

**PERANCANGAN DAN IMPLEMENTASI WEBSITE ULASAN DAN  
INFORMASI PESANTREN BERBASIS LARAVEL MENGGUNAKAN  
METODE AGILE**

**TUGAS AKHIR**

**Diajukan Oleh:**

**Faiz Ramadhana Surya**

**210705037**

**Mahasiswa Fakultas Sains dan Teknologi  
Program Studi Teknologi Informasi**



**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI AR-RANIRY  
BANDA ACEH  
2025/2026**

## LEMBAR PENGESAHAN

### PERANCANGAN DAN IMPLEMENTASI *WEBSITE* ULASAN DAN INFORMASI PESANTREN BERBASIS LARAVEL MENGGUNAKAN METODE AGILE

#### TUGAS AKHIR


Telah Diuji Oleh Panitia Ujian Munaqasyah Tugas Akhir  
Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh dan Dinyatakan Lulus  
Serta Diterima Sebagai Salah Satu Beban Studi Program Sarjana (S-1)  
Dalam Program Studi Teknologi Informasi

Pada Hari/Tanggal: Senin, 2 Februari 2026  
14 Sya'ban 1447 H


Di Darussalam, Banda Aceh

Panitia Ujian Munaqasyah Tugas Akhir :

Ketua,

  
Mulkan Fadhli, M.T.  
NIP. 198811282020121006

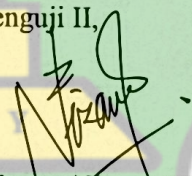
Sekretaris,

  
Dr. Hendri Ahmadian, S.Si., M.I.M.  
NIP. 198301042014031002

Penguji I,

  
Khairan AR, M.Kom  
NIP. 198607042014031001

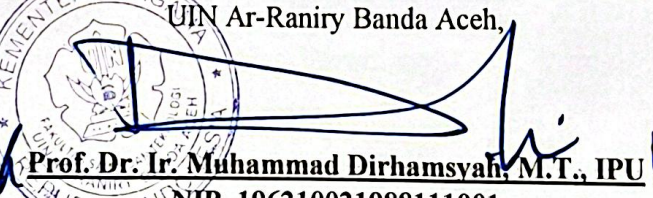
Penguji II,

  
Nizam Albar, ST., M.T

Mengetahui:

Dekan Fakultas Sains dan Teknologi  
UIN Ar-Raniry Banda Aceh,



  
Prof. Dr. Ir. Muhammad Dirhamsyah, M.T., IPU  
NIP. 196210021988111001

**LEMBAR PERSETUJUAN**

**PERANCANGAN DAN IMPLEMENTASI *WEBSITE* ULASAN DAN INFORMASI  
PESANTREN BERBASIS LARAVEL MENGGUNAKAN METODE AGILE**

**TUGAS AKHIR**

Diajukan Kepada Fakultas Sains dan Teknologi  
Universitas Islam Negeri (UIN) Ar-Raniry Banda Aceh  
Sebagai Salah Satu Beban Studi Memperoleh Gelar Sarjana  
pada Program Studi Teknologi Informasi

Oleh:


**FAIZ RAMADHANA SURYA**  
NIM. 210705037


**Mahasiswa Fakultas Sains dan Teknologi  
Program Studi Teknologi Informasi**

Disetujui untuk Dimunaqasyahkan Oleh:

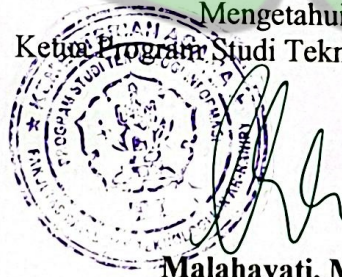
Pembimbing I,

Pembimbing II,

  
**Mulkan Fadhli, M.T.**  
NIP. 198811282020121006

  
**Dr. Hendri Ahmadian, M.I.M.**  
NIP. 198301042014031002

Mengetahui,  
Ketua Program Studi Teknologi Informasi



**Malahayati, M.T.**  
NIP. 198301272015032003

## LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Faiz Ramadhana Surya  
NIM : 210705037  
Program Studi : Teknologi Informasi  
Fakultas : Sains dan Teknologi  
Judul Tugas Akhir : Perancangan dan Implementasi Website Ulasan dan Informasi berbasis  
Laravel menggunakan Metode Agile

Dengan ini menyatakan bahwa dalam penulisan tugas akhir ini, saya:

1. Tidak menggunakan ide orang lain tanpa mampu mengembangkan dan mempertanggung jawabkan.
2. Tidak melakukan plagiasi terhadap naskah orang lain.
3. Tidak menggunakan karya orang lain tanpa menyebutkan sumber asli atau tanpa izin pemilik karya.
4. Tidak memanipulasi dan memalsukan data.
5. Mengerjakan sendiri karya ini dan mampu bertanggung jawab atas karya ini.

Bila dikemudian hari ada tuntutan dari pihak lain atas karya saya, dan telah melalui pembuktian yang dapat dipertanggung jawabkan dan ternyata memang ditemukan bukti bahwa saya telah melanggar pernyataan ini, maka saya siap dikenakan sanksi berdasarkan aturan yang berlaku di Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh.

Demikian Pernyataan ini saya buat dengan sesungguhnya dan tanpa paksaan dari pihak manapun.

Banda Aceh, 02 Februari 2026

Yang Menyatakan



(Faiz Ramadhana Surya)

## ABSTRAK

Nama : Faiz Ramadhana Surya  
NIM : 210705037  
Program Studi : Teknologi Informasi  
Judul : Perancangan Dan Implementasi Website Informasi Dan  
Ulasan Pesantren Berbasis Laravel Menggunakan  
Metode Agile  
Halaman : 99  
Pembimbing I : Mulkan Fadhli, M.T.  
Pembimbing II : Dr. Hendri Ahmadian, M.I.M.  
Kata Kunci : Website, Pesantren, Sistem Informasi, Laravel, Metode Agile,  
Ulasan Pengguna

Perkembangan teknologi informasi mendorong kebutuhan akan sistem informasi berbasis *web* yang mampu menyajikan informasi secara terpusat dan mudah diakses. Permasalahan yang dihadapi masyarakat adalah keterbatasan informasi dan ulasan pesantren yang masih tersebar dan bersifat subjektif. Penelitian ini bertujuan merancang dan mengimplementasikan *website* informasi dan ulasan pesantren berbasis Laravel menggunakan metode Agile. Metode Agile diterapkan untuk mendukung pengembangan sistem secara iteratif dan adaptif terhadap kebutuhan pengguna. Sistem menyediakan fitur informasi pesantren, pencarian data, serta ulasan dan *rating* pengguna yang dikelola melalui *admin* panel Filament. Pengujian dilakukan menggunakan *black box testing*. Hasil penelitian menunjukkan bahwa sistem mampu menyajikan informasi pesantren secara terstruktur, interaktif, dan mudah digunakan sehingga membantu masyarakat dalam pengambilan keputusan.

## ABSTRACT

Name : Faiz Ramadhana Surya

Student ID : 210705037

Study Program : Information Technology

Title : Design and Implementation of a Laravel-Based Islamic Boarding School Information and Review Website Using the Agile Method

Pages : 99

Supervisor I : Mulkan Fadhli, M.T.

Supervisor II : Dr. Hendri Ahmadian, M.I.M.

Keywords : Website, Islamic Boarding School, Information System, Laravel, Agile Method, User Reviews

The development of information technology has increased the need for web-based information systems that can provide centralized and easily accessible information. One of the problems faced by the community is the limited availability of reliable information and reviews of Islamic boarding schools, as the information is still scattered and subjective. This study aims to design and implement a Laravel-based website for providing information and reviews of Islamic boarding schools using the Agile method. The Agile method is applied to support an iterative and adaptive system development process based on user requirements. The system provides features such as Islamic boarding school information, data search, and user reviews and ratings, which are managed through a Filament-based admin panel. System testing is conducted using black box testing. The results show that the developed system is able to present Islamic boarding school information in a structured, interactive, and user-friendly manner, thereby assisting the community in decision-making.

## KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT atas segala limpahan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi ini dengan judul “Perancangan dan Implementasi *Website* Informasi dan Ulasan Pesantren Berbasis Laravel Menggunakan Metode Agile.” Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana pada Program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Ar-Raniry Banda Aceh.

Dalam proses penyusunan skripsi ini, penulis menyadari bahwa keberhasilan penelitian tidak terlepas dari dukungan, bimbingan, dan bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Kepada kedua orang tua tercinta yang selalu memberikan doa, dukungan, serta motivasi yang tiada henti selama penulis menjalani studi dan Kerja Praktik.
2. Mulkan Fadhli, M.T., selaku Pembimbing I, yang telah meluangkan waktu untuk memberikan arahan, bimbingan, dan masukan yang sangat berarti selama proses penyusunan skripsi ini.
3. Dr. Hendri Ahmadian, M.I.M., selaku Pembimbing II, atas kesabaran, saran, serta koreksi yang membangun demi kesempurnaan penelitian ini.
4. Malahayati, M. T, selaku Ketua Program Studi Teknologi Informasi yang telah memberikan izin dan dukungan yang diberikan dalam penyusunan dan penyelesaian penelitian ini.
5. Kepada Ibu Cut Ida Rahmadiana, S.Si., yang telah banyak membantu dalam berbagai urusan administrasi selama perkuliahan dan penyusunan skripsi.
6. Seluruh dosen dan staf pengajar di Fakultas Sains dan Teknologi UIN Ar- Raniry Banda Aceh yang telah memberikan ilmu dan wawasan selama masa perkuliahan.
7. Keluarga tercinta yang selalu memberikan doa, semangat, dan dukungan tak terhingga selama penulis menjalani pendidikan.
8. Rekan-rekan seperjuangan yang telah menjadi teman diskusi, saling mendukung, dan memberikan semangat sepanjang perjalanan perkuliahan ini.

Penulis menyadari bahwa skripsi ini masih memiliki keterbatasan dan jauh dari kesempurnaan. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang bersifat membangun demi penyempurnaan di masa yang akan datang. Semoga skripsi ini dapat memberikan manfaat dan kontribusi dalam pengembangan ilmu pengetahuan di bidang teknologi informasi, khususnya dalam pengembangan sistem informasi berbasis *web* dan penerapan metode Agile.

Akhir kata, penulis berharap agar hasil penelitian ini dapat memberikan manfaat bagi masyarakat, khususnya dalam membantu penyediaan informasi dan ulasan pesantren secara terstruktur, akurat, dan mudah diakses.



Banda Aceh, 28 Januari 2026

Penulis

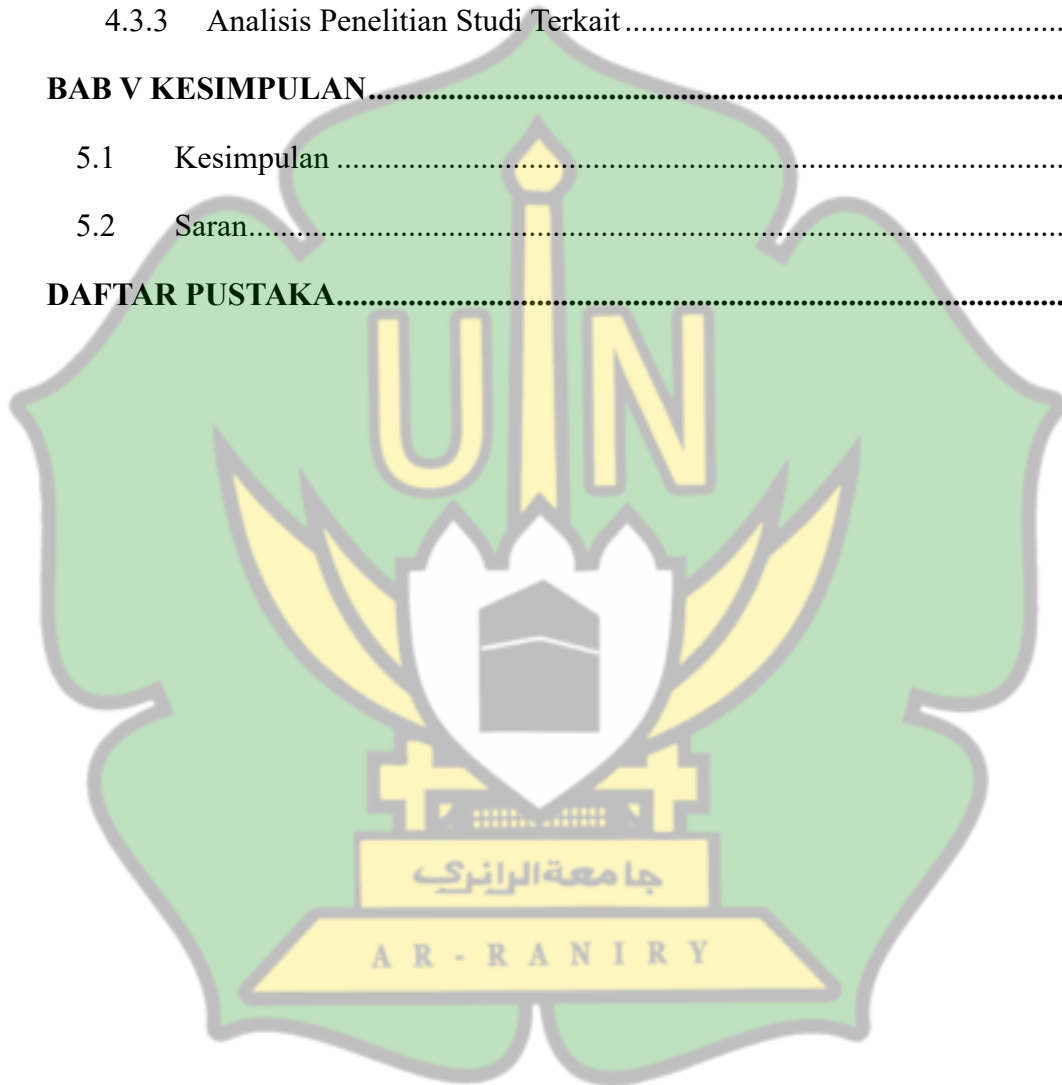
Faiz Ramadhana Surya

## DAFTAR ISI

<b>ABSTRAK .....</b>	<b>ii</b>
<b>ABSTRACT .....</b>	<b>iii</b>
<b>KATA PENGANTAR.....</b>	<b>iv</b>
<b>DAFTAR ISI.....</b>	<b>vi</b>
<b>DAFTAR GAMBAR .....</b>	<b>ix</b>
<b>DAFTAR TABEL .....</b>	<b>xi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1    Latar Belakang .....	1
1.2    Rumusan Masalah.....	3
1.3    Tujuan Penelitian.....	3
1.4    Manfaat Penelitian .....	3
1.5    Batasan Masalah.....	4
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>5</b>
2.1    Studi Terkait .....	5
2.2    Sistem Informasi .....	8
2.3    Website Ulasan.....	9
2.4    Pesantren .....	10
2.5    PHP .....	11
2.6 <i>User Interface dan User Experience</i> .....	12
2.6.1    UI ( <i>User Interface</i> ).....	12
2.6.2    UX ( <i>User Experience</i> ).....	13
2.6.3    Integrasi UI dan UX dalam Sistem Informasi Web.....	13

2.7	<i>Laravel</i> .....	14
2.8	Xampp.....	14
2.8.1	Kelebihan dan Kekurangan XAMPP .....	15
2.9	MySQL.....	16
2.9.1	Kelebihan dan Keterbatasan MySQL.....	17
2.10	Metode Agile.....	18
2.10.1	Jenis Jenis Metode Agile.....	19
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>21</b>
3.1	Tahapan Penelitian .....	21
3.1.1	Identifikasi Masalah.....	22
3.1.2	Pengumpulan Data .....	23
3.1.3	Perancangan sistem.....	24
3.1.4	Implementasi Sistem.....	33
3.1.5	Evaluasi.....	34
3.2	Waktu dan Lokasi Penelitian.....	35
3.3	Alat dan Bahan.....	35
3.3.1	Perangkat Keras ( <i>Hardware</i> ).....	35
3.3.2	Perangkat Lunak ( <i>Software</i> ).....	36
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>		<b>37</b>
4.1	Implementasi.....	37
4.1.1	Implementasi Sistem.....	37
4.1.2	Implementasi Basis data.....	58
4.1.3	Implementasi antar muka Pengguna (UI) .....	63
4.2	Pengujian sistem.....	74
4.2.1	Pengujian fungsional.....	74

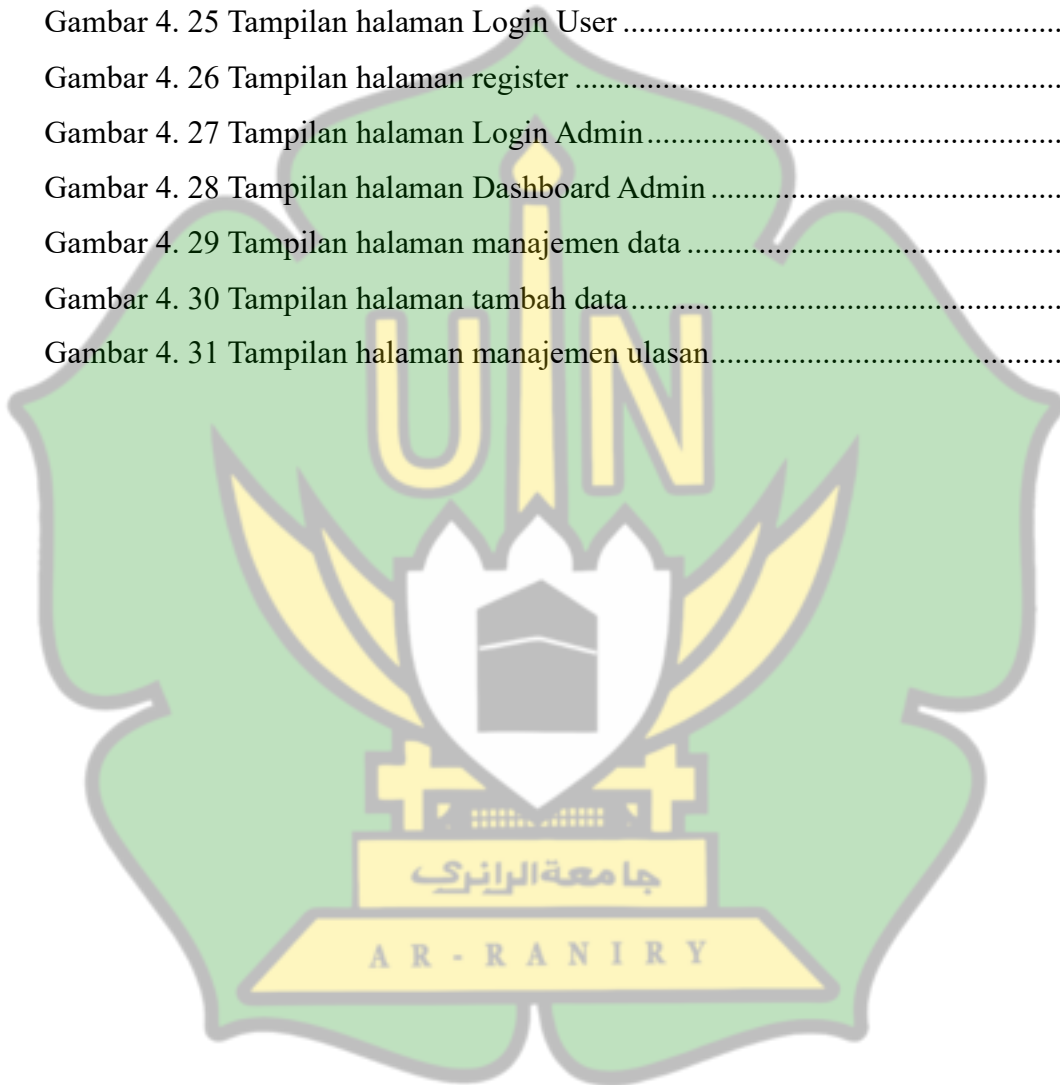
4.2.2	Pengujian integritas data dan logika bisnis .....	77
4.3	Pembahasan Hasil .....	79
4.3.1	Analisis Implementasi Teknologi.....	79
4.3.2	Pencapaian tujuan penelitian.....	80
4.3.3	Analisis Penelitian Studi Terkait.....	81
<b>BAB V KESIMPULAN.....</b>		<b>86</b>
5.1	Kesimpulan .....	86
5.2	Saran.....	86
<b>DAFTAR PUSTAKA.....</b>		<b>87</b>



## DAFTAR GAMBAR

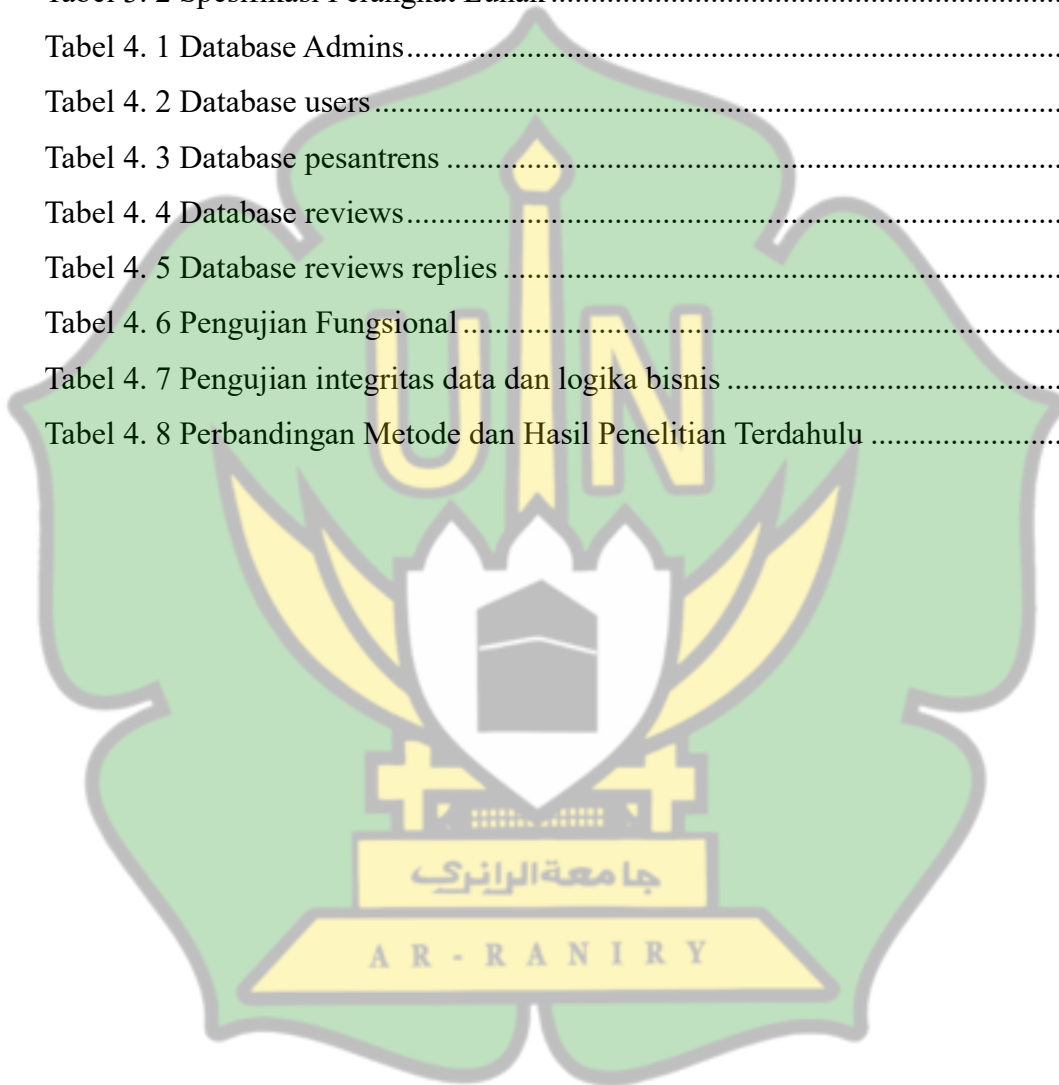
Gambar 2. 1 Alur Metode Agile.....	19
Gambar 3. 1 Tahapan Penelitian .....	21
Gambar 3. 2 Use Case Diagram.....	25
Gambar 3. 3 ERD Database .....	27
Gambar 3. 4 Wireframe Login Page .....	29
Gambar 3. 5 Wireframe Register Page.....	30
Gambar 3. 6 Wireframe Home Page .....	31
Gambar 3. 7 Wireframe Content Page .....	32
Gambar 3. 8 Wireframe Reviews Page .....	33
Gambar 4. 1 Kode Program Routing web.php.....	38
Gambar 4. 2 Kode Program method index PesantrenController.....	39
Gambar 4. 3 Kode Program method show PesantrenController.....	40
Gambar 4. 4 Kode Program method search PesantrenController .....	41
Gambar 4. 5 Kode Program method search PesantrenController .....	41
Gambar 4. 6 Kode Program method store ReviewController.....	42
Gambar 4. 7 Kode Program method create ReviewController .....	43
Gambar 4. 8 Kode Program index store, reply reviewreplycontroller.....	44
Gambar 4. 9 Kode Program models Pesantren.php .....	46
Gambar 4. 10 Kode Program models Review.php .....	48
Gambar 4. 11 Kode Program models ReviewReply.php.....	49
Gambar 4. 12 Kode Program Migration tabel pesantren .....	50
Gambar 4. 13 Kode Program Migration tabel review .....	51
Gambar 4. 14 Filament form dan table Pesantren Resource.....	53
Gambar 4. 15 Table Review Resource .....	54
Gambar 4. 16 Blade Views Show .....	55
Gambar 4. 17 Blade Views Search.....	56
Gambar 4. 18 Blade Views Ulasan.....	57
Gambar 4. 19 Schema Database Pesantrenly .....	59

Gambar 4. 20 Tampilan Halaman Utama.....	64
Gambar 4. 21 Tampilan Fitur Pencarian .....	65
Gambar 4. 22 Tampilan Detail Konten .....	66
Gambar 4. 23 Tampilan Detail Konten .....	67
Gambar 4. 24 Tampilan halaman Ulasan dan Rating.....	68
Gambar 4. 25 Tampilan halaman Login User .....	69
Gambar 4. 26 Tampilan halaman register .....	70
Gambar 4. 27 Tampilan halaman Login Admin .....	71
Gambar 4. 28 Tampilan halaman Dashboard Admin .....	72
Gambar 4. 29 Tampilan halaman manajemen data .....	72
Gambar 4. 30 Tampilan halaman tambah data.....	73
Gambar 4. 31 Tampilan halaman manajemen ulasan.....	74



## DAFTAR TABEL

Tabel 2. 1 Studi terkait .....	6
Tabel 3. 1 Spesifikasi Perangkat Keras .....	35
Tabel 3. 2 Spesifikasi Perangkat Lunak .....	36
Tabel 4. 1 Database Admins .....	60
Tabel 4. 2 Database users .....	60
Tabel 4. 3 Database pesantrens .....	61
Tabel 4. 4 Database reviews .....	62
Tabel 4. 5 Database reviews replies .....	62
Tabel 4. 6 Pengujian Fungsional .....	75
Tabel 4. 7 Pengujian integritas data dan logika bisnis .....	77
Tabel 4. 8 Perbandingan Metode dan Hasil Penelitian Terdahulu .....	82



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi informasi pada era digital saat ini telah membawa perubahan besar dalam cara masyarakat memperoleh informasi. Hampir seluruh aktivitas sehari-hari, termasuk di bidang pendidikan, kini dapat diakses secara cepat melalui internet (W. M. B. B. Sinaga & Firmansyah, 2024). Diperlukan sistem informasi berbasis *website* yang mampu memberikan layanan informasi pendidikan secara efektif dan terintegrasi, meliputi penyajian profil lembaga, kegiatan, hingga ulasan dari pengguna agar masyarakat dapat memperoleh gambaran yang lebih jelas dan akurat.

Pesantren merupakan salah satu lembaga pendidikan Islam tertua di Indonesia yang telah berperan besar dalam membentuk karakter, akhlak, dan keilmuan generasi muda (Rahma & Perawironegoro, 2024). Sejak berdirinya pesantren-pesantren tradisional pada abad ke-18 hingga kini, lembaga ini menjadi pusat pembelajaran agama sekaligus tempat pembinaan moral masyarakat. Pesantren tidak hanya mengajarkan ilmu agama seperti tafsir, hadits, fiqh, dan bahasa Arab, tetapi juga mengembangkan keterampilan hidup, kepemimpinan, serta wawasan kebangsaan bagi santri (Putri; et al., 2023).

Beberapa pesantren menawarkan pendidikan formal setingkat SMP atau SMA, sementara yang lain menyediakan pelatihan keterampilan, tahfiz Al-Qur'an, hingga program dakwah dan pemberdayaan Masyarakat (Asnawan, 2021). Meskipun jumlah pesantren di Indonesia terus meningkat, informasi mengenai lokasi, program, fasilitas, dan kualitasnya belum tersaji secara terpusat dan mudah diakses. Masyarakat masih mengandalkan cara tradisional seperti informasi dari mulut ke mulut, brosur cetak, dan kunjungan langsung. Namun, cara tersebut memiliki keterbatasan karena informasi sering subjektif, tidak selalu diperbarui, dan

memerlukan waktu serta biaya bagi calon santri atau orang tua yang ingin membandingkan beberapa pesantren sekaligus.

Pemanfaatan teknologi informasi berupa sistem informasi berbasis *website* dapat menjadi solusi dalam menyediakan informasi pesantren secara terstruktur, mudah diakses, serta dilengkapi fitur ulasan dan penilaian pengguna untuk memberikan gambaran kualitas pesantren (Lucita, 2025). *Framework* Laravel merupakan salah satu *framework* PHP yang banyak digunakan dalam pengembangan *website* karena memiliki struktur yang teratur, mendukung keamanan data, serta mempermudah pengembangan fitur CRUD (*Create, Read, Update, Delete*), (Mahardiansyah, 2024).

Pengembangan *website* ulasan dan informasi pesantren merupakan salah satu solusi yang relevan dengan kebutuhan masyarakat saat ini. Penelitian oleh (Ernungtyas et al., 2024) menyebut bahwa kualitas informasi dan kualitas sistem memengaruhi kepuasan pengguna terhadap situs *website*, yang relevan dengan konteks platform yang menyajikan informasi dan ulasan. Dalam Pembangunan sistem ini diperlukan pendekatan pengembangan yang fleksibel dan adaptif terhadap kebutuhan pengguna.

Metode Agile menjadi pilihan yang tepat dalam pengembangan sistem ini karena mampu memberikan hasil secara iteratif dan memungkinkan pengembang untuk terus menyesuaikan sistem berdasarkan masukan dari pengguna akhir (Kurniawan et al., 2025). Metode Agile lebih unggul karena tidak hanya berfokus pada perencanaan awal, tetapi juga menekankan kolaborasi, keterlibatan pengguna, dan pengembangan berulang untuk peningkatan sistem yang berkelanjutan.

Oleh karena itu, penulis melakukan penelitian berjudul “Perancangan dan Implementasi *Website* Ulasan dan Informasi Pesantren Menggunakan Metode Agile dan *Framework* Laravel” sebagai upaya menyediakan sistem informasi yang membantu masyarakat mengakses dan membandingkan informasi pesantren secara akurat dan interaktif.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah dalam penelitian ini adalah:

- 1) Bagaimana merancang sistem informasi berbasis *web* yang dapat menyajikan informasi dan ulasan pesantren secara efektif?
- 2) Bagaimana proses pengembangan sistem informasi pesantren menggunakan metode Agile agar dapat menyesuaikan kebutuhan pengguna secara iteratif?

## 1.3 Tujuan Penelitian

Penelitian ini memiliki tujuan sebagai berikut:

- 1) Merancang sistem informasi berbasis *web* untuk menampilkan informasi dan ulasan pesantren secara terstruktur dan mudah diakses.
- 2) Mengembangkan sistem informasi pesantren menggunakan metode Agile guna meningkatkan fleksibilitas dan kolaborasi selama proses pengembangan.

## 1.4 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

### a. Manfaat Teoritis

- 1) Menambah literatur terkait penerapan metode Agile dalam pengembangan sistem informasi berbasis *web*.
- 2) Menjadi referensi untuk pengembangan sistem informasi Pendidikan berbasis ulasan pengguna.

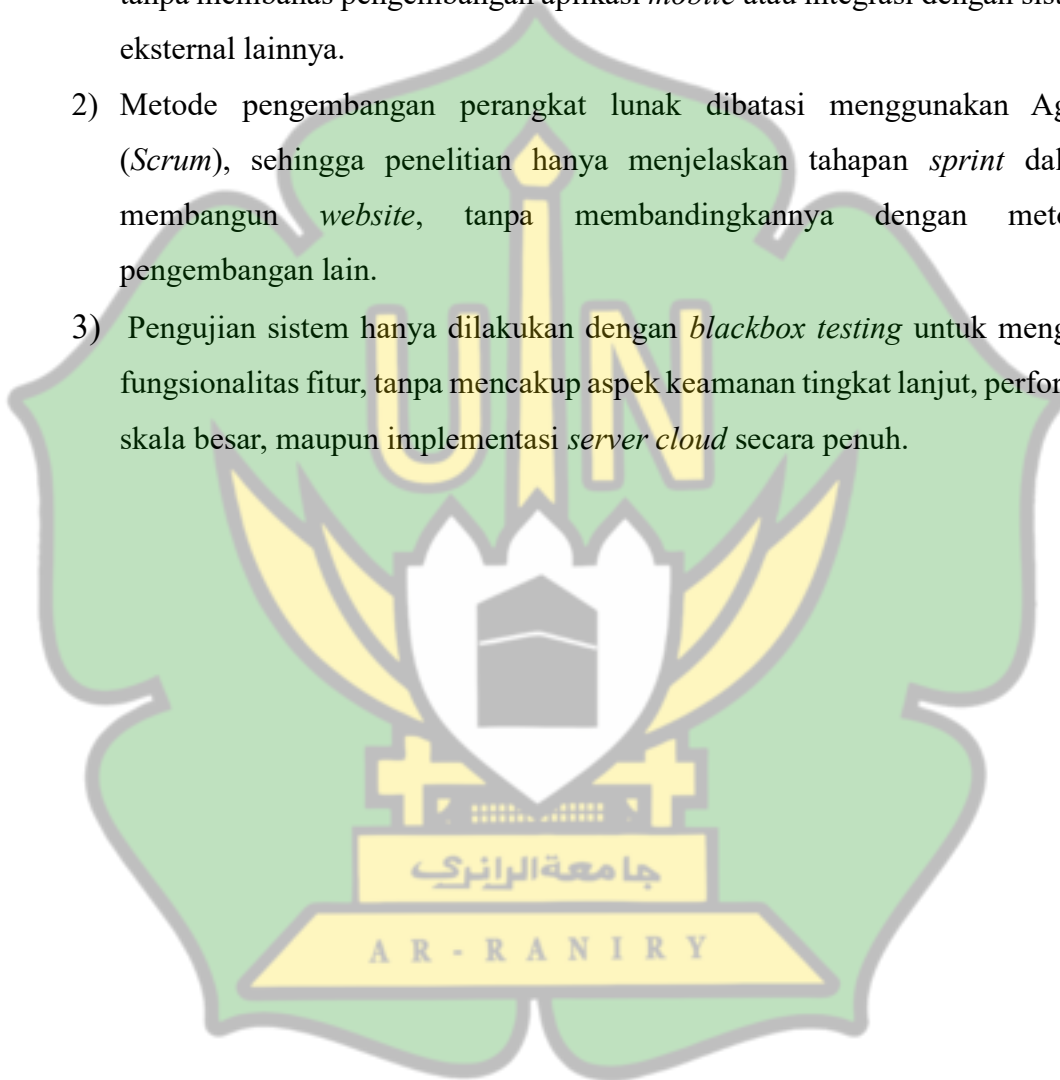
### b. Manfaat Praktis

- 1) Memberikan solusi teknologi kepada masyarakat dalam mengakses informasi pesantren secara cepat, akurat, dan objektif.
- 2) Membantu pihak pesantren dalam menyebarluaskan informasi dan memperkuat *branding* lembaga mereka.
- 3) Memberikan pengalaman penerapan *Framework* Laravel dengan pendekatan Agile bagi pengembang sistem.

## 1.5 Batasan Masalah

Penelitian ini memiliki beberapa batasan sebagai berikut:

- 1) Fokus penelitian hanya pada perancangan dan implementasi *website* berbasis Laravel yang menyediakan fitur informasi pesantren dan ulasan masyarakat, tanpa membahas pengembangan aplikasi *mobile* atau integrasi dengan sistem eksternal lainnya.
- 2) Metode pengembangan perangkat lunak dibatasi menggunakan Agile (*Scrum*), sehingga penelitian hanya menjelaskan tahapan *sprint* dalam membangun *website*, tanpa membandingkannya dengan metode pengembangan lain.
- 3) Pengujian sistem hanya dilakukan dengan *blackbox testing* untuk menguji fungsionalitas fitur, tanpa mencakup aspek keamanan tingkat lanjut, performa skala besar, maupun implementasi *server cloud* secara penuh.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Studi Terkait

Beberapa penelitian terdahulu yang relevan turut mendukung pengembangan *website* ulasan dan informasi pesantren dengan menggunakan metode Agile dan *framework* Laravel, karena keduanya terbukti mampu meningkatkan efisiensi proses pengembangan serta kualitas sistem yang dihasilkan.

Penelitian yang dilakukan oleh (Zaharuddin et al., 2020), mengembangkan sistem penilaian santri berbasis *web* menggunakan *framework* Laravel di Al-Izzah Leadership School. Hasil penelitian menunjukkan bahwa Laravel mampu menyediakan sistem yang terintegrasi dan mudah diakses oleh wali santri melalui internet.

Dari sisi teknis, penelitian oleh (Rahmawati & Sumarsono, 2024), yang berjudul Desain Pengembangan *Website* dengan Arsitektur *Model View Controller* pada *Framework* Laravel menunjukkan bahwa penggunaan Laravel dengan arsitektur MVC dan desain responsif dapat meningkatkan performa pencarian serta keaktualan informasi.

Lebih jauh, (Haris, 2023), menekankan pentingnya transformasi *digital* pendidikan pesantren dalam era Society 5.0. Ia menyatakan bahwa penggunaan *website* dan media sosial sangat penting sebagai sarana *branding*, administrasi, serta literasi *digital*. Namun, banyak pesantren masih menghadapi kendala pada ketersediaan tenaga teknis untuk memelihara sistem *digital* tersebut.

Selain itu, penelitian oleh (Hakam et al., 2022), menunjukkan bahwa penggabungan metode Agile dengan *framework* Laravel mampu meningkatkan efisiensi dalam pengelolaan data zakat, termasuk input dan penyimpanan transaksi, serta menghasilkan sistem yang lebih cepat dan responsif.

Secara keseluruhan, penelitian-penelitian tersebut menegaskan bahwa penggunaan *framework* Laravel sangat relevan untuk membangun sistem berbasis *web* di lingkungan pesantren. Di sisi lain, penerapan metode Agile, terutama *Scrum*, terbukti mampu mempercepat proses pengembangan dan menyesuaikan kebutuhan pengguna secara fleksibel. Hal ini menjadi dasar penting dalam perancangan dan implementasi *website* ulasan dan informasi pesantren menggunakan metode Agile dan *framework* Laravel.

Tabel 2. 1 Studi terkait

NO	REFERENSI	METODE	HASIL
1	Zaharuddin, Pradana, & Brata (2020)	<i>Framework</i> Laravel (MVC) digunakan untuk merancang sistem penilaian santri berbasis <i>web</i> dengan pemisahan antara <i>model, view, dan controller</i> sehingga memudahkan pengelolaan data.	Hasil penelitian menunjukkan Laravel mampu menyediakan sistem yang terintegrasi, <i>user friendly</i> , dan dapat diakses oleh wali santri melalui internet, sehingga proses monitoring nilai santri menjadi lebih transparan dan efektif.
2	Rahmawati & Sumarsono (2024)	Arsitektur MVC dengan <i>Framework</i> Laravel digunakan untuk membangun desain <i>website</i> , dengan pemisahan logika aplikasi, antarmuka, dan basis data.	Penggunaan Laravel berbasis MVC menghasilkan performa pencarian data yang lebih cepat, tampilan yang responsif, serta

			keaktualan informasi yang lebih baik dibandingkan pendekatan konvensional.
3	Haris (2023)	Pendekatan studi literatur dan analisis fenomena digitalisasi pesantren yang menekankan pemanfaatan <i>website</i> dan media sosial dalam pendidikan.	Hasil menunjukkan bahwa digitalisasi melalui <i>website</i> dan media sosial sangat penting untuk <i>branding</i> pesantren, administrasi, dan literasi digital, meskipun masih terdapat kendala SDM teknis dalam pengelolaan sistem.
4	Hakam, Triayudi, & Hayati (2022)	Metode Agile (Scrum) dikombinasikan dengan <i>Framework</i> Laravel (MVC) dalam membangun sistem manajemen zakat berbasis <i>web</i> untuk mengelola data dan transaksi.	Integrasi Agile dan Laravel terbukti meningkatkan efisiensi input, penyimpanan, serta pengolahan data zakat, menghasilkan sistem yang lebih cepat, responsif, dan mampu menyesuaikan

			kebutuhan pengguna secara berkelanjutan.
--	--	--	--

## 2.2 Sistem Informasi

Sistem Informasi adalah sistem yang menggabungkan elemen teknologi, manusia, dan data untuk mengelola pengumpulan, pemrosesan, penyimpanan, dan distribusi informasi agar mendukung operasi dan pengambilan keputusan organisasi. Misalnya, menurut (Juriatun Nasirin et al., 2023), sistem informasi membantu memecahkan masalah proses pendaftaran yang masih manual, lambat, dan terkadang tidak efisien karena penggunaan kertas. Dengan sistem *web* dan basis data MySQL, informasi pendaftaran menjadi lebih cepat disebarkan, proses pendaftaran lebih ringkas, dan laporan penerimaan siswa baru dibuat lebih sistematis.

Komponen inti sistem informasi meliputi *input*, proses, *output*, basis data, serta pengguna / *user* dan kontrol. Dari (Amarulla et al., 2023), sistem informasi akademik berbasis *web* mencakup pengelolaan data siswa, guru, nilai, dan kehadiran. Data tersebut dimasukkan oleh *admin* atau guru sebagai *input*, kemudian diproses oleh sistem (misalnya menghitung nilai, rekap kehadiran), disimpan di basis data, dan dikeluarkan dalam bentuk tampilan laporan, grafik, atau halaman akademik yang dapat diakses oleh pengguna yang berwenang (guru, kepala sekolah, wali). Sistem juga memerlukan kontrol keamanan seperti autentikasi dan hak akses agar hanya pengguna yang berwenang yang bisa melihat atau memodifikasi data sensitif.

Manfaat penerapan sistem informasi berbasis *web* di organisasi pendidikan antara lain peningkatan efisiensi operasional, percepatan penyebaran informasi, serta peningkatan transparansi dan akuntabilitas. Berdasarkan studi di MTS Al-Fajar dan SD Xaverius, di mana sebelum ada sistem *web*, proses masih banyak manual dan rentan kesalahan. Setelah sistem terpasang, proses pendaftaran dan pengelolaan akademik menjadi lebih cepat, laporan tersedia dalam waktu lebih singkat, dan pengguna (guru / admin / wali) bisa lebih mudah memonitor status data. Ini sangat relevan untuk penelitian perancangan *web* ulasan dan informasi pesantren: sistem

harus dapat menangani input ulasan, moderasi, penyimpanan, dan menampilkan *output* kepada publik/pengguna/pesantren dengan cara yang aman dan efisien.

### 2.3 Website Ulasan

Website ulasan merupakan sistem berbasis *web* yang menyediakan ruang bagi pengguna untuk memberikan penilaian (*rating*) dan tanggapan (*review*) terhadap layanan, produk, atau lembaga tertentu. *Website* ini menjadi media berbagi pengalaman yang sangat berpengaruh dalam pengambilan keputusan pengguna lain. Menurut (Ariansyah et al., 2025), ulasan *online* berperan penting dalam membentuk persepsi dan kepercayaan calon pengguna terhadap suatu layanan. Melalui sistem ulasan yang terbuka, pengguna dapat memperoleh informasi yang lebih objektif karena bersumber langsung dari pengalaman orang lain. *Website* seperti TripAdvisor misalnya, menjadi rujukan utama wisatawan dalam memilih destinasi, hotel, dan layanan perjalanan. Dalam konteks pendidikan atau pesantren, konsep ini dapat diadaptasi untuk menampilkan pengalaman santri, wali santri, atau alumni dalam bentuk ulasan yang bersifat membangun dan informatif, sehingga membantu calon santri dalam memilih pesantren yang sesuai dengan kebutuhan mereka.

Komponen utama dari *website* ulasan biasanya meliputi fitur *rating* (bintang atau angka), kolom komentar pengguna, pencarian dan penyaringan (*filtering*), serta pengelolaan ulasan oleh admin. Menurut (S. G. Sinaga & Aprillia, 2025), keberadaan *rating* dan *review* meningkatkan interaksi pengguna dan memperkuat kredibilitas sistem karena menampilkan pendapat langsung dari konsumen. Fitur ini berfungsi sebagai bentuk partisipasi aktif pengguna yang dapat memengaruhi reputasi suatu lembaga. Dalam sistem informasi ini, penulis menggunakan *framework* salah satunya adalah Laravel. menggunakan *framework* Laravel karena Laravel merupakan *framework* aplikasi *web* terkini yang bersifat *open source* dan banyak digunakan dalam perancangan aplikasi *web* yang cepat dan mudah (Indah Melyani et al., 2023).

Penerapan sistem ulasan berbasis *web* tidak hanya berdampak pada peningkatan transparansi informasi, tetapi juga pada penguatan kepercayaan publik. Studi oleh (Gunawan Utami & Aghniarahma Junia, 2024), juga menunjukkan bahwa perancangan fitur ulasan yang interaktif dapat mendorong pengguna lebih aktif

memberikan penilaian dan masukan. Hal ini menunjukkan bahwa desain antarmuka dan kemudahan penggunaan sangat berperan dalam keberhasilan sistem ulasan. Dalam konteks pengembangan *website* ulasan pesantren, fitur seperti *rating* kualitas pengajaran, fasilitas asrama, dan lingkungan sosial pesantren dapat diterapkan. Sistem ini tidak hanya menampilkan informasi profil lembaga, tetapi juga memperlihatkan pengalaman nyata dari santri dan wali santri, sehingga mendukung pengambilan keputusan yang lebih rasional dan transparan di kalangan masyarakat.

#### **2.4 Pesantren**

Pesantren merupakan lembaga pendidikan Islam tradisional yang telah menjadi bagian penting dari sistem pendidikan di Indonesia. Secara etimologis, istilah “pesantren” berasal dari kata “santri” yang berarti pelajar agama, dengan awalan “pe-” dan akhiran “-an” yang menunjukkan tempat tinggal santri. Pesantren memiliki ciri khas berupa adanya kyai sebagai pemimpin spiritual dan pendidik utama, santri sebagai peserta didik, masjid sebagai pusat kegiatan ibadah dan belajar, serta asrama (pondok) sebagai tempat tinggal santri. Menurut (Andy Soebroto, 2025), pesantren tidak hanya berfungsi sebagai lembaga pengajaran agama, tetapi juga sebagai lembaga sosial dan pembentuk karakter umat. Di dalam pesantren, santri belajar berbagai disiplin ilmu Islam, seperti tafsir, fiqh, hadits, dan akhlak, yang semuanya diarahkan untuk mencetak generasi berilmu dan berakhlak mulia.

Dalam perkembangannya, pesantren mengalami transformasi dari model salafiyah (tradisional) menuju khalafiyah (modern) yang memadukan antara pendidikan agama dan umum. Pesantren salafiyah masih mempertahankan metode pengajaran klasik berbasis kitab kuning dan halaqah, sedangkan pesantren modern mulai mengadopsi sistem pendidikan formal serta kurikulum nasional. Berdasarkan penelitian (Putri & Kurniawan, 2025), transformasi pesantren sangat dipengaruhi oleh kebutuhan masyarakat modern terhadap pendidikan yang integratif. Pesantren kini tidak hanya fokus pada pendidikan keagamaan, tetapi juga penguasaan teknologi, kewirausahaan, dan literasi digital. Adaptasi ini menjadikan pesantren lebih inklusif dan relevan dalam menghadapi era revolusi industri 4.0.

## 2.5 PHP

PHP adalah bahasa pemrograman yang melengkapi HTML, dengan PHP memudahkan *developer* sistem/ aplikasi dapat membuat *website* dengan dinamis yang digunakan untuk memproses data dan mengolahnya menjadi sebuah tampilan yang bagus. semua *syntax* yang ada di dalam PHP dijalankan sepenuhnya di *server*, sehingga tampilah yang ada di *browser* hanya hasilnya saja. PHP merupakan bahasa berbentuk *script* yang hanya di proses di *server* dan ditampatkan di *server*, hasilnya akan ditampilkan di *browser* tempat pengguna aplikasi/sistem. PHP adalah bahasa yang berbentuk *script* yang 13 berhubungan dengan HTML, kemudian sintaxnya diproses di server yang digunakan untuk membuat tampilan *website* yang dinamis, seperti halnya *Active Server Pages* (ASP) atau *Java Server Pages* (JSP), PHP adalah bahasa pemrograman yang sering digunakan, *syntax* yang ada dalam PHP bersifat *Open Source* (Hermiati et al., 2021).

PHP (*Hypertext Preprocessor*) merupakan salah satu bahasa pemrograman yang paling banyak digunakan dalam pengembangan aplikasi berbasis *web*. Menurut (Hartati et al., 2023), dalam penelitiannya yang Menggunakan Bahasa Pemrograman PHP dan Database MySQL, PHP berfungsi sebagai bahasa pemrograman sisi server (*server-side scripting*) yang mampu menghubungkan antarmuka pengguna (*front-end*) dengan basis data (*back-end*). PHP digunakan untuk memproses data masukan dari pengguna, menampilkan hasil dalam bentuk halaman *web* dinamis, dan melakukan operasi seperti penyimpanan data ke *database* MySQL. Dengan kemampuan ini, PHP sangat cocok diterapkan untuk aplikasi pendidikan, bisnis, dan layanan publik berbasis *web* yang membutuhkan interaksi data secara *real-time*.

Penggunaan PHP dalam pengembangan sistem informasi modern umumnya dilakukan dengan bantuan *framework* seperti Laravel, karena mampu mempercepat proses pembuatan aplikasi serta menjaga konsistensi dan keamanan kode. Menurut (Pratiwi & Heriadi, 2023), *framework* Laravel yang berbasis PHP menyediakan struktur *Model-View-Controller* (MVC) yang memisahkan antara logika aplikasi, tampilan, dan basis data, sehingga pengembangan lebih terorganisir dan mudah dipelihara. Selain itu, Laravel memiliki fitur seperti *routing*, *middleware*, dan

otentikasi pengguna yang memperkuat keamanan sistem. Dalam konteks penelitian ini, PHP Laravel dapat digunakan untuk membangun sistem *web* ulasan dan informasi pesantren karena mendukung pengelolaan data ulasan, manajemen pengguna, serta keamanan akses dengan efisien.

Selain framework, PHP juga dapat digunakan secara native (tanpa *framework* besar) untuk aplikasi berskala kecil hingga menengah. Menurut penelitian (Menrisal et al., 2022), penggunaan PHP Native memberikan fleksibilitas tinggi bagi pengembang dalam mengatur struktur kode sesuai kebutuhan proyek. PHP Native cocok untuk sistem yang tidak memerlukan arsitektur kompleks, namun tetap membutuhkan pengolahan data dinamis dan koneksi dengan basis data MySQL. Hasil penelitian tersebut menunjukkan bahwa sistem *e-learning* berbasis PHP Native terbukti efektif, praktis, dan mudah dikembangkan sesuai kebutuhan lembaga pendidikan. Berdasarkan hasil-hasil penelitian ini, PHP dapat disimpulkan sebagai bahasa pemrograman yang handal, fleksibel, dan relevan untuk pengembangan sistem *web* dinamis seperti aplikasi ulasan dan informasi pesantren.

## **2.6 *User Interface* dan *User Experience***

### **2.6.1 UI (*User Interface*)**

*User Interface* (UI) merupakan aspek penting dalam pengembangan sistem informasi berbasis web karena berfungsi sebagai jembatan interaksi antara pengguna dan sistem. Menurut penelitian oleh (Fauzi et al., 2025), UI memiliki peran utama dalam menentukan tingkat kemudahan, kejelasan, dan efisiensi interaksi pengguna terhadap sistem digital. UI yang baik mampu menyajikan tampilan yang konsisten, terstruktur, dan intuitif, sehingga pengguna dapat memahami fungsi setiap elemen dengan cepat tanpa memerlukan panduan khusus.

Penelitian oleh (Wijayanti et al., 2022), desain antarmuka pengguna yang efektif harus memperhatikan aspek estetika, warna, tipografi, dan penempatan elemen agar menciptakan pengalaman visual yang nyaman. UI juga harus responsif dan adaptif terhadap berbagai perangkat seperti desktop, tablet, maupun *smartphone* agar dapat diakses oleh seluruh lapisan pengguna. Penelitian tersebut menegaskan

bahwa penggunaan prinsip *user-centered design* merupakan pendekatan terbaik dalam pengembangan UI karena menempatkan kebutuhan dan kenyamanan pengguna sebagai prioritas utama.

### 2.6.2 UX (*User Experience*)

*User Experience* (UX) adalah pengalaman menyeluruh yang dirasakan oleh pengguna saat berinteraksi dengan sistem. Menurut penelitian oleh (Maulidiyah et al., 2022), *User Experience* (UX) merupakan aspek yang berfokus pada bagaimana pengguna berinteraksi, merasakan, dan menilai kemudahan serta kenyamanan suatu sistem atau aplikasi. Penelitian ini menunjukkan bahwa elemen UX seperti *usability*, *accessibility*, dan *aesthetic design* berpengaruh signifikan terhadap tingkat kepuasan pengguna. Semakin baik pengalaman pengguna dalam mengakses fitur aplikasi, semakin tinggi pula minat mereka untuk menggunakan aplikasi tersebut secara berulang.

Selain itu, studi oleh (Wijaya et al., 2025), menjelaskan bahwa desain UX yang baik tidak hanya memperhatikan aspek visual, tetapi juga alur interaksi pengguna yang intuitif. Dengan pendekatan berbasis UX, sistem informasi dapat memberikan kenyamanan penggunaan serta mengurangi tingkat kesalahan pengguna dalam melakukan navigasi atau input data. Hal ini sejalan dengan prinsip *human-centered design* yang menempatkan kebutuhan pengguna sebagai prioritas utama dalam proses pengembangan sistem.

### 2.6.3 Integrasi UI dan UX dalam Sistem Informasi Web

UI dan UX memiliki hubungan yang saling melengkapi dalam menentukan keberhasilan sebuah *website*. Menurut (Rahayu et al., 2024), pendekatan *User Centered Design* (UCD) menempatkan pengguna sebagai pusat dari seluruh proses perancangan. Metode ini melibatkan beberapa tahap penting yaitu pengumpulan kebutuhan pengguna, pembuatan prototipe, dan pengujian antarmuka. Hasil penelitian menunjukkan bahwa penerapan prinsip UCD mampu meningkatkan nilai *usability website* karena rancangan dibuat berdasarkan kebutuhan dan perilaku pengguna sebenarnya. Dalam konteks pengembangan *web* ulasan dan informasi

pesantren, penerapan prinsip UCD sangat relevan agar pengguna dapat dengan mudah memberikan ulasan, membaca informasi, dan menavigasi situs tanpa hambatan. Desain UI/UX yang baik juga akan meningkatkan kepercayaan dan loyalitas pengguna terhadap sistem informasi pesantren.

## 2.7 *Laravel*

Laravel adalah sebuah *framework* PHP modern yang bersifat *open source*, yang dirancang untuk memudahkan pengembangan aplikasi *web* dengan menyediakan struktur yang terorganisir, serta banyak fitur bawaan yang mempercepat pembangunan aplikasi. Pada penelitian yang dilakukan oleh (Pratiwi & Heriadi, 2023), Laravel digunakan untuk membangun aplikasi manajemen keuangan yang mengelola transaksi penjualan, pembelian, stok barang, hutang/piutang dan laporan keuangan. Di studi ini Laravel terbukti memberi kemudahan dalam *routing*, pengendalian akses, manajemen basis data, dan menghasilkan antarmuka yang relatif stabil dan mudah dipelihara.

Selain itu, Laravel juga mendukung metode pengembangan yang lebih adaptif dan efisien. Contohnya dalam penelitian oleh (Robbani et al., 2025), menunjukkan bahwa Laravel dipakai bersama pendekatan Agile; setiap iterasi memperbaiki fitur berdasarkan umpan balik pengguna gudang, mempercepat penyampaian modul kerja, dan membantu menangani perubahan kebutuhan sistem tanpa harus merombak total arsitektur.

Selain itu penelitian oleh (Viani et al., 2024), juga memberikan gambaran bagaimana Laravel mampu menyediakan fitur-otentikasi, manajemen data, tampilan *web* yang responsif, serta pengelolaan alur kerja pendaftaran siswa baru secara digital. Studi ini menyebut bahwa Laravel mempermudah pengelolaan *form input*, validasi data secara otomatis, dan penyimpanan data dalam basis data yang aman dan konsisten, selain meminimalisir kesalahan manusia dibanding sistem manual.

## 2.8 *Xampp*

XAMPP merupakan perangkat lunak *open source* yang menyediakan paket lengkap untuk membangun dan menguji aplikasi *web* secara lokal sebelum diunggah

ke *server* publik. Berdasarkan penelitian (Kalsum Siregar et al., 2024), XAMPP menggabungkan Apache sebagai *web server*, MySQL sebagai *database*, PHP sebagai bahasa pemrograman sisi *server*, serta phpMyAdmin sebagai alat bantu pengelolaan basis data. Dengan adanya paket ini, pengembang tidak perlu melakukan instalasi terpisah untuk tiap komponen *server*, sehingga proses pengembangan sistem informasi menjadi lebih cepat dan praktis. Lingkungan lokal yang disediakan XAMPP juga memungkinkan pengujian fitur *web* tanpa memerlukan koneksi internet.

Selanjutnya, penelitian (Kalsum Siregar et al., 2024) menunjukkan bahwa XAMPP memiliki antarmuka yang sederhana dan mendukung berbagai sistem operasi seperti Windows, Linux, dan macOS. Dalam penelitian tersebut, XAMPP digunakan untuk merancang sistem informasi perpustakaan berbasis *web*. Proses pengembangan dilakukan secara *offline* sehingga tim peneliti dapat menguji dan memperbaiki kode sebelum dipublikasikan ke *server* daring. Hal ini membuktikan bahwa XAMPP efektif digunakan pada tahap pengembangan awal, terutama bagi mahasiswa atau peneliti yang membutuhkan *server* lokal sebagai sarana uji coba sistem informasi.

### **2.8.1 Kelebihan dan Kekurangan XAMPP**

XAMPP merupakan perangkat lunak web server bersifat open-source yang digunakan sebagai lingkungan pengembangan aplikasi web karena mudah diinstal dan telah terintegrasi dengan Apache, MySQL/MariaDB, serta PHP dalam satu paket. Kelebihan XAMPP adalah kemudahan penggunaan dan kesesuaiannya untuk proses pengembangan dan pengujian aplikasi secara lokal. Namun, XAMPP memiliki keterbatasan dari sisi keamanan dan performa sehingga tidak disarankan untuk digunakan pada lingkungan produksi atau aplikasi dengan beban tinggi. Berikut beberapa kelebihan dan kekurangannya.

#### **a. Kelebihan XAMPP**

Berikut merupakan beberapa kelebihan XAMPP

- Mudah Diinstal dan Digunakan

XAMPP menyediakan paket lengkap (Apache, MySQL, PHP, phpMyAdmin) sehingga tidak perlu instalasi terpisah.

- Mendukung *Multi-Platform*  
Dapat dijalankan di berbagai sistem operasi seperti Windows, Linux, dan macOS.
- Cocok untuk Pengembangan Lokal  
Ideal untuk pengujian aplikasi secara *offline* sebelum dipublikasikan ke *server* sebenarnya.
- Memiliki Antarmuka Sederhana  
*Control Panel* memudahkan pengguna dalam mengatur layanan *server* seperti Apache dan MySQL.
- Terintegrasi dengan phpMyAdmin  
Mendukung pengelolaan *database* secara visual dan mudah diakses.

#### **b. Kekurangan XAMPP**

Berikut merupakan beberapa kekurangan XAMPP

- Keamanan Rendah untuk Produksi  
Pengaturan default XAMPP bersifat terbuka, sehingga tidak disarankan untuk *server* publik.
- Kurang Efisien untuk Data Besar  
Tidak optimal dalam menangani beban aplikasi atau data skala besar.
- Konsumsi Sumber Daya Tinggi  
Saat semua layanan aktif, dapat memperlambat kinerja perangkat dengan spesifikasi rendah.
- Versi Komponen Tidak Selalu Terbaru  
Pembaruan PHP dan MySQL pada XAMPP sering tertinggal dari versi resmi.

## **2.9 MySQL**

MySQL adalah sistem manajemen basis data relasional (*Relational Database Management System*, RDBMS) gratis dan open source yang menggunakan bahasa SQL (*Structured Query Language*) untuk mengelola data dalam tabel terstruktur.

Dalam jurnal penelitian oleh (Sukardi & Putra, 2022), MySQL digunakan sebagai basis data untuk menyimpan informasi seperti data pengguna (santri), materi pembelajaran, tugas, dan penilaian. Penggunaan MySQL dalam penelitian ini menunjukkan bahwa MySQL cukup andal dalam menangani operasi CRUD (*Create, Read, Update, Delete*) dan query pencarian sederhana dalam konteks e-learning. Karena MySQL mendukung transaksi dan relasi antar tabel, data yang saling terkait (misalnya data pengguna dan nilai) dapat dikelola dengan baik.

Selain itu, dalam jurnal penelitian oleh (Wardani, 2013), aplikasi ini menggunakan MySQL untuk menyimpan data nilai siswa, mengkalkulasi rata-rata nilai, menghasilkan laporan, serta menyediakan antarmuka pencarian dan filter data nilai siswa. MySQL digunakan karena kecepatan responsnya relatif baik dalam lingkungan sekolah, kemudahan dalam integrasi dengan PHP, dan karena paket *hosting / server* lokal banyak mendukung MySQL. Data nilai bisa diakses secara *real time* oleh guru dan wali melalui antarmuka *web*, memanfaatkan koneksi *backend* ke MySQL.

### **2.9.1 Kelebihan dan Keterbatasan MySQL**

MySQL merupakan sistem manajemen basis data relasional yang banyak digunakan karena bersifat open-source, memiliki performa yang baik, serta mudah diintegrasikan dengan berbagai aplikasi berbasis *web*. (PRAYOGA, 2024) MySQL mendukung transaksi ACID sehingga mampu menjaga konsistensi dan keakuratan data, serta cocok digunakan untuk aplikasi skala kecil hingga menengah. Namun, MySQL memiliki keterbatasan dalam menangani beban data yang sangat besar dan query kompleks tanpa optimasi khusus, berikut merupakan beberapa kelebihan dan keterbatasan MySQL.

#### **a. Kelebihan MySQL**

Kelebihan MySQL yang muncul dari studi-studi tersebut mencakup:

- Kemudahan integrasi dengan PHP dan *framework web* berbasis PHP, memudahkan pengembangan aplikasi berbasis *web*.

- Kecepatan dalam operasi CRUD dan pencarian data dalam jumlah yang tidak terlalu besar (skala pendidikan, inventaris, perpustakaan).
- Biaya relatif rendah karena bersifat *open source* dan banyak tutorial serta sumber daya pendukung.
- Dukungan terhadap *backup*, *indexing*, dan struktur relasi antar tabel yang membantu menjaga integritas data.

#### b. Keterbatasan MySQL

Keterbatasan juga dibahas dalam beberapa penelitian:

- Ketika data sangat besar atau banyak *join* antar tabel, performa MySQL bisa menurun jika tidak dioptimasi (contoh belum ditemukan spesifik di studi yang disebut, tapi implisit dari studi perbandingan MySQL vs MongoDB bahwa NoSQL seperti MongoDB unggul dalam beban tertentu).
- Untuk fitur-fitur tertentu seperti skala sangat besar (juta baris), *clustering*, *high availability*, mungkin membutuhkan konfigurasi lanjutan atau tambahan arsitektur pendukung.

### 2.10 Metode Agile

Metode Agile merupakan pendekatan pengembangan perangkat lunak yang bersifat adaptif, kolaboratif, dan iteratif, di mana sistem dikembangkan secara bertahap (*incremental development*) dan dievaluasi secara berulang. Tujuan utama Agile adalah agar pengembang dapat merespons perubahan kebutuhan pengguna secara cepat tanpa harus memulai proyek dari awal. Contoh alur dari metode Agile dapat dilihat pada gambar 2.1.



Gambar 2. 1 Alur Metode Agile

Menurut (Gunawan & StevanieBK, 2025), Agile dikembangkan sebagai solusi atas kekakuan model tradisional seperti Waterfall, karena Agile memungkinkan tim melakukan perubahan fitur di setiap iterasi berdasarkan umpan balik pengguna. Agile menekankan kerja tim, komunikasi efektif, dan keterlibatan aktif dari pengguna selama proses pengembangan berlangsung.

Dalam penerapannya, Agile membagi proyek ke dalam siklus pengembangan pendek (biasanya 1–4 minggu), di mana setiap siklus melibatkan tahap perencanaan, pengkodean, pengujian, dan umpan balik. Hal ini membuat Agile sangat cocok untuk pengembangan sistem informasi berbasis *web* seperti *website* ulasan dan informasi pesantren, karena kebutuhan pengguna dapat berubah sewaktu-waktu dan harus segera diakomodasi oleh tim pengembang.

### 2.10.1 Jenis Jenis Metode Agile

Metode Agile memiliki beberapa kerangka kerja (*framework*) atau jenis pendekatan yang umum digunakan dalam pengembangan perangkat lunak modern. Berdasarkan jurnal (Ambara et al., 2025), terdapat tiga metode Agile yang paling populer, yaitu:

#### a. Scrum

Scrum merupakan pendekatan Agile yang membagi proyek ke dalam beberapa *sprint* (periode kerja singkat, biasanya 2 minggu). Setiap *sprint* menghasilkan produk yang siap diuji dan dapat digunakan sebagian. Scrum memiliki tiga peran utama, yaitu *Product Owner* (penanggung jawab kebutuhan pengguna), Scrum

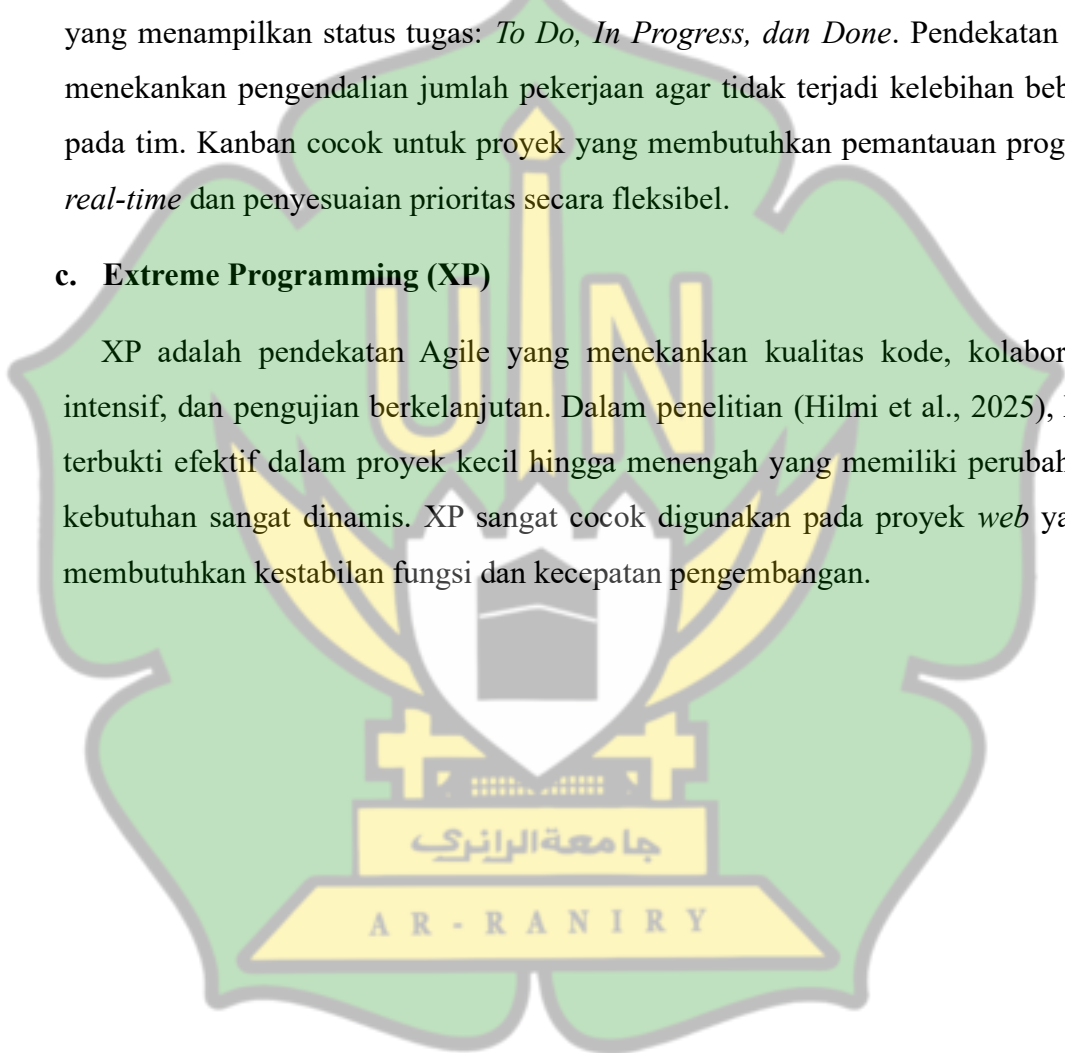
*Master* (fasilitator proses), dan *Development Team* (pengembang sistem). Scrum cocok untuk proyek besar yang memerlukan koordinasi antar tim dan pembaruan berkala, misalnya sistem informasi akademik atau portal layanan publik.

#### **b. Kanban**

Kanban berfokus pada visualisasi alur kerja melalui papan (Kanban *board*) yang menampilkan status tugas: *To Do*, *In Progress*, dan *Done*. Pendekatan ini menekankan pengendalian jumlah pekerjaan agar tidak terjadi kelebihan beban pada tim. Kanban cocok untuk proyek yang membutuhkan pemantauan progres *real-time* dan penyesuaian prioritas secara fleksibel.

#### **c. Extreme Programming (XP)**

XP adalah pendekatan Agile yang menekankan kualitas kode, kolaborasi intensif, dan pengujian berkelanjutan. Dalam penelitian (Hilmi et al., 2025), XP terbukti efektif dalam proyek kecil hingga menengah yang memiliki perubahan kebutuhan sangat dinamis. XP sangat cocok digunakan pada proyek *web* yang membutuhkan kestabilan fungsi dan kecepatan pengembangan.

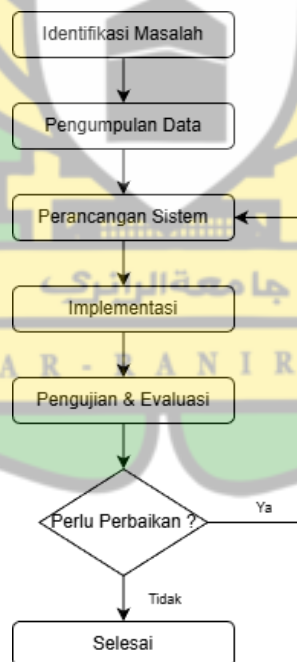


## BAB III METODOLOGI PENELITIAN

### 3.1 Tahapan Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode Agile (*Scrum*). Agile merupakan metode pengembangan perangkat lunak yang bersifat iteratif dan fleksibel, di mana Tahapan penelitian dilakukan menggunakan pendekatan Agile secara iteratif. Apabila pada tahap pengujian dan evaluasi ditemukan kekurangan, maka dilakukan perbaikan dengan kembali ke tahap perancangan dan implementasi hingga sistem memenuhi kebutuhan pengguna.

Menurut (Rakhmawati, 2024), penerapan metode Agile dalam pengembangan sistem informasi mampu meningkatkan kualitas perangkat lunak karena adanya proses perbaikan berkelanjutan berdasarkan umpan balik pengguna. Gambar 3.1 menampilkan diagram alur tahapan penelitian yang digunakan dalam penelitian ini.



Gambar 3. 1 Tahapan Penelitian

### 3.1.1 Identifikasi Masalah

Tahap indentifikasi masalah merupakan langkah awal yang dilakukan untuk memperoleh gambaran yang jelas mengenai kondisi dan kebutuhan yang ada di lapangan. Pada penelitian ini, proses indentifikasi dilakukan di beberapa Pesantren yang ada di Banda Aceh. Informasi tentang lembaga pendidikan, termasuk pesantren, menjadi kebutuhan penting bagi masyarakat. Namun, di Kota Banda Aceh, banyak calon santri dan orang tua mengalami kesulitan dalam memperoleh informasi yang akurat dan terverifikasi mengenai pesantren. Sebagian besar pesantren belum memiliki *platform digital* yang terintegrasi untuk menampilkan profil, program pendidikan, fasilitas, serta ulasan dari pengguna atau alumni. Akibatnya, proses pengambilan keputusan untuk memilih pesantren yang sesuai sering kali hanya didasarkan pada rekomendasi mulut ke mulut, bukan pada data objektif dan pengalaman pengguna lain. Kondisi ini menunjukkan adanya kesenjangan informasi yang perlu diatasi melalui sistem berbasis *web*.

Selain itu, media sosial sering kali digunakan oleh masyarakat untuk mencari informasi pesantren, tetapi *platform* tersebut tidak memiliki mekanisme verifikasi dan penyajian ulasan yang terstruktur. Ulasan tanpa sistem moderasi dan validasi pengguna dapat menimbulkan informasi palsu atau bias. Hal ini berdampak pada menurunnya kepercayaan pengguna terhadap informasi yang beredar. Oleh karena itu, dibutuhkan sistem ulasan berbasis *web* yang memiliki autentikasi pengguna, sehingga hanya pengguna yang benar-benar pernah berinteraksi dengan pesantren dapat memberikan ulasan.

Dari sisi pengelola pesantren, keterbatasan *platform digital* juga menjadi kendala dalam memperkenalkan lembaganya secara luas. Sebagian pesantren masih menggunakan metode konvensional seperti brosur dan papan pengumuman untuk menyebarkan informasi. Sistem informasi berbasis *web* mampu membantu lembaga pendidikan dalam meningkatkan transparansi, kredibilitas, serta jangkauan publikasi. Dengan demikian, penerapan sistem berbasis Laravel dapat menjadi solusi efektif untuk menampilkan informasi dan ulasan pesantren secara terintegrasi dan interaktif.

Selain permasalahan teknis, proses pengembangan sistem juga sering menghadapi kendala dalam hal fleksibilitas terhadap perubahan kebutuhan pengguna. Oleh sebab itu, penggunaan metode Agile, menjadi relevan dalam konteks penelitian ini. Agile memungkinkan pengembang dan pengguna berkolaborasi secara berkelanjutan melalui siklus iteratif, sehingga setiap perubahan kebutuhan dapat segera diakomodasi. Dengan demikian, sistem *web* ulasan dan informasi pesantren yang dibangun dapat benar-benar sesuai dengan kebutuhan pengguna dan mudah dikembangkan di masa mendatang.

Berdasarkan analisis tersebut, dapat disimpulkan bahwa pengembangan *website* ulasan dan informasi pesantren berbasis Laravel dengan metode Agile merupakan solusi yang relevan dan efektif. Sistem ini tidak hanya membantu masyarakat dalam mendapatkan informasi terpercaya, tetapi juga mendukung pesantren dalam mempromosikan lembaga mereka secara profesional.

### **3.1.2 Pengumpulan Data**

Dalam penelitian ini, teknik pengumpulan data dilakukan melalui dua metode utama, yaitu observasi langsung dan penyebaran kuesioner (angket). Metode ini dipilih untuk memperoleh data yang akurat dan relevan terkait kebutuhan pengguna, kondisi eksisting di lapangan, serta tanggapan terhadap sistem yang dikembangkan.

Metode observasi dilakukan dengan cara mengamati secara langsung kondisi di beberapa pesantren di Kota Banda Aceh, khususnya terkait sistem informasi yang telah digunakan, cara pengelolaan data santri, serta media publikasi yang tersedia. Observasi merupakan metode pengumpulan data yang efektif untuk memahami proses dan permasalahan yang terjadi di lingkungan pengguna sebelum sistem dikembangkan. Melalui observasi ini, peneliti menemukan bahwa sebagian besar pesantren belum memiliki sistem digital terintegrasi untuk menampilkan profil lembaga maupun ulasan pengguna. Hasil observasi ini menjadi dasar untuk menentukan fitur-fitur utama dalam *web*, seperti halaman profil pesantren, sistem ulasan, dan rating pengguna.

Selain observasi, pengumpulan data juga dilakukan dengan menyebarkan kuesioner kepada calon pengguna sistem, yaitu santri, wali santri, serta masyarakat umum yang mencari informasi pesantren di Banda Aceh. Kuesioner merupakan instrumen pengumpulan data yang berguna untuk memperoleh tanggapan pengguna terhadap rancangan sistem, terutama dalam aspek kebutuhan informasi, kemudahan penggunaan, dan kejelasan tampilan. Pertanyaan dalam kuesioner mencakup persepsi responden terhadap pentingnya ulasan pesantren, tingkat kepercayaan terhadap informasi digital, serta preferensi fitur yang diharapkan dalam sistem *web*. Data dari kuesioner ini kemudian diolah secara deskriptif untuk membantu peneliti dalam menyesuaikan desain sistem dengan kebutuhan pengguna.

Berdasarkan hasil observasi dan kuesioner, diperoleh gambaran yang jelas mengenai kebutuhan pengguna serta masalah yang dihadapi pesantren dalam penyebaran informasi. Kedua metode ini memberikan dasar empiris dalam proses perancangan sistem, sehingga pengembangan *web* ulasan dan informasi pesantren dapat dilakukan secara terarah, sesuai kebutuhan pengguna, dan memiliki tingkat relevansi yang tinggi di lapangan.

Dengan demikian, data yang diperoleh dari metode pengumpulan ini menjadi dasar yang kuat dalam perancangan dan juga pengembangan *web* ulasan dan informasi pesantren pada penelitian ini.

### 3.1.3 Perancangan sistem

Perancangan sistem merupakan tahap penting dalam proses pengembangan aplikasi *web* ulasan dan informasi pesantren. Pada tahap ini, dilakukan proses perencanaan dan penyusunan komponen sistem agar dapat bekerja secara terstruktur sesuai dengan kebutuhan pengguna. Tujuan utama dari perancangan sistem adalah untuk menghasilkan rancangan yang jelas dan terarah sebelum sistem diimplementasikan menggunakan *framework* Laravel. Perancangan dilakukan berdasarkan hasil analisis kebutuhan yang diperoleh dari observasi dan kuesioner. Informasi yang dikumpulkan tersebut menjadi dasar dalam menentukan fitur-fitur yang akan dikembangkan, seperti halaman profil pesantren, sistem ulasan dan *rating*, pencarian lokasi pesantren, serta fitur *admin* untuk pengelolaan data. Dalam proses

perancangan ini juga digunakan Filament sebagai framework admin panel pada Laravel yang berfungsi untuk mempermudah pengelolaan data pesantren, ulasan, pengguna, dan informasi pendukung lainnya. Dengan adanya rancangan yang matang dan dukungan Filament, pengembangan sistem menjadi lebih efisien, terarah, dan mudah dalam proses pengelolaan sistem oleh *admin*, yaitu:

**a. Perancangan Use Case Diagram**

*Use case diagram* menggambarkan interaksi antara pengguna dengan sistem. Melalui diagram ini, dapat diketahui siapa saja aktor yang terlibat (seperti *admin* dan pengguna umum) serta fungsi-fungsi apa saja yang dapat dilakukan oleh masing-masing aktor.



Gambar 3. 2 Use Case Diagram

Diagram *use case* diatas menggambarkan interaksi antara pengguna dengan sistem *web* informasi dan ulasan pesantren. Diagram ini menjelaskan fungsi-fungsi utama yang dapat dilakukan oleh dua aktor dalam sistem, yaitu *Admin* dan *User*, serta hubungan antara masing-masing aktor dengan fitur-fitur yang tersedia.

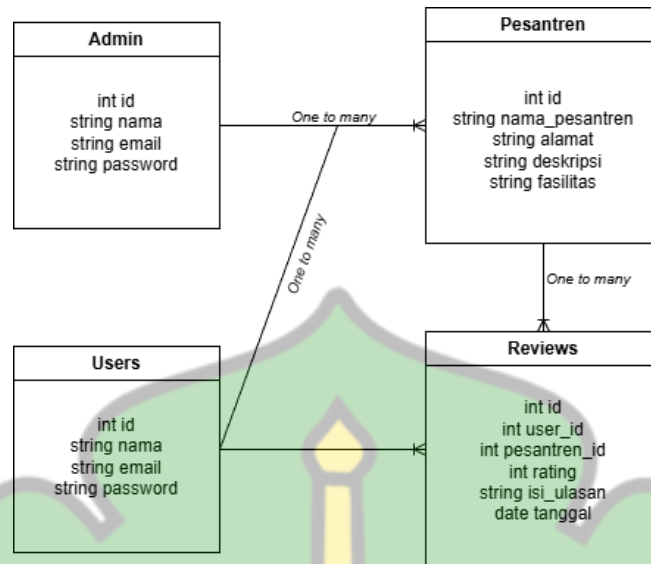
Pada sisi *Admin*, terdapat beberapa fungsi yang dapat dilakukan. *Admin* dapat melakukan *Login* untuk masuk ke dalam sistem menggunakan akun yang terdaftar, serta *Logout* untuk keluar dari sistem dengan aman setelah selesai menggunakan aplikasi. Fungsi utama admin adalah Kelola Konten, yaitu mengelola data informasi pesantren, mengatur rating, serta memantau ulasan yang diberikan oleh pengguna. Dengan fungsi ini, *admin* memiliki peran penting dalam memastikan bahwa seluruh data pesantren dan ulasan yang ditampilkan dalam sistem valid, akurat, serta sesuai dengan ketentuan.

Sementara itu, *User* atau pengguna umum memiliki peran sebagai pihak yang mengakses informasi pesantren dan memberikan umpan balik berupa ulasan. Sebelum dapat mengakses fitur utama, pengguna dapat melakukan *Register* untuk membuat akun baru dan *Login* untuk masuk ke dalam sistem. Setelah berhasil masuk, pengguna dapat Melihat Konten berupa informasi tentang pesantren, rating, serta ulasan dari pengguna lain. Selain itu, pengguna juga dapat memberikan *reviews* atau menambahkan ulasan pribadi terhadap pesantren tertentu yang pernah dikunjungi atau dikenal, lengkap dengan nilai rating dan komentar.

Secara keseluruhan, diagram *use case* ini menunjukkan bagaimana sistem *web* informasi dan ulasan pesantren bekerja dalam memfasilitasi interaksi dua aktor utama, yaitu *Admin* dan *User*. *Admin* bertugas menjaga validitas data dan konten, sementara *User* berperan aktif dalam memberikan ulasan serta penilaian terhadap pesantren. Hubungan antar fungsi dalam diagram ini menciptakan alur yang jelas dan efisien dalam pengelolaan serta penyampaian informasi pesantren berbasis *web*.

## **b. Perancangan *Database***

Tahap ini meliputi pembuatan tabel dan relasi antar tabel yang digunakan untuk menyimpan data pesantren, pengguna, ulasan, dan *rating*. Rancangan *database* dibuat menggunakan model relasional agar data dapat dikelola dengan efisien dan terhindar dari duplikasi.



Gambar 3. 3 ERD Database

Perancangan basis data pada sistem ulasan pesantren ini bertujuan untuk mengelola dan menghubungkan seluruh data secara terstruktur agar proses penyimpanan, pengambilan, serta pengolahan data dapat dilakukan dengan efisien. Dalam sistem ini terdapat empat entitas utama yaitu *Admin*, *Users*, *Pesantren*, dan *Reviews* yang saling berhubungan satu sama lain.

Entitas *Admin* berperan sebagai pengelola utama sistem yang memiliki hak akses penuh terhadap data pesantren dan ulasan pengguna. *Admin* dapat menambah, mengubah, maupun menghapus data pesantren. Atribut yang dimiliki oleh *admin* meliputi *id*, *nama*, *email*, dan *password* yang berfungsi sebagai identitas dan informasi *login*. Satu *admin* dapat mengelola banyak data pesantren sehingga hubungan antara entitas *Admin* dan *Pesantren* bersifat *one to many*.

Entitas *Users* merupakan pengguna umum yang dapat memberikan ulasan dan penilaian terhadap pesantren. Setiap pengguna memiliki atribut *id*, *nama*, *email*, dan *password* sebagai identitas pribadi untuk melakukan autentikasi dalam sistem. Satu pengguna dapat memberikan banyak ulasan terhadap beberapa pesantren, sehingga relasi antara *Users* dan *Reviews* juga bersifat *one to many*.

Entitas Pesantren digunakan untuk menyimpan informasi lengkap mengenai pesantren yang terdaftar di sistem, seperti nama\_pesantren, alamat, deskripsi. Data pesantren ini dikelola oleh *admin* dan menjadi objek yang diulas oleh pengguna. Karena satu pesantren dapat memiliki banyak ulasan dari pengguna yang berbeda, maka hubungan antara Pesantren dan *Reviews* bersifat *one to many*.

Selanjutnya, entitas *Reviews* berfungsi untuk menyimpan data ulasan atau penilaian yang diberikan oleh pengguna terhadap pesantren tertentu. Atribut pada entitas ini terdiri dari *id*, *user\_id*, *pesantren\_id*, *isi\_ulasan*, *rating*, dan *tanggal*. Ulasan ini terhubung ke pengguna dan pesantren melalui *foreign key* yaitu *user\_id* dan *pesantren\_id*. Dengan adanya entitas ini, sistem dapat menampilkan berbagai opini dan penilaian yang diberikan oleh pengguna terhadap masing-masing pesantren.

### c. Perancangan Antarmuka (UI/UX)

Pada tahap ini, dilakukan pembuatan rancangan tampilan (*mockup*) agar sistem mudah digunakan oleh pengguna. Antarmuka dirancang dengan prinsip sederhana, informatif, dan responsif sehingga pengguna dapat mengakses informasi pesantren dengan cepat dan nyaman, baik melalui komputer maupun perangkat seluler.

Setiap halaman memiliki fungsi dan peran berbeda namun tetap terhubung secara logis dalam satu alur sistem *web* ulasan pesantren. Antarmuka utama yang dirancang terdiri dari 5 *wireframe* yaitu *Login*, *Register*, *Home*, *Konten*, dan *Reviews*.

- *Wireframe login page*

Alur *wireframe website* ini dimulai dari halaman *Login*, yang berfungsi sebagai gerbang utama bagi pengguna untuk masuk ke sistem. Pada halaman ini terdapat logo di bagian atas, *form login* yang berisi kolom *email* dan *password*, tombol “*Login*”, serta tautan “*Create New*” bagi pengguna yang belum memiliki akun. Pengguna yang telah terdaftar dapat langsung

mengisi *form* dan menekan tombol *login* untuk diarahkan ke halaman utama, sedangkan pengguna baru dapat memilih opsi “*Create New*” untuk menuju halaman pendaftaran. Untuk tampilan Wireframe nya dapat dