

Sistem Informasi Penyuluh Pertanian dan Kelompok Tani di Balai Penyuluh Pertanian Manokwari Barat Menggunakan Framework Laravel 9 dan Bootstrap 4.6

Agricultural Extension and Farmer Group Information System at the West Manokwari Agricultural Extension Center Using the Laravel 9 and Bootstrap 4.6 Frameworks

Natalia Christi Rumbarar¹, Julius Panda Putra Naibaho², Marlinda Sanglise³

^{1,2,3}Teknik Informatika, Universitas Papua, Manokwari, Papua Barat

¹christynatalia67@gmail.com, ²j.naibaho@unipa.ac.id, ³m.sanglise@unipa.ac.id

Info Artikel

Riwayat Artikel:

Diterima 29 Oktober 2025

Direvisi 30 Oktober 2025

Disetujui 31 Oktober 2025

Keywords:

Agricultural extension worker

Information system

Laravel 9

Bootstrap 4.6

MVC

Kata Kunci:

Penyuluh Pertanian

Sistem Informasi

Laravel 9

Bootstrap 4.6

MVC

ABSTRACT

This study discusses the design of a web-based information system utilizing Laravel 9 and Bootstrap 4.6 for the West Manokwari Agricultural Extension Center (BPP). This system aims to optimize information management, increase public access, and accelerate data management. The research method uses a waterfall approach with stages of requirements, design, implementation, verification, and maintenance. The main features developed include managing extension schedules, farmer group data, work area data, and agency profiles. Testing using black-box testing shows that all functions run as required. This system is expected to improve the efficiency, transparency, and accountability of data management within the West Manokwari BPP.

ABSTRAK

Penelitian ini membahas perancangan sistem informasi berbasis web dengan memanfaatkan Laravel 9 dan Bootstrap 4.6 untuk Balai Penyuluh Pertanian (BPP) Manokwari Barat. Sistem ini bertujuan mengoptimalkan pengelolaan informasi, meningkatkan keterbukaan akses bagi masyarakat, dan mempercepat proses pengelolaan data. Metode penelitian menggunakan pendekatan waterfall dengan tahapan requirement, design, implementation, verification, dan maintenance. Fitur utama yang dikembangkan meliputi pengelolaan jadwal penyuluhan, data kelompok tani, data wilayah kerja, serta profil instansi. Pengujian menggunakan black-box testing menunjukkan bahwa seluruh fungsi berjalan sesuai kebutuhan. Sistem ini diharapkan dapat meningkatkan efisiensi, transparansi, dan akuntabilitas pengelolaan data di lingkungan BPP Manokwari Barat.

Koresponden:

Julius P. P. Naibaho

Jurusan Informatika, Fakultas Teknik, Universitas Papua, Manokwari, Papua Barat, Indonesia

Jl. Gunung Salju, Amban, Manokwari, Papua Barat, 98314

Email: j.naibaho@unipa.ac.id

1. PENDAHULUAN

Balai Penyuluh Pertanian adalah salah satu instansi di Kabupaten Manokwari dibawah Dinas Pertanian Ketahanan dan Pangan Kabupaten Manokwari. Sejatinya tugas seorang penyuluh pertanian atau yang lebih sering disebut dengan PPL adalah memberikan penyuluhan pertanian kepada pelaku utama (petani) dan pelaku usaha beserta keluarganya dalam rangka meningkatkan kesejahteraannya. Penyuluhan Pertanian memegang peranan penting dalam meningkatkan kualitas sumber daya petani [1]. Dalam pelaksanaan sistem penyuluhan ada hambatan-hambatan yang mempengaruhi jalannya kegiatan dan keberhasilan penyuluhan [2].

Adapun hambatan-hambatan yang didapat berdasarkan hasil dari observasi dan wawancara yang dilakukan adalah yaitu, tim penyuluh memiliki kesulitan dalam penyebaran informasi kepada petani/masyarakat. Hambatan berikutnya adalah belum adanya website dari instansi terkait yang bisa memberikan informasi mengenai jadwal penyuluhan dan kelompok tani. Selain itu masyarakat membutuhkan media informasi tentang jadwal penyuluhan yang meliputi tempat dan juga waktu pelaksanaan serta judul kegiatan yang akan dilaksanakan [2].

Untuk mengatasi permasalahan tersebut maka perlu di rancang sebuah sistem informasi berbasis web agar dapat membantu pihak Kantor BPP Manokwari Barat dan juga masyarakat, maka dari itu penulis membuat sebuah aplikasi sistem informasi yang berjudul "SIPEKTA". Sipekta merupakan Sistem Informasi Penyuluh Pertanian dan Kelompok Tani berbasis web yang diharapkan dapat memberikan solusi dalam mengatasi hambatan-hambatan yang ada [3].

2. METODE

2.1. Metode Penelitian

Metode yang digunakan adalah metode waterfall yang terdiri dari lima tahap:

1. *Requirement*
Analisis kebutuhan sistem melalui observasi dan wawancara dengan pegawai BPP.
2. *Design*
Perancangan sistem menggunakan use case diagram, perancangan basis data, dan desain antarmuka.
3. *Implementation*
Pengembangan sistem menggunakan Laravel 9 untuk backend dan Bootstrap 4.6 untuk frontend.
4. *Verification*
Pengujian menggunakan black-box testing untuk memastikan fungsionalitas berjalan sesuai spesifikasi.
5. *Maintenance*
Pemeliharaan sistem pasca implementasi.

Alat dan bahan yang digunakan meliputi Processor AMD A4-9120e RADEON R3, 4 COMPUTER CORES 2C+2G 1.50 GHz, bb fVisual Studio Code, XAMPP, MySQL, Composer, dan Git.

2.2. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan penulis dalam menyelesaikan skripsi ini ada 2 metode, yaitu yang pertama adalah metode "Studi Literatur", yang mana pada tahapan ini dilakukan pengumpulan referensi yang diperlukan dalam penelitian. Hal ini dilakukan untuk memperoleh informasi dan data yang di perlukan untuk penulis. Referensi yang digunakan dapat berupa buku, jurnal, paper, makalah, baik berupa media cetak maupun media internet. Selain itu pada tahap ini penulis melakukan pengumpulan data sekunder dari instansi terkait, pada kasus ini adalah Balai Penyuluh Pertanian Kabupaten Manokwari. Selanjutnya metode yang kedua adalah "Observasi dan Wawancara". Metode observasi dilakukan dengan cara melakukan penelitian secara langsung pada lokasi penelitian, sedangkan metode wawancara ditempuh menggunakan pertanyaan-pertanyaan secara lisan dengan pegawai di lokasi penelitian tersebut. Kegiatan observasi dan wawancara dilakukan di lingkungan Balai Penyuluh Pertanian Manokwari Barat, untuk mengetahui permasalahan yang ada.

