foreach ($data as $dataPhotographer)
            <tr>
              <td class="text-center">{{ $loop->iteration }}</td>
              <td>{{ $dataPhotographer->name }}</td>
              <td>
                <!-- BASIC MODAL -->
                <a href="" class="btn btn-info tx-10 pd-x-10 pd-y-5"
                data-toggle="modal" data-target="#modalAboutPhotographer{{
                $dataPhotographer->id }}">Lihat <i class="bi bi-eye-fill"></i></a>
              </td>
              <td>{{ $dataPhotographer->email }}</td>
              <td>{{ $dataPhotographer->phone_number }}</td>
              <td>
                type }}">
              </td>
              <td>
                <div class="d-flex justify-content-center">
                  {{-- Tombol Modal Update --}}
                  <a href="" class="btn btn-warning tx-10 pd-x-10 pd-y-
                  5 mr-2" data-toggle="modal" data-
                  target="#modalUpdatePhotographer{{ $dataPhotographer->id
                  }}"><i class="bi bi-pen mr-1"></i> Edit</a>
                  <div data-kode="{{ $dataPhotographer->id }}"
                  class="btn btn-danger tx-10 pd-x-10 pd-y-5 swal-confirm">

```

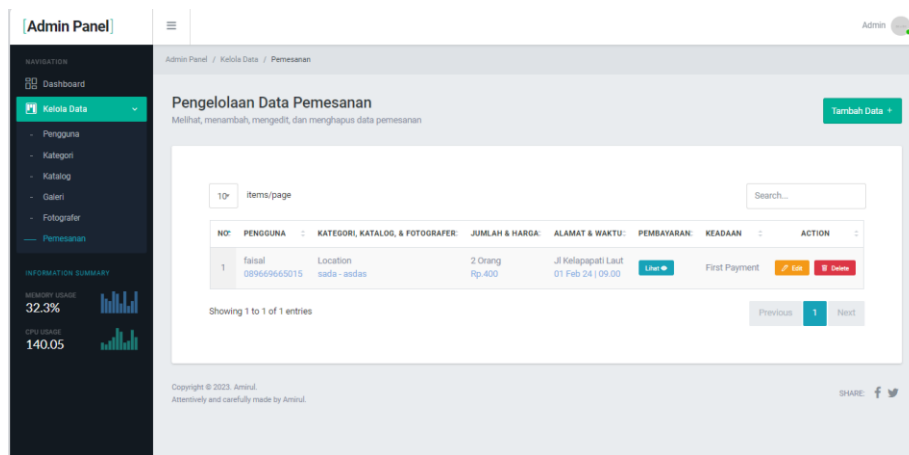
```

<form action="/manage/photographer/{ {
$dataPhotographer->id } }" id="delete{ { $dataPhotographer->id } }"
method="post">
    @csrf
    @method('delete')
</form>
<i class="bi bi-trash mr-1"></i>
Delete
</div>
</div>
</td>
</tr>
@endforeach
    
```

i. Kelola Pembayaran

Pada menu ini *admin* dapat mengelola data pembayaran yang telah ada pada aplikasi.

Berikut potongan *source code* dan hasil pembayaran:



Gambar 11. Kelola Pembayaran

```

<div class="modal-body pd-25">
    <div class="form-group">
        <label class="form-control-label">Pengguna: <span
class="tx-danger">*</span></label>
        <select class="form-control" name="user_id" required>
            <option>Pilih Pengguna</option>
            @foreach ($users as $user)
                <option value="{{ $user->id }}">{{ $user->name
                }}</option>
            @endforeach
        </select>
    </div>
    <div class="form-group">
        <label class="form-control-label">Katalog: <span class="tx-
danger">*</span></label>
        <select class="form-control" name="catalog_id" required>
            <option>Pilih Tipe</option>
            @foreach ($catalogs as $catalog)
    
```

```

        <option value="{{ $catalog->id }}">{{ $catalog-
>category->type }} - {{ $catalog->name }}</option>
    @endforeach
</select>
</div>
<div class="form-group">
    <label class="form-control-label">Fotografer: <span
class="tx-danger">*</span></label>
    <select class="form-control" name="photographer_id"
required>
        <option>Pilih Fotografer</option>
        @foreach ($photographers as $photographer)
            <option value="{{ $photographer->id }}">{{
$photographer->name }}</option>
        @endforeach
    </select>
</div>
<div class="form-group">
    <label class="form-control-label">Alamat: <span class="tx-
danger">*</span></label>
    <input class="form-control @error('address') is-invalid
@enderror" type="text" name="address" placeholder="Masukkan
alamat" required>
        @error('address')
    <div class="invalid-feedback">{{ $message

```

Kelebihan Membangun *website* dengan menggunakan *framework Laravel*:

- 1) *Website* dibangun dengan cepat karena dengan menggunakan *framework Laravel* hanya sedikit baris kode saja yang digunakan sebagai contoh dapat dilihat pada perbandingan 2 codingan halaman login berikut:

a) *Code Login Framework Laravel* Pada *website photography*

```

if (!isset($_SESSION['csrf_token'])) {
    $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
}
public function login(Request $request)
{
    $validasi = $request->validate([
        'username' => 'required',
        'password' => 'required',
    ]);

    if (Auth::attempt($validasi)) {
        $request->session()->regenerate();

        $getRole = User::where('username', $request->username)->first();
        if ($getRole->role == 'Admin') {
            return redirect()->intended('/dashboard');

```

```
    } else {  
        return redirect()->intended('/service');  
    }  
  
    } else {  
        return back()->with('error', 'Username atau password salah!');  
    }  
}
```

b) Code login PHP native

```
<?php  
session_start(); // Start session to manage user login state  
  
// Define database credentials (replace with your actual values)  
$db_host = 'your_database_host';  
$db_name = 'your_database_name';  
$db_user = 'your_database_username';  
$db_password = 'your_database_password';  
  
// Connect to the database  
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);  
  
// Check for connection error  
if (!$conn) {  
    die("Connection failed: " . mysqli_connect_error());  
}  
  
// Handle login form submission  
if (isset($_POST['login'])) {  
    $username = mysqli_real_escape_string($conn, $_POST['username']);  
    $password = mysqli_real_escape_string($conn, $_POST['password']);  
  
    // Validate username and password (replace with your validation logic)  
    if (!empty(<span class="math-inline">username) &&  
        \!empty(</span>password)) {  
        $sql = "SELECT * FROM users WHERE username = '$username'";  
        $result = mysqli_query($conn, $sql);  
  
        if (mysqli_num_rows($result) > 0) {  
            $row = mysqli_fetch_assoc($result);  
  
            if (password_verify($password, $row['password'])) { // Use  
password_verify for secure hashing  
                $_SESSION['user_id'] = $row['id']; // Store user ID in session  
                header("Location: welcome.php"); // Redirect to welcome page on  
successful login  
            } else {  
                $error_message = "Invalid username or password.";  
            }  
        }  
    }  
}
```



```

    } else {
        $error_message = "Invalid username or password.";
    }
} else {
    $error_message = "Please enter your username and password.";
}
}
<?php if (isset($error_message)) : ?>
    <div class="error-message"><?php echo $error_message; ?></div>
<?php endif; ?>
<form class="login-form" method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
    <div class="input-field">
        <label for="username">Username:</label>
        <input type="text" name="username" id="username">
    </div>
    <div class="input-field">
        <label for="password">Password:</label>
        <input type="password" name="password" id="password">
    </div>
    <button type="submit" name="login">Login</

```

Dari perbandingan 2 code diatas dapat dilihat bahwa *framework Laravel* menggunakan lebih sedikit *code* dibandingkan dengan *php native*.

2) Keamanan

Dengan menggunakan *framework Laravel* dalam mengembangkan sebuah *website* akan jauh lebih aman jika dibandingkan dengan *php native*, *framework Laravel* telah menyediakan *csrf token* dan *XSS protection* yang mana berfungsi sebagai mekanisme pertahanan penting untuk melindungi aplikasi *web* dari serangan. Berikut adalah perbandingan antara penggunaan *csrf token* dan tidak menggunakan *csrf token* dan *XSS protection*

a) Code Login Framework Laravel Pada website photography

```

if (!isset($_SESSION['csrf_token'])) {
    $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
}
public function login(Request $request)
{
    $validasi = $request->validate([
        'username' => 'required',
        'password' => 'required',
    ]);
    if (Auth::attempt($validasi)) {
        $request->session()->regenerate();

        $getRole = User::where('username', $request->username)->first();
        if ($getRole->role == 'Admin') {

```

```
    return redirect()->intended('/dashboard');  
  } else {  
    return redirect()->intended('/service');  
  }  
  
  } else {  
    return back()->with('error', 'Username atau password salah!');  
  }  
}
```

b) *Code login PHP native*

```
php  
session_start(); // Start session to manage user login state  
  
// Define database credentials (replace with your actual values)  
$db_host = 'your_database_host';  
$db_name = 'your_database_name';  
$db_user = 'your_database_username';  
$db_password = 'your_database_password';  
  
// Connect to the database  
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);  
  
// Check for connection error  
if (!$conn) {  
    die("Connection failed: " . mysqli_connect_error());  
}  
  
// Handle login form submission  
if (isset($_POST['login'])) {  
    $username = mysqli_real_escape_string($conn, $_POST['username']);  
    $password = mysqli_real_escape_string($conn, $_POST['password']);  
  
    // Validate username and password (replace with your validation logic)  
    if (!empty(<span class="math-inline">username) &&  
        \!empty(</span>password)) {  
        $sql = "SELECT * FROM users WHERE username = '$username'";  
        $result = mysqli_query($conn, $sql);  
  
        if (mysqli_num_rows($result) > 0) {  
            $row = mysqli_fetch_assoc($result);  
  
            if (password_verify($password, $row['password'])) { // Use  
password_verify for secure hashing  
                $_SESSION['user_id'] = $row['id']; // Store user ID in session  
                header("Location: welcome.php"); // Redirect to welcome page on  
successful login  
            } else {  
                $error_message = "Invalid username or password.";  
            }  
        }  
    }  
}
```

```

    } else {
        $error_message = "Invalid username or password.";
    }
} else {
    $error_message = "Please enter your username and password.";
}
}
<?php if (isset($error_message)) : ?>
    <div class="error-message"><?php echo $error_message; ?></div>
<?php endif; ?>
<form class="login-form" method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
    <div class="input-field">
        <label for="username">Username:</label>
        <input type="text" name="username" id="username">
    </div>
    <div class="input-field">
        <label for="password">Password:</label>
        <input type="password" name="password" id="password">
    </div>
    <button type="submit" name="login">Login</

```

Dengan menggunakan *framework Laravel* akan mendapatkan fitur keamanan bawaan seperti *CSRF* dan *XSS protection* yang berguna sebagai melindungi aplikasi *web* dari berbagai serangan. *CSRF protection* mencegah serangan pemalsuan permintaan, melindungi pengguna dari situs *web* jahat yang secara otomatis mengirimkan permintaan berbahaya. *XSS protection* mencegah injeksi skrip berbahaya, melindungi pengguna dari skrip berbahaya yang disematkan di halaman *web* sah. Secara keseluruhan, *CSRF* dan *XSS protection* meningkatkan keamanan aplikasi *web*, meningkatkan kepercayaan pengguna, dan mencegah penyalahgunaan dan kerugian.

- 3) Untuk mempertimbangkan efisiensi dalam pembuatan fungsi dan kode program, sebuah parameter penting dalam pengembangan perangkat lunak. *Framework* adalah kerangka kerja yang menyediakan berbagai pustaka untuk mendukung pengembangan aplikasi. Beberapa sampel telah digunakan dalam penelitian ini untuk membandingkan efisiensi pembuatan kode program
 - a) *Code Login Framework Laravel Pada website photography*

```

if (!isset($_SESSION['csrf_token'])) {
    $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
}
public function login(Request $request)
{
    $validasi = $request->validate([
        'username' => 'required',
        'password' => 'required',

```

```
]);

if (Auth::attempt($validasi)) {
    $request->session()->regenerate();

    $getRole = User::where('username', $request->username)->first();
    if ($getRole->role == 'Admin') {
        return redirect()->intended('/dashboard');
    } else {
        return redirect()->intended('/service');
    }

} else {
    return back()->with('error', 'Username atau password salah!');
}
}
```

b) Code login PHP native

```
<?php
session_start(); // Start session to manage user login state

// Define database credentials (replace with your actual values)
$db_host = 'your_database_host';
$db_name = 'your_database_name';
$db_user = 'your_database_username';
$db_password = 'your_database_password';

// Connect to the database
$conn = mysqli_connect($db_host, $db_user, $db_password, $db_name);

// Check for connection error
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Handle login form submission
if (isset($_POST['login'])) {
    $username = mysqli_real_escape_string($conn, $_POST['username']);
    $password = mysqli_real_escape_string($conn, $_POST['password']);

    // Validate username and password (replace with your validation logic)
    if (!empty(<span class="math-inline">username) &&
    \!empty(</span>password)) {
        $sql = "SELECT * FROM users WHERE username = '$username'";
        $result = mysqli_query($conn, $sql);

        if (mysqli_num_rows($result) > 0) {
            $row = mysqli_fetch_assoc($result);
```

```

        if (password_verify($password, $row['password'])) { // Use
password_verify for secure hashing
            $_SESSION['user_id'] = $row['id']; // Store user ID in session
            header("Location: welcome.php"); // Redirect to welcome page on
successful login
        } else {
            $error_message = "Invalid username or password.";
        }
    } else {
        $error_message = "Invalid username or password.";
    }
} else {
    $error_message = "Please enter your username and password.";
}
}
<?php if (isset($error_message)) : ?>
    <div class="error-message"><?php echo $error_message; ?></div>
<?php endif; ?>
<form class="login-form" method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
    <div class="input-field">
        <label for="username">Username:</label>
        <input type="text" name="username" id="username">
    </div>
    <div class="input-field">
        <label for="password">Password:</label>
        <input type="password" name="password" id="password">
    </div>
    <button type="submit" name="login">Login</

```

Berdasarkan potongan *code* diatas, dalam *PHP Native*, kode tersebut harus digunakan berulang-ulang pada halaman yang hanya dapat diakses oleh pengguna yang telah login. Namun, pada *Laravel*, cukup menggunakan sebuah *Middleware Auth* yang dideklarasikan pada *route Laravel*. Dengan demikian, saat menggunakan *Laravel* untuk membuat suatu perintah atau fungsi, pengembang tidak perlu memulainya dari awal; cukup dengan memanggil *class* yang diperlukan sesuai dengan kebutuhan aplikasi.

- 4) *Laravel* menggunakan arsitektur *MVC (Model, View, dan Controller)* dan *OOP (Object Oriented Programming)*. Sementara *PHP Native* hanya menggunakan *OOP* saja, tanpa menggunakan *MVC*. Kehadiran *MVC* dan *OOP* dalam *Laravel* sangat membantu pengembang dalam proses pembangunan aplikasi *web*. Penelitian ini menemukan perbedaan signifikan dalam proses pembangunan *web* pada kedua *platform* tersebut. Hal ini disebabkan oleh ketersediaan dokumentasi yang lengkap mengenai arsitektur *Laravel* dan fungsi-fungsi lainnya yang terdapat pada situs resmi *Laravel*. Sehingga

Pembahasan

Pada tahap ini dilakukan pengujian untuk memastikan aplikasi yang dihasilkan sesuai dengan kebutuhan dan menemukan kesalahan atau kekurangan pada aplikasi yang akan diuji

a. Pengujian *Black Box Testing*

Black box testing merupakan proses menguji guna fitur lunak(aplikasi) ataupun aplikasi dari sudut pandang pengguna, tanpa mengenali struktur *internal* ataupun desain struktur tersebut. Berikut merupakan prosedur pengujian Aplikasi menggunakan *black box testing*.

Tabel 1 Tabel Pengujian *Black Box testing*

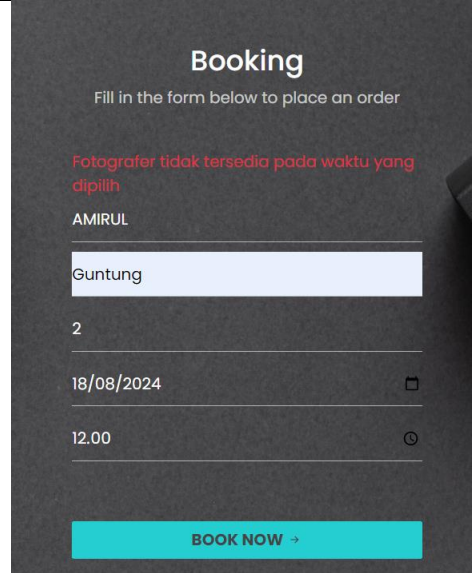
Test Case	Hasil Yang diharapkan	Hasil Pengujian	Kesimpulan
Admin dan pelanggan memasukan <i>username</i> dan <i>password</i>	Pelanggan dan admin dapat melakukan <i>login</i>	Pelanggan dan admin dapat melakukan <i>login</i>	Sesuai
Pelanggan mengakses halaman registrasi lalu mengisi data diri dan melakukan registrasi	Pelanggan dapat melakukan registrasi	Pelanggan dapat melakukan registrasi	Sesuai
Pelanggan mengakses halaman katalog	Pelanggan dapat melihat katalog	Pelanggan dapat melihat katalog	Sesuai
Pelanggan mengakses halaman lalu menekan salah satu katalog	Pelanggan dapat melihat kategori	Pelanggan dapat melihat kategori	Sesuai
Pelanggan mengakses halaman lalu menekan salah satu katalog	Pelanggan dapat melihat <i>galery</i>	Pelanggan dapat melihat <i>galery</i>	Sesuai
Pelanggan mengakses halaman lalu menekan salah satu katalog kemudian melakukan pemesanan	Pelanggan dapat melakukan pemesanan	Pelanggan dapat melakukan pemesanan	Sesuai
Pelangan mengakses halaman pemesanan	Pelanggan dapat melihat status pemesanan	Pelanggan dapat melihat status pemesanan	Sesuai
Pelangan mengakses halaman	Pelanggan dapat mengupload	Pelanggan dapat mengupload bukti pembayaran	Sesuai

**IMPLEMENTASI FRAMEWORK LARAVEL PADA APLIKASI PEMESANAN JASA PHOTOGRAPHY
MENGUNAKAN METODE RAPID APPLICATION DEVELOPMENT**

pemesanan kemudian mengupload bukti pembayaran	bukti pembayaran		
Admin melakukan login kemudian mengakses data pengguna lalu menambah data pengguna	Admin dapat menambah data pengguna	Admin dapat menambah data pengguna	Sesuai
Admin melakukan login kemudian mengakses data kategori lalu menambah kategori	Admin dapat menambah kategori	Admin dapat menambah kategori	Sesuai
Admin melakukan login kemudian mengakses data kategori lalu merubah kategori	Admin dapat merubah kategori	Admin dapat merubah kategori	Sesuai
Admin melakukan login kemudian mengakses data kategori lalu menghapus kategori	Admin dapat menghapus kategori	Admin dapat menghapus kategori	Sesuai
Admin melakukan login kemudian mengakses data katalog lalu menambah katalog	Admin dapat menambah katalog	Admin dapat menambah katalog	Sesuai
Admin melakukan login kemudian mengakses data katalog lalu merubah katalog	Admin dapat merubah katalog	Admin dapat merubah katalog	Sesuai
Admin melakukan login kemudian mengakses data katalog lalu menghapus katalog	Admin dapat menghapus katalog	Admin dapat menghapus katalog	Sesuai
Admin melakukan login kemudian mengakses data galeri lalu menambah galeri	Admin dapat menambah galeri	Admin dapat menambah galeri	Sesuai
Admin melakukan	Admin dapat	Admin dapat merubah galeri	Sesuai

<i>login</i> kemudian mengakses data galeri lalu merubah galeri	merubah galeri		
Admin melakukan <i>login</i> kemudian mengakses data galeri lalu menghapus galeri	Admin dapat menghapus galeri	Admin dapat menghapus galeri	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data <i>photographer</i> lalu menambah <i>photographer</i>	Admin dapat menambah <i>photographer</i>	Admin dapat menambah <i>photographer</i>	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data <i>photographer</i> lalu merubah <i>photographer</i>	Admin dapat merubah <i>photographer</i>	Admin dapat merubah <i>photographer</i>	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data <i>photographer</i> lalu menghapus <i>photographer</i>	Admin dapat menghapus <i>photographer</i>	Admin dapat menghapus <i>photographer</i>	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data pesanan lalu menerima pemesanan	Admin dapat menerima pemesanan	Admin dapat menerima pemesanan	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data pesanan lalu menolak pemesanan	Admin dapat menolak pemesanan	Admin dapat menolak pemesanan	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data pesanan lalu merubah pemesanan	Admin dapat merubah pemesanan	Admin dapat merubah pemesanan	Sesuai
Admin melakukan <i>login</i> kemudian mengakses data pesanan lalu	Admin dapat menghapus pemesanan	Admin dapat menghapus pemesanan	Sesuai

**IMPLEMENTASI FRAMEWORK LARAVEL PADA APLIKASI PEMESANAN JASA PHOTOGRAPHY
MENGUNAKAN METODE RAPID APPLICATION DEVELOPMENT**

menghapus pemesanan			
Pelanggan melakukan <i>login</i> tanpa registrasi terlebih dahulu	Pelanggan sebelum melakukan <i>login</i> harus registrasi terlebih dahulu.	Pelanggan tidak bisa mengakses sebelum melakukan <i>login</i> harus registrasi terlebih dahulu.	Sesuai
Pelanggan melakukan registrasi kemudian menerima <i>email</i> sebelum melakukan transaksi	Setelah pelanggan melakukan registrasi harus melakukan verifikasi <i>email</i> terlebih dahulu	Setelah pelanggan melakukan registrasi pelanggan menerima verifikasi <i>email</i>	Sesuai
Visitors melakukan <i>login</i> tanpa registrasi terlebih dahulu	Sistem dapat memastikan bahwa data yang digunakan dalam sistem tidak dapat diakses oleh orang yang tidak berkepentingan	Sistem dapat memastikan bahwa data yang digunakan dalam sistem tidak dapat diakses oleh orang yang tidak berkepentingan	Sesuai
Pengguna melakukan pemesanan di jadwal yang bersamaan	Sistem menampilkan notifikasi peringatan bahwa fotografer tidak tersedia di jadwal yang dipilih		Sesuai

b. Pengujian sistem ke pengguna

Pengujian aplikasi ke pengguna di lakukan ke beberapa pihak seperti pihak dari Lensa *Photographer* nya langsung dan dari pihak pengguna.

Tabel 2 Tabel wawancara pemilik Lensa *Photography*

Nama Narasumber : Salman				
Jabatan/Posisi : Pemilik Lensa <i>Photography</i>				
No	Pertanyaan	Setuju	Kurang Setuju	Tidak Setuju
1	Fitur dan tampilan aplikasi mudah di mengerti	✓		
2	Sistem ini membantu dalam menghemat waktu dalam mengelola jadwal pemesanan.	✓		
3	Sistem ini membantu mengurangi kesalahan penjadwalan.	✓		
4	Sistem ini membantu dalam mengurangi biaya operasional dalam mengelola Lensa <i>Photography</i> .	✓		
5	Sistem ini membantu meningkatkan akurasi penjadwalan pemesanan.	✓		
6	Antarmuka pengguna (<i>user interface</i>) sistem ini intuitif dan mudah dipahami.	✓		
7	Manajemen pemesanan menjadi lebih terorganisir dengan sistem ini.	✓		
8	Secara keseluruhan, apakah anda puas dengan sistem ini.	✓		

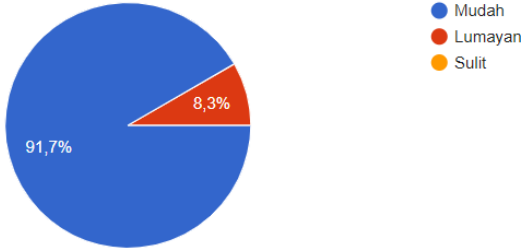

Tabel 3 Tabel wawancara anggota Lensa *Photography*

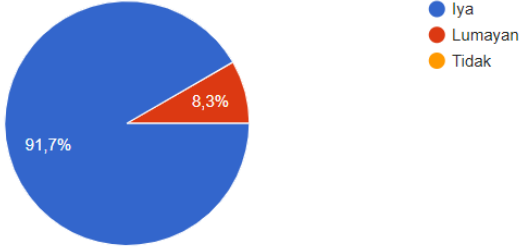


Nama Narasumber : Muhammad Haikal				
Jabatan/Posisi : Anggota				
No	Pertanyaan	Setuju	Kurang Setuju	Tidak Setuju
1	Fitur dan tampilan aplikasi mudah di mengerti	✓		
2	Sistem ini membantu dalam menghemat waktu dalam mengelola jadwal pemesanan.	✓		
3	Sistem ini membantu mengurangi kesalahan penjadwalan.	✓		
4	Sistem ini membantu dalam mengurangi biaya operasional	✓		

**IMPLEMENTASI FRAMEWORK LARAVEL PADA APLIKASI PEMESANAN JASA PHOTOGRAPHY
MENGUNAKAN METODE RAPID APPLICATION DEVELOPMENT**

	dalam mengelola Lensa <i>Photography</i> .			
5	Sistem ini membantu meningkatkan akurasi penjadwalan pemesanan.	✓		
6	Antarmuka pengguna (<i>user interface</i>) sistem ini intuitif dan mudah dipahami.	✓		
7	Manajemen pemesanan menjadi lebih terorganisir dengan sistem ini.	✓		
8	Secara keseluruhan, apakah anda puas dengan sistem ini.	✓		

Tabel 4 Tabel kuesioner pengguna

Kuesioner Pengguna		
No	Pertanyaan	Jawaban
1	Fitur dan tampilan aplikasi mudah di mengerti	<p>12 jawaban</p>  <p>Legend: Mudah (blue), Lumayan (red), Sulit (yellow)</p>
2	Sistem ini membantu menghemat waktu dalam melakukan pemesanan jasa <i>potographer</i> .	<p>12 jawaban</p>  <p>Legend: Iya (blue), Lumayan membantu (red), Tidak membantu (yellow)</p>

3	Apakah sistem ini telah meningkatkan kepuasan anda sebagai pelanggan.	<p>12 jawaban</p>  <p>Legend: Iya (blue), Lumayan (red), Tidak (yellow)</p>
4	Tampilan sistem (Antarmuka Pengguna) mudah dipahami.	<p>12 jawaban</p>  <p>Legend: Iya (blue), Lumayan (red), Tidak (yellow)</p>
5	Secara keseluruhan, apakah anda puas dengan sistem ini.	<p>12 jawaban</p>  <p>Legend: Iya (blue), Lumayan (red), Tidak (yellow)</p>

5. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, sistem pemesanan Lensa *Photography* berbasis *website* berhasil memenuhi tujuan penelitian dengan baik.

- Fitur-fitur fungsional sistem Lensa *Photography* telah berjalan dengan baik melalui pengujian *black box testing*.
- Sistem ini telah diadopsi oleh Lensa *Photography* dan memberikan manfaat termasuk jadwal pemesanan yang lebih rapi dan terorganisir, pengurangan kesalahan penjadwalan, efisiensi waktu dan biaya pemesanan yang lebih baik, serta peningkatan kepuasan pelanggan.
- Hasil dari penerapan metode pengembangan RAD (*Rapid Application Development*) dan penggunaan *framework Laravel* juga sebagai faktor yang berkontribusi terhadap

IMPLEMENTASI FRAMEWORK LARAVEL PADA APLIKASI PEMESANAN JASA PHOTOGRAPHY MENGUNAKAN METODE RAPID APPLICATION DEVELOPMENT

keberhasilan sistem, memungkinkan pengembangan yang cepat, iteratif, dan responsif terhadap kebutuhan pengguna.

- d. Penggunaan *framework Laravel* dalam pengembangan sistem ini telah memberikan berbagai kemudahan, mulai dari proses pembuatan aplikasi yang lebih terstruktur dan efisien, hingga kemampuan integrasi fitur yang kompleks secara lebih sederhana. *Laravel* mendukung pengembangan yang cepat dan fleksibel, memungkinkan penyesuaian yang mudah terhadap kebutuhan pengguna, serta memastikan sistem yang dihasilkan lebih responsif.

Saran

Berdasarkan hasil penelitian ini, dapat disarankan agar Sistem pemesanan ini dapat dikembangkan lebih lanjut dengan menambahkan fitur-fitur baru, seperti fitur pembayaran *online*.

DAFTAR PUSTAKA

- Aasinjery. (2020). Pengantar Teknologi Sistem Informasi. In Aasinjery.
- Afiifah, K., Azzahra, Z. F., & Anggoro, A. D. (2022). Analisis Teknik Entity- Relationship Diagram dalam Perancangan Database Sebuah Literature Review. *Intech*, 3(2), 18–22. <https://doi.org/10.54895/intech.v3i2.1682>
- Arief, M., Ambarsari, N., & Adi, T. N. (2015). membangun aplikasi e-commerce jasa fotografi pre wedding berbasis web crowdsourcing modul fotografer menggunakan metode iterative incremental. *e-Proceeding of Engineering*, (pp. 5782-5790).
- Dedi, Sidik, A., Raya, M., & Ryando, M. (2021). Perancangan Sistem Informasi Promosi Jasa Foto dan Studio Musik Pada M2N Studio Production. *JURNAL SISFOTEK GLOBAL*, 48-52.
- Habibi, R., & Aprilian, R. (2020). Tutorial dan penjelasan aplikasi e-office berbasis web menggunakan metode RAD.
- Huda, B., & Priyatna, B. (2019). enggunaan Aplikasi Content Manajement System (CMS)Untuk Pengembangan Bisnis Berbasis E-Commerce. *SYSTEMATICS*.
- Kendall, K. (2006). Analisis dan Perancangan Sistem. Jakarta: PT.Index. Mandasari, M., & Kaban, R. (2020). perancangan sistem informasi perpustakaan berbasis webdengan metoderapid application development (rad) danframework css bootstrap. *Jurnal Poliprofes*, 2-12.

- M., & Werstantia, N. (2018). perancangan aplikasi pemesanan catering berbasis mobile android. *Jurnal Ilmiah Ilmu dan Teknologi Rekayasa*, 86-95.
- Manullang, A. H., Aritonang, M., & Purba, M. J. (2021). sistem informasi bimbingan belajar number one medan berbasis web. *tamika: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi*, 44-49.
- Nistrina, K., & Sahidah, L. (2022). Unified Modelling Language (Uml) Untuk Perancangan Sistem Informasi Penerimaan Siswa Baru Di Smk Marga Insan Kamil. *Jurnal Sistem Informasi*, 04, 12–23.
- Novendri, M. S., Saputra, A., & Firman, C. E. (2019). aplikasi inventaris barang pada mts nurul islam dumai menggunakan php dan mysql. *l e n t e r a d u m a i*, 46-57.
- Nurjani, Y., & Kurnia Dewi, R. M. (2022). web site sistem pemesanan jasa fotografi berbasis web pada bunglon fotografi. *forech*, 44-49.
- Pasaribu, J. S. (2021). pembuatan aplikasi pemesanan banner diwarna printkota cimahi. *jurnal ilmiah teknologi informasi terapan*, 138-147.
- Prasetiadi, A. (2011). web 3.0: teknologi web masa depan. indept. saputra, a. d., & Borman, R. I. (2020). sistem informasi pelayanan jasa foto berbasis android (studi kasus: ace photography way kanan). *jurnal teknologi dan sistem informasi (jtsi)*, 87-94.
- Satri, & Seabtian, D. T. (2019). sistem informasi e-marketplace pada pemesanan jasa fotografi berbasis web di kotawaringin timur. *jurnal penelitian dosen fikom (unda)*, 1- 8.
- Solichin, A. (2016). pemrograman web dengan php dan mysql. in a. solichin. syani,
- Wistika, K. P., Pramana, D., & Setiasih, N. W. (2023). Sistem Informasi Pemesanan Jasa Fotografi pada Julian Photography Menggunakan Framework Laravel. *Jurnal Informasi dan Teknologi*, 240-249.