2.3. Metode Perancangan dan Pembuatan Sistem

Metode perancangan dan pembuatan sistem melalui 4 tahap, yaitu yang pertama adalah tahap "Analisis dan Perancangan". Pada tahap ini penulis mengolah data dari studi literatur dan observasi yang kemudian dilakukan analisis dan perancangan sistem. Kemudian sistem dirancang dengan membuat rancangan use case diagram, skenario use case dan desain. Setelah analisis dan perancangan, penulis kemudian masuk pada tahap kedua yaitu tahap "Implementasi". Pada tahap ini, penulis menerjemahkan data yang telah dirancang kedalam algoritma bahasa pemrograman yang telah ditentukan. Penulis menggunakan Visual Studio Code Editor sebagai text editor dan PHP sebagai bahasa pemrograman dan MYSQL sebagai database. Serta implementasi desain antarmuka hasil dari sistem yang dirancang. Kemudian pada tahap berikutnya yaitu "Pengujian Sistem". Setelah sistem dibangun, maka dilakukan proses pengujian website untuk menguji atau mengetahui fungsional dari aplikasi yang telah dibangun menggunakan metode black box. Kemudian, tahap terakhir yang dilakukan adalah "Dokumentasi", yang mana pada tahap ini berisi laporan dan kesimpulan akhir dari hasil akhir analisa dan perancangan sistem.

3. HASIL DAN PEMBAHASAN

3.1. Tahap Perancangan dan Pembuatan Sistem

Melakukan kegiatan menganalisa kebutuhan fungsionalitas sistem informasi yang akan dikembangkan. Analisa kebutuhan fungsionalitas sistem informasi berdasarkan dari hasil observasi dan wawancara. Berikut kebutuhan fungsionalitas yang berhasil diperoleh:

1. Sistem informasi dapat memberikan informasi jadwal penyuluhan, baik tempat pelaksanaan, waktu pelaksanaan, petugas penyuluhan, dan juga judul kegiatan.
2. Memberikan informasi daftar kelompok tani.
3. Memberikan informasi wilayah kerja.
4. Sistem informasi memiliki fitur untuk membuat jadwal penyuluhan, mengolah data pegawai, mengolah data kelompok tani, dan mengolah data wilayah kerja, serta mengolah data profil website dan juga profil instansi terkait.

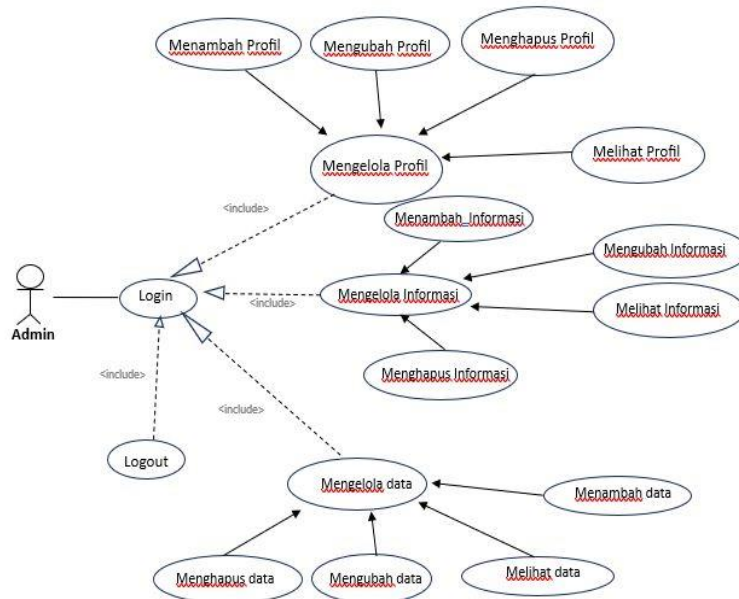
3.1.1. Use Case Diagram

Use case diagram merupakan salah satu implementasi dari Unified Modelling Language (UML) yang berguna untuk memberi gambaran hubungan antara pengguna dengan sistem, serta memberikan hak akses dari pengguna dalam menggunakan sistem tersebut. Penulis menggunakan use case Diagram untuk menggambarkan interaksi antara aktor dengan sistem yang dibuat. Dalam hal ini penulis bermaksud untuk memberikan pemahaman tentang fungsi apa saja yang terdapat didalam sistem dan siapa saja yang dapat menggunakan fungsi-fungsi tersebut. Use case diagram disini sendiri dibuat berdasarkan permasalahan dan deskripsi yang sudah diperoleh. Pada sistem informasi unit Kantor Penyuluh Pertanian berbasis web ini terdapat 2 aktor, yaitu admin dan pengunjung. Deskripsi aktor tersebut dapat dilihat pada tabel 3.1.

Tabel 3.1. Deskripsi Aktor

Aktor	Deskripsi
Admin	Pegawai Kantor Penyuluh Pertanian bertindak sebagai admin yang mengendalikan sistem. Admin harus login terlebih dahulu agar dapat mengelola data pada sistem ini. Pengelolaan admin mencakup pengelolaan data informasi jadwal penyuluhan, kelompok tani, data pegawai serta data lainnya terkait instansi dan juga kegiatan penyuluhan.
Pengunjung	Masyarakat/petani bertindak sebagai pengunjung pada sistem ini. Pengunjung dapat melihat informasi jadwal penyuluhan, kelompok tani, petugas penyuluhan, maupun info penting yang diberitakan dari pihak Balai Penyuluh Pertanian tanpa melakukan login terlebih dahulu.

Ditambahkan login dan logout sebagai proses autentifikasi hak akses admin untuk keamanan pengelolaan data pada sistem. Gambar use case diagram admin dapat dilihat pada Gambar 3.1 dan deskripsi use case admin dapat dilihat pada tabel 3.2.



Gambar 3.2. Use Case Diagram Admin

Tabel 3.2 Deskripsi *Use Case Admin*

No	Use Case	Fungsi
1	Login	Use case ini berfungsi untuk masuk sebagai admin pada sistem informasi.
2	Logout	Use case ini berfungsi untuk keluar sebagai admin pada sistem informasi.
3	Mengelola Informasi	Use case ini berfungsi untuk mengelola data informasi yang meliputi membuat, melihat, mengubah, dan menghapus data.
4	Mengelola Data	Use case ini berfungsi untuk mengelola data yang meliputi membuat, melihat, mengubah, dan menghapus data.
5	Mengelola Profil	Use case ini berfungsi untuk mengelola data profil yang meliputi membuat, melihat, mengubah, dan menghapus data.

Sedangkan untuk pengunjung dapat melihat informasi dari Balai Penyuluh Pertanian tanpa harus melakukan proses autentifikasi terlebih dahulu. Deskripsi *use case* pengunjung dapat dilihat pada tabel 3.3.

Tabel 3.3. Deskripsi *Use Case Pengunjung*

No	Use Case	Fungsi
1	Melihat Informasi	Use case ini berfungsi untuk menampilkan informasi terkait jadwal penyuluhan, baik tempat, waktu dan juga petugas penyuluhan. Serta informasi kelompok tani, dan juga informasi lainnya terkait pertanian yang di posting oleh admin/pengelola.
2	Melihat Profil	Use case ini berfungsi untuk menampilkan data profil.

Use case memiliki skenario yang menunjukkan alur kerja antara sistem dan aktor beserta alur alternatif jika skenario tidak berjalan sesuai alur normal. Salah satu skenario *use case* dapat dilihat pada tabel 3.4 berikut:

Tabel 3.4. Skenario *Use Case Login*

Aksi Aktor	Reaksi Sistem
	<i>Main Flow</i>
1. Membuka halaman Login.	2. Menampilkan halaman login.
3. Memasukan username dan password.	4. Mencocokkan username dan password dengan database admin.
<i>Exception</i>	4a. Jika username dan password tidak sesuai maka akan keluar peringatan username atau password salah.
<i>Post Conditions</i>	Aktor masuk sebagai admin.

Daftar skenario *use case* dapat dilihat pada tabel 3.5.

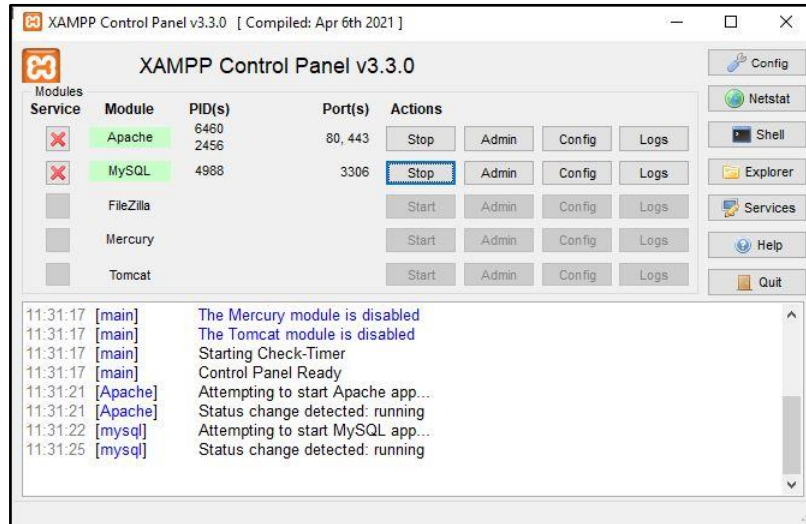
Tabel 3.5. Daftar *Use Case*

No	Nama Skenario <i>Use Case</i>
1	Memasukkan data
2	Melihat data
3	Mengubah data
4	Menghapus data
5	Logout

3.1.2. Cara Menjalankan Website

1. Jalankan Xampp

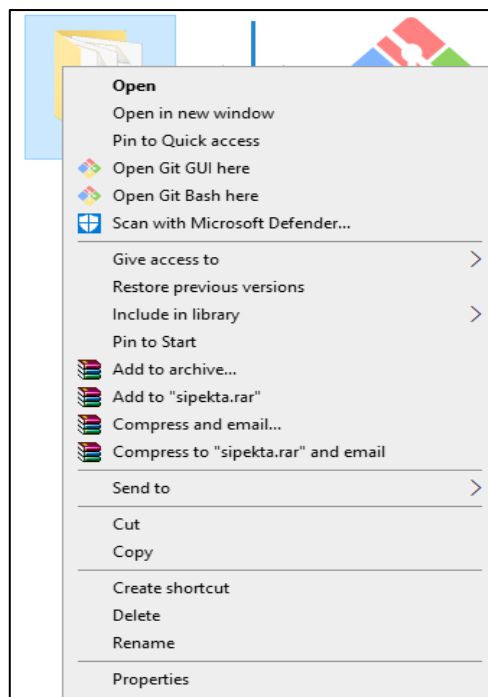
Untuk tahap pertama menjalankan website, terutama pastikan Xampp sudah dijalankan. Seperti pada gambar 3.2.



Gambar 3.2. Jalankan Xampp

2. Open Git Bash Here

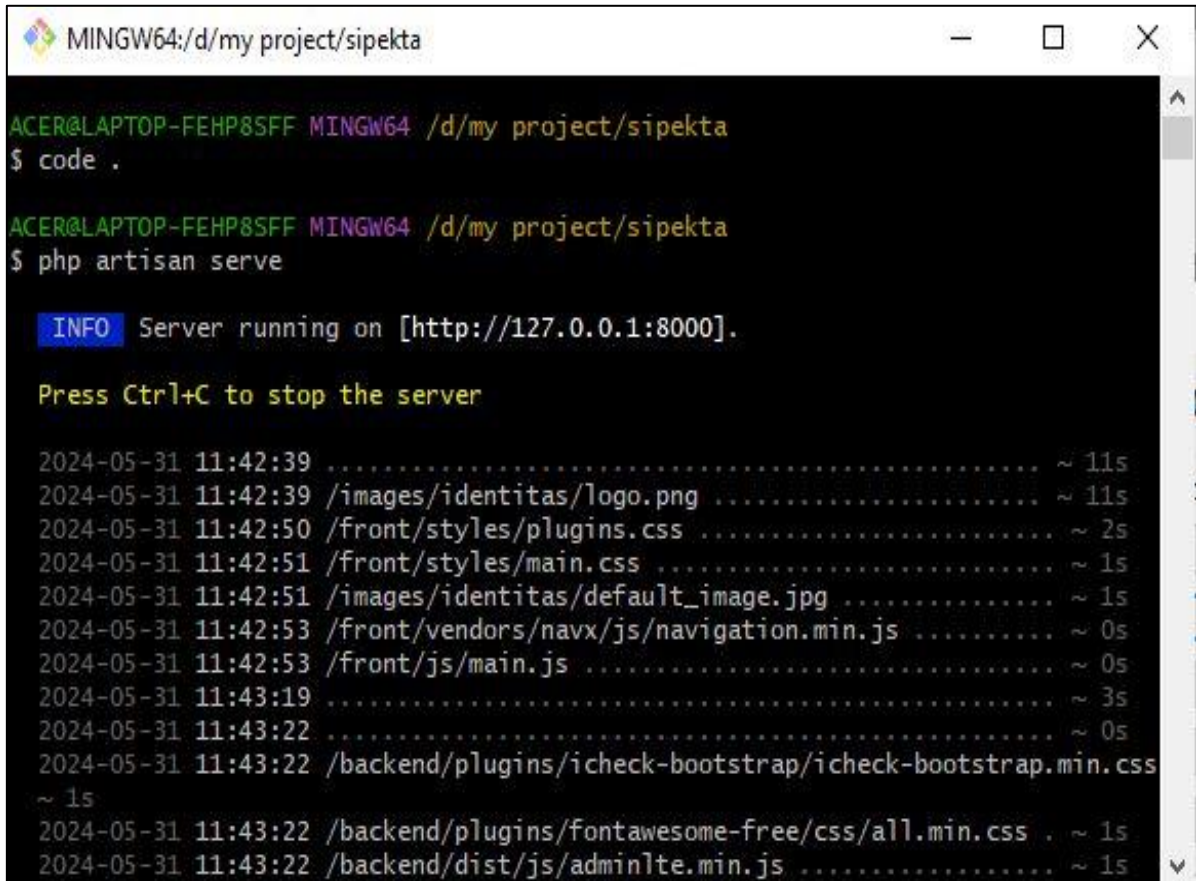
Setelah Xampp dipastikan sudah berjalan, kemudian klik kanan folder pada data yang akan dibuka, kemudian klik "Open Git Bash Here", seperti terlihat pada gambar 3.3.



Gambar 3.3. Membuka Folder Pada Git

3. Masukkan kode Pada CMD

- 1) Masukkan kode. pada cmd lalu enter.
- 2) Setelah itu masukkan php artisan serve lalu tekan enter.



```

MINGW64:/d/my project/sipekta
ACER@LAPTOP-FEHP8SFF MINGW64 /d/my project/sipekta
$ code .
ACER@LAPTOP-FEHP8SFF MINGW64 /d/my project/sipekta
$ php artisan serve

[INFO] Server running on [http://127.0.0.1:8000].

Press Ctrl+C to stop the server

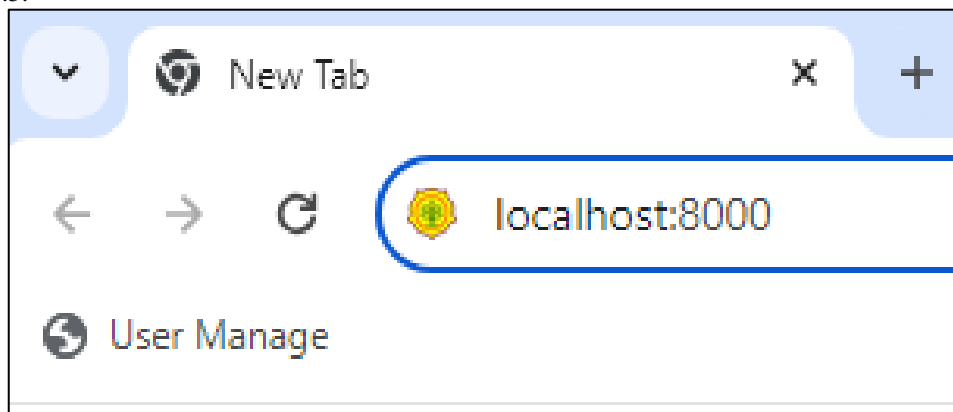
2024-05-31 11:42:39 ..... ~ 11s
2024-05-31 11:42:39 /images/identitas/logo.png ..... ~ 11s
2024-05-31 11:42:50 /front/styles/plugins.css ..... ~ 2s
2024-05-31 11:42:51 /front/styles/main.css ..... ~ 1s
2024-05-31 11:42:51 /images/identitas/default_image.jpg ..... ~ 1s
2024-05-31 11:42:53 /front/vendors/navx/js/navigation.min.js ..... ~ 0s
2024-05-31 11:42:53 /front/js/main.js ..... ~ 0s
2024-05-31 11:43:19 ..... ~ 3s
2024-05-31 11:43:22 ..... ~ 0s
2024-05-31 11:43:22 /backend/plugins/ichack-bootstrap/ichack-bootstrap.min.css
~ 1s
2024-05-31 11:43:22 /backend/plugins/fontawesome-free/css/all.min.css . ~ 1s
2024-05-31 11:43:22 /backend/dist/js/adminlte.min.js ..... ~ 1s

```

Gambar 3.4. Masukkan Kode Pada CMD

4. Membuka *website*

Setelah langkah-langkah sebelumnya sudah dilakukan dengan benar, kemudian menuju browser untuk membuka *website*. Ketik "localhost:8000" pada kolom pencarian di browser lalu tekan enter. Seperti pada gambar 3.5.

Gambar 3.5. Membuka *Website*

3.2. Implementasi Program

Proses implementasi pembuatan program pada SIPEKTA berbasis web ini menggunakan framework laravel 9, maka konstruksi program dibagi menjadi 4 bagian yaitu routes, model, view, dan controller. Berikut ini adalah implementasi program pada SIPEKTA berbasis web.

1. Model

Model berisi kumpulan fungsi-fungsi kode program yang berhubungan dengan database. Contoh potongan kode program model dapat dilihat pada gambar 3.6.

```
namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class JadwalKegiatan extends Model
{
    use HasFactory;

    protected $table = 'jadwal_kegiatan';
    protected $fillable = [
        'kota_id',
        'distrik_id',
        'kampung_id',
        'kantor_id',
        'pegawai_kegiatan',
        'judul_kegiatan',
        'hari_tgl',
    ];

    protected $hidden = [];

    public function kota()
    {
        return $this->belongsTo('App\Models\Kota', 'kota_id');
    }

    public function kampung()
    {
        return $this->belongsTo('App\Models\Kampung', 'kampung_id');
    }
}
```

Gambar 3.6. Potongan Kode Program Model

2. Controller

Controller berisi kumpulan fungsi-fungsi kode program untuk mengolah logika dan data yang akan ditampilkan pada view. Contoh potongan kode program controller dapat dilihat pada gambar 3.7.

```
namespace App\Http\Controllers;

use App\Models\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use RealRashid\SweetAlert\Facades\Alert;

class AdminController extends Controller
{
    public function dashboard()
    {
        return view('pages.admin.dashboard');
    }

    public function accountSetting()
    {
        $id = Auth::user()->id;
        $user = User::find($id);
        return view('pages.admin.users.user_profile', compact('user'));
    }

    public function editProfile()
    {
        $id = Auth::user()->id;
        $user = User::find($id);
        return view('pages.admin.users.edit_profile', compact('user'));
    }

    public function updateProfile(Request $request, $id)
    {
    }
}
```

Gambar 3.7. Potongan Kode Pogram Controller

3. View

View berisi kumpulan fungsi-fungsi kode program untuk menunjukkan tampilan kepada pengguna. Contoh potongan kode program *view* dapat dilihat pada gambar 3.8.

```
@section('title')
    Jadwal Sosialisasi {{ $about->nama_kantor }}
@endsection

@section('content')
    <section class="rlr-about-hero">
        <div class="rlr-about-hero_content">
            
            <div class="rlr-about-hero_content-inner">
                <h1 class="type-d1 rlr-about-hero_title">{{ $header->nama_header_title }} {{ $about->nama_kantor }}</h1>
            </div>
        </div>
    </section>

    <aside class="row rlr-search-results-page_dynamic-filter-section">
        <!-- Swiper -->
        <div class="swiper rlr-dynamic-filter-swiper rlr-js-dynamic-filter-swiper">
            <ul class="swiper-wrapper">
                @foreach ($kantor as $k)
                    <li class="rlr-search-results-page_dynamic-filter-item swiper-slide">
                        <a href="{{ route('jadwal.front.bykantor', $k->id) }}"
                            class="rlr-icon-text btn rlr-icon-text_card rlr-icon-text--anchor" <i
                                class="rlr-icon-font fa fa-building"> </i> <span
                                    class="rlr-icon-text_card-title">{{ $k->nama_kantor }} </span> </a>
                    </li>
                @endforeach
            </ul>
        </div>
    </aside>
</body>
</html>
```

Gambar 3.8. Potongan Kode Pogram *View*

4. Routes

Pembuatan *routes* bertujuan untuk memetakan alamat website ke controller yang sesuai. Contoh potongan kode program *routes* dapat dilihat pada gambar 3.9.

```
Route::get('/', [HomeController::class, 'index']->name(''));
// TENTANG
Route::get('tentang', [FrontTentangController::class, 'index']->name('tentang.front.index'));
// KANTOR
Route::get('daftar-kantor', [FrontKantorController::class, 'index']->name('kantor.front.index'));
Route::get('detail-kantor/{nama_kantor}', [FrontKantorController::class, 'detail']->name('kantor.front.detail'));
// BLOG
Route::get('blog', [FrontBlogController::class, 'index']->name('blog.front.index'));
Route::get('blog/{slug}', [FrontBlogController::class, 'detail']->name('blog.front.detail'));
// PostByCategory
Route::get('category/{category_name}', [FrontCategoryController::class, 'index']->name('category.front.index'));
// Jadwal
Route::get('jadwal-sosialisasi', [FrontJadwalController::class, 'index']->name('jadwal.front.index'));
Route::get('jadwal-sosialisasi/{id}', [FrontJadwalController::class, 'jadwalbykantor']->name('jadwal.front.bykantor'));
// Poktan
Route::get('poktan', [FrontPoktanController::class, 'index']->name('poktan.front.index'));

Route::middleware([
    'auth:sanctum',
    config('jetstream.auth_session'),
    'verified'
])->group(function () {
    // Admin Route
    Route::get('dashboard', [AdminController::class, 'dashboard']->name('dashboard'));
});
```

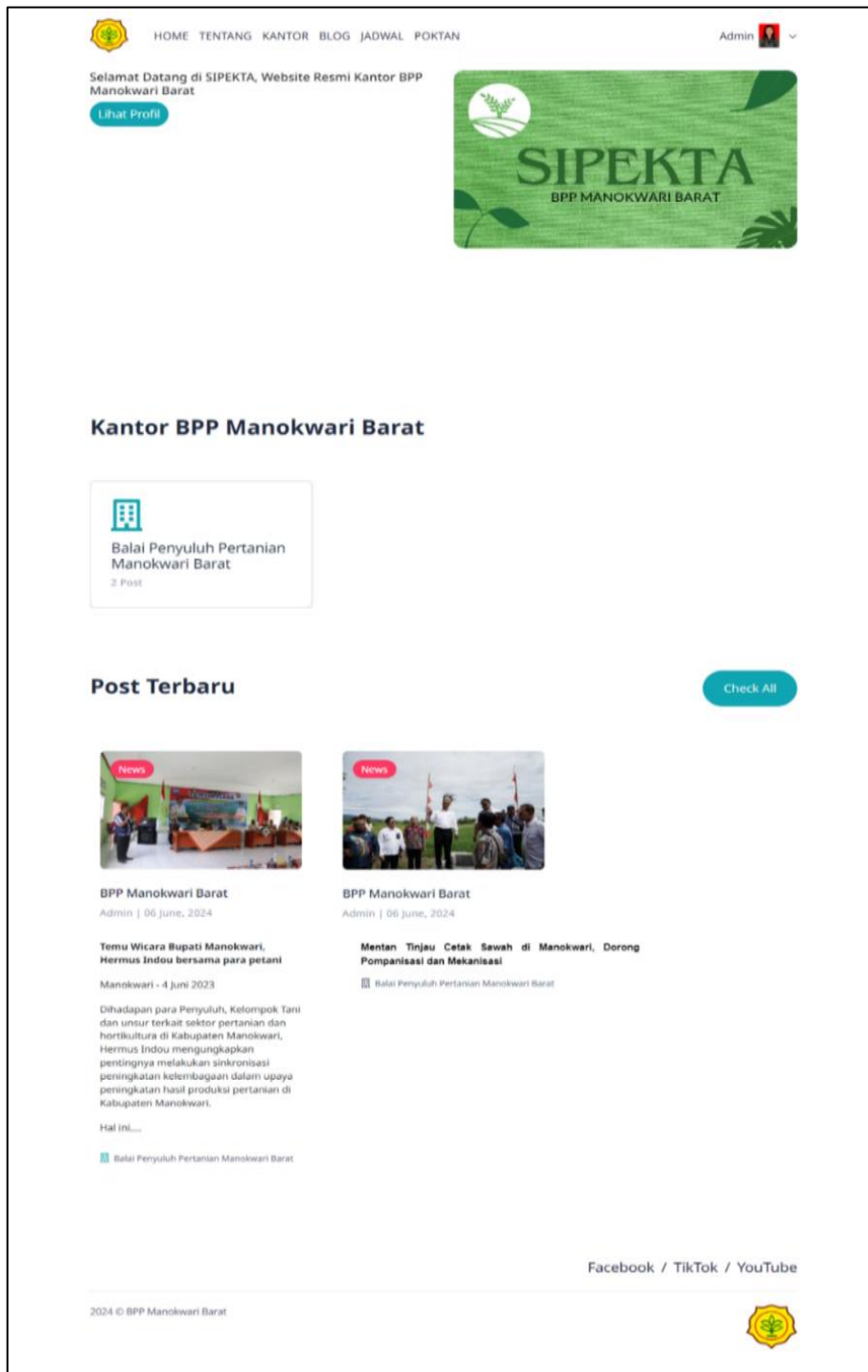
Gambar 3.9. Potongan Kode Pogram *Routes*

3.3. Implementasi Desain Antar Muka

Pada kegiatan ini dilakukan sebatas penulisan kode program berupa fungsi dan tampilan. Implementasi desain antar muka halaman admin atau halaman dashboard dapat dilihat pada Lampiran. Sedangkan Implementasi desain antar muka halaman pengunjung dapat dilihat pada gambar-gambar berikut:

1. Home

Menu *home* menampilkan bagian paling awal dari aplikasi yang telah dibuat. Pada bagian ini, memuat ucapan selamat datang bagi pengunjung. Selain itu, juga terdapat informasi atau berita-berita terupdate terkait pertanian.



Gambar 3.10. Hasil Implementasi Tampilan Menu Home Pada Halaman Depan

2. Tentang

Pada bagian ini, memuat penjelasan singkat terkait apa itu SIPEKTA.



Gambar 3.11. Hasil Implementasi Tampilan Menu Tentang Pada Halaman Depan

3. Kantor

Menu kantor sendiri memuat tentang profil kantor BPP Manokwari Barat, penjelasan singkat mengenai kantor tersebut dan juga alamat serta kontak beserta akun media sosial dari kantor tersebut.



Gambar 3.12. Hasil Implementasi Tampilan Menu Kantor Pada Halaman Depan

4. Sub Menu Kantor

Sub Menu Kantor merupakan bagian dari Menu Kantor yang memuat penjelasan singkat terkait kantor BPP Manokwari Barat beserta alamat lokasi kantor, seperti yang sudah dijelaskan pada menu Kantor diatas.

HOME TENTANG KANTOR BLOG JADWAL POKTAN Admin

HOME > Daftar Kantor > Balai Penyuluh Pertanian Manokwari Barat

Balai Penyuluh Pertanian Manokwari Barat

—BALAI PENYULUH PERTANIAN, MANOKWARI BARAT

Balai Penyuluhan Pertanian, atau yang disingkat BPP adalah lembaga yang mempunyai peran, tugas dan fungsi pada kegiatan Penyuluhan Pertanian pada tingkat Distrik, serta merupakan unit kerja dengan wilayah kerja satu atau beberapa Distrik.

Balai Penyuluh Pertanian (BPP) Manokwari Barat sendiri merupakan salah satu dari 9 lembaga BPP yang berada di Kabupaten Manokwari. BPP Manokwari Barat berkedudukan di Distrik Manokwari Barat, dengan wilayah kerja Distrik Manokwari Barat.

admin@gmail.com Jl. Angkasa Mulyono No.6, Amban, Kec. Manokwari Barat, Kab. Manokwari, Papua Barat. 98312

Lokasi Kantor
-- 0°50'32"S 134°03'25"E

Post From : Balai Penyuluh Pertanian Manokwari Barat

Check All

BPP Manokwari Barat
Admin | 06 June, 2024

Temu Wicara Bupati Manokwari, Hermus Indou bersama para petani
Manokwari - 4 Juni 2023

Dihadapan para Penyuluh, Kelompok Tani dan unsur terkait sektor pertanian dan hortikultura di Kabupaten Manokwari, Hermus Indou mengungkapkan pentingnya melakukan sinkronisasi peningkatan kelembagaan dalam upaya peningkatan hasil produksi pertanian di Kabupaten Manokwari.

Hal ini...

Balai Penyuluh Pertanian Manokwari Barat

BPP Manokwari Barat
Admin | 06 June, 2024

Mentan Tinjau Cetak Sawah di Manokwari, Dorong Pompanisasi dan Mekanisasi
Balai Penyuluh Pertanian Manokwari Barat

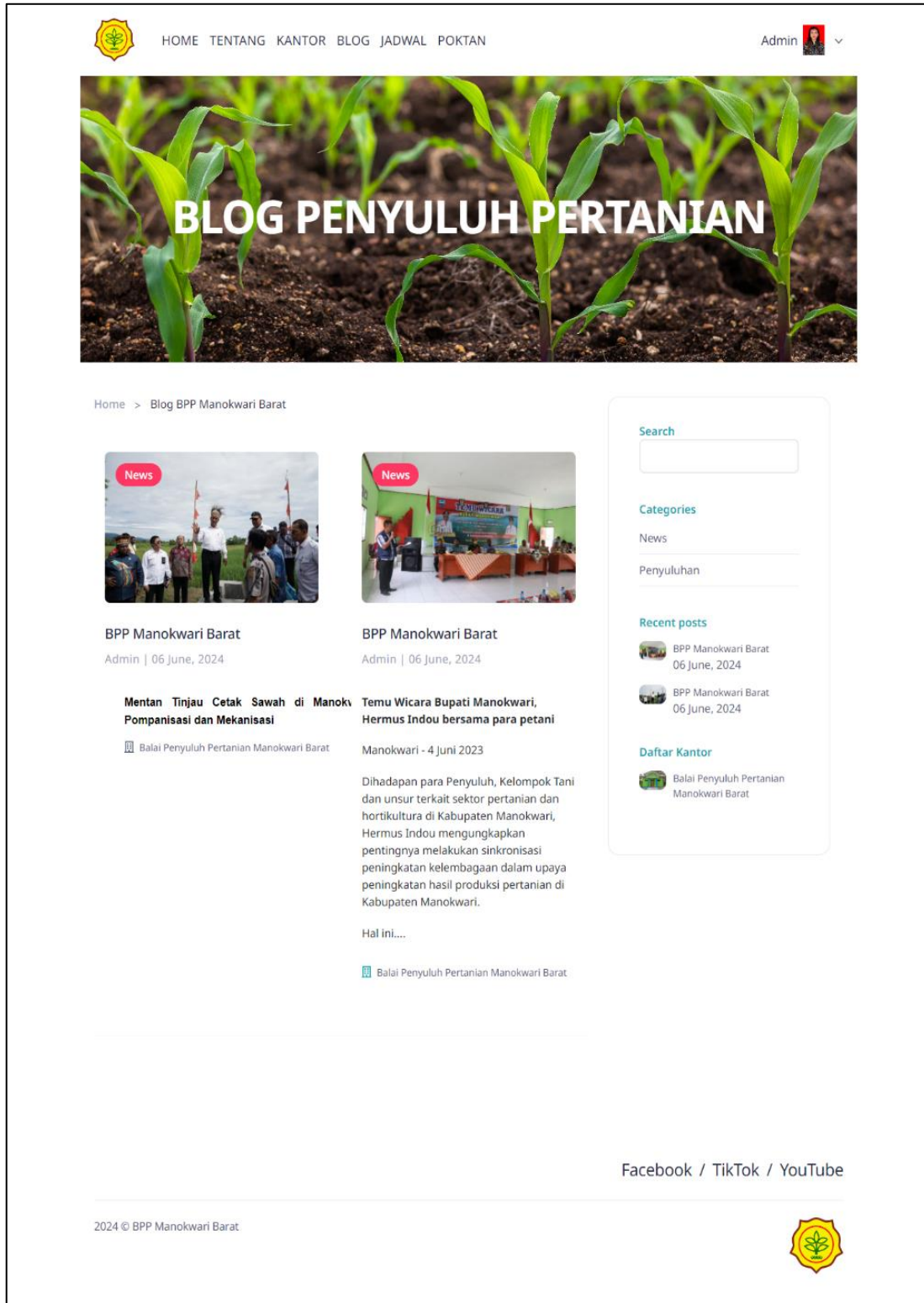
Facebook / TikTok / YouTube

2024 © BPP Manokwari Barat

Gambar 3.13. Hasil Implementasi Tampilan Sub Menu Kantor Pada Halaman Depan

5. Blog

Blog memuat halaman berita maupun informasi terkait pertanian, baik yang terupdate sampai berita-berita yang terdahulu atau yang sudah tidak ditampilkan di halaman lain karena merupakan berita lama, pengunjung masih bisa menemukan berita-berita tersebut pada menu Blog ini.



Gambar 3.14. Hasil Implementasi Tampilan Menu Blog Pada Halaman Depan

6. Jadwal Penyuluhan

Pada menu ini, masyarakat/petani dalam lingkup wilayah kerja BPP Manokwari Barat dapat melihat atau menemukan informasi jadwal penyuluhan yang akan dilaksanakan oleh tim penyuluh, baik tempat pelaksanaan, waktu dan juga petugas yang akan melaksanakan penyuluhan.

HOME TENTANG KANTOR BLOG JADWAL POKTAN Admin

JADWAL PENYULUHAN

Balai Penyuluh Pertanian Manokwari Barat

Jadwal Sosialisasi BPP Manokwari Barat

Show 10 entries Search:

Judul Kegiatan	Hari / Tgl	Kota	Distrik	Kampung	Kantor
Menanam Kacang Panjang / Cara Budidaya Tanaman Kacang Panjang	12 Jun 2024	Manokwari	Manokwari barat	Amban	Balai Penyuluh Pertanian Manokwari Barat
Penyuluhan	06 Jun 2024	Manokwari	Manokwari barat	Inggramui	Balai Penyuluh Pertanian Manokwari Barat

Showing 1 to 2 of 2 entries Previous 1 Next

Facebook / TikTok / YouTube

2024 © BPP Manokwari Barat

Gambar 3.15. Hasil Implementasi Tampilan Menu Jadwal Pada Halaman Depan

7. Kelompok Tani

Menu Kelompok Tani memuat data kelompok tani dalam lingkup wilayah kerja BPP Manokwari Barat, yang terdiri daftar nama kelompok tani, ketua, anggota, jumlah anggota, beserta nama kampung.



Gambar 3.16. Hasil Implementasi Tampilan Menu Kelompok Tani Pada Halaman Depan

3.4. Pengujian Aplikasi

Pengujian yang dilakukan pada aplikasi ini adalah metode pengujian Black box testing, yaitu pengujian yang berfokus pada persyaratan fungsional perangkat lunak untuk melihat apakah program aplikasi menghasilkan output yang diinginkan dan sesuai dengan fungsi dari program aplikasi yang dibuat tersebut, pengujian fungsional dilakukan oleh pengembang. Berikut adalah hasil pengujian sistem aplikasi, yang dapat dilihat pada tabel 3.6.

Tabel 3.6. Hasil Pengujian Aplikasi (*Black Box Testing*)

Daftar pengujian	Prosedur yang dijalankan	Hasil yang diharapkan	Hasil
<i>Pengunjung</i>	Pengunjung mengakses alamat <i>web</i> Sistem Informasi Penyuluh Pertanian dan Kelompok Tani BPP Manokwari Barat	Pengunjung masuk ke halaman <i>web</i> Sistem Informasi Penyuluh Pertanian dan Kelompok Tani BPP Manokwari Barat	Berhasil
<i>Login Admin</i>	<i>Admin</i> memasukkan <i>username</i> dan <i>password</i>	<i>Admin</i> berhasil masuk kehalaman <i>dashboard</i>	Berhasil
<i>Admin</i> menambahkan data judul <i>header</i> pada setiap menu	<i>Admin</i> pilih menu <i>header tittle</i> dan menambahkan data pada setiap menu yang akan tertampil di halaman depan.	<i>Admin</i> berhasil menambahkan data pada setiap menu yang akan tertampil di halaman depan.	Berhasil
<i>Admin</i> menambahkan data identitas <i>website</i>	<i>Admin</i> memilih halaman identitas <i>website</i> .	<i>Admin</i> berhasil menambahkan data identitas <i>website</i> .	Berhasil
<i>Admin</i> menambahkan data profil kantor	<i>Admin</i> memilih halaman profil kantor lalu menambahkan data profil kantor.	<i>Admin</i> berhasil menambahkan data profil kantor dan berhasil ditampilkan ke halaman depan.	Berhasil
<i>Admin</i> menambahkan data profil pegawai	<i>Admin</i> memilih halaman profil pegawai lalu menambahkan data profil tiap pegawai.	<i>Admin</i> berhasil menambahkan data profil pegawai.	Berhasil
<i>Admin</i> membuat kategori postingan	<i>Admin</i> memilih halaman kategori dan kemudian klik tambah data untuk membuat kategori postingan pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil menambahkan data kateori postingan.	Berhasil
<i>Admin</i> mengubah data kategori postingan	<i>Admin</i> pilih halaman kategori, lalu klik edit, lalu akan muncul halaman edit kategori, lalu <i>admin</i> mengubah data kategori.	<i>Admin</i> berhasil mengubah data kategori.	Berhasil
<i>Admin</i> menghapus data kategori postingan	<i>Admin</i> memilih halaman kategori, lalu klik tombol hapus hapus, maka akan muncul jendela pesan hapus kategori, lalu klik tombol hapus.	<i>Admin</i> berhasil menghapus data kategori.	Berhasil
<i>Admin</i> membuat postingan/berita yang akan muncul di halaman depan	<i>Admin</i> memilih halaman buat postingan, lalu pilih tambah data, kemudian <i>admin</i> membuat postingan/berita pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil membuat data postingan/berita, lalu ketika <i>admin</i> berhasil maka postingan akan muncul di halaman depan dan secara otomatis akan masuk ke halaman daftar postingan	Berhasil
<i>Admin</i> mengubah data postingan/berita	<i>Admin</i> memilih halaman daftar postingan, lalu klik tombol edit, kemudian <i>admin</i> mengubah postingan/berita pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil mengubah data postingan/berita, lalu ketika <i>admin</i> berhasil maka postingan akan terupdate di halaman depan dan juga pada daftar postingan.	Berhasil
<i>Admin</i> menghapus data postingan/ berita	<i>Admin</i> memilih halaman daftar postingan, lalu klik tombol hapus pada postingan/berita yang hendak dihapus.	<i>Admin</i> berhasil menghapus data postingan/berita, lalu ketika <i>admin</i> berhasil maka akan terupdate di halaman depan dan juga pada daftar postingan.	Berhasil
<i>Admin</i> menambahkan data Kota, Distrik dan Kampung	<i>Admin</i> memilih halaman Kota, Distrik dan Kampung, kemudian klik tambah data untuk memasukan data Kota, Distrik dan Kampung	<i>Admin</i> berhasil menambahkan data Kota, Distrik dan Kampung.	Berhasil
<i>Admin</i> membuat agenda/jadwal penyuluhan yang akan muncul di halaman depan	<i>Admin</i> memilih halaman jadwal penyuluhan, kemudian klik tambah data, lalu <i>admin</i> membuat agenda/jadwal penyuluhan pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil membuat agenda/jadwal penyuluhan, lalu ketika <i>admin</i> berhasil maka jadwal penyuluhan akan tertampil di halaman depan dan juga pada halaman daftar jadwal kegiatan.	Berhasil
<i>Admin</i> mengubah data agenda/ jadwal penyuluhan	<i>Admin</i> memilih halaman daftar jadwal kegiatan, lalu klik tombol edit, kemudian <i>admin</i> mengubah agenda pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil mengubah data jadwal penyuluhan, lalu ketika <i>admin</i> berhasil maka perubahan jadwal penyuluhan akan terupdate di halaman depan dan juga pada halaman daftar jadwal kegiatan.	Berhasil
<i>Admin</i> menambahkan data kelompok tani yang akan muncul di halaman depan	<i>Admin</i> memilih halaman kelompok tani, kemudian klik tambah data, lalu <i>admin</i> menambahkan data kelompok tani pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil menambahkan data kelompok tani, lalu ketika <i>admin</i> berhasil maka data kelompok tani akan tertampil di halaman depan dan juga pada halaman daftar kelompok tani.	Berhasil

Daftar pengujian	Prosedur yang dijalankan	Hasil yang diharapkan	Hasil
<i>Admin</i> mengubah data kelompok tani	<i>Admin</i> memilih halaman daftar kelompok tani, lalu klik tombol edit, kemudian <i>admin</i> mengubah data kelompok tani pada form yang tersedia, setelah itu klik submit.	<i>Admin</i> berhasil mengubah data kelompok tani, lalu ketika <i>admin</i> berhasil maka perubahan data kelompok tani akan terupdate di halaman depan dan juga pada halaman daftar kelompok tani.	Berhasil
<i>Admin</i> menghapus data agenda/ jadwal penyuluhan	<i>Admin</i> memilih halaman data kelompok tani, lalu klik tombol hapus pada kelompok tani yang hendak dihapus.	<i>Admin</i> berhasil menghapus data kelompok tani, lalu ketika <i>admin</i> berhasil maka akan terupdate di halaman depan dan juga pada halaman daftar kelompok tani.	Berhasil
<i>Admin sign out</i>	<i>Admin</i> memilih <i>button sign out</i> .	<i>Admin</i> berhasil keluar dari <i>dashboard</i> . Ketika <i>admin</i> berhasil keluar dari halaman <i>dashboard</i> maka akan kembali kehalaman depan.	Berhasil

4. KESIMPULAN

Berdasarkan hasil penelitian ini, maka dapat ditarik kesimpulan sebagai berikut:

1. Perancangan Sistem informasi Penyuluh Pertanian di Kantor Balai Penyuluh Pertanian Manokwari Barat dibuat menggunakan *framework laravel 9* dan *bootstrap 4.6*.
2. Sistem Informasi yang dibuat dapat memudahkan masyarakat dalam lingkup wilayah kerja BPP Manokwari Barat untuk mendapatkan informasi mengenai jadwal penyuluhan, kelompok tani dan informasi-informasi lain terkait pertanian yang diberikan oleh pihak Kantor BPP Manokwari Barat.
3. Pengujian tampilan dan fungsional aplikasi menggunakan metode *black box testing*.
4. Hasil pengujian tampilan dan fungsional aplikasi menggunakan metode *black box testing* dinyatakan berhasil selurunya dan aplikasi siap digunakan.

REFERENSI

- [1] Pemerintah Kabupaten Manokwari. (2022). Peraturan Bupati Manokwari Nomor 13 Tahun 2022 tentang Pembentukan Kelembagaan Fungsional Penyuluhan Pertanian Kabupaten Manokwari. Manokwari: Pemerintah Kabupaten Manokwari.
- [2] Alberki, R. (2020). Situs Web Perusahaan PT. Rahnza Menggunakan Framework Codeigniter 4.0.4 Dan Bootstrap 4.6 (Studi Kasus: PT. Rahnza Papua Barat). *Skripsi*. Universitas Papua.
- [3] Aprianto, S. (2019). Penerapan Framework Laravel Pada Sistem Informasi Pengesahan Perseroan Terbatas Di Notaris. *Skripsi*. STMIK Akakom Yogyakarta.
- [4] Dinullah, A. S. (2022). Payment Gateway Pada Aplikasi Pemesanan E-Travel (Studi Kasus WOI Travel Jogja Group). *Skripsi*. Universitas Teknologi Digital Indonesia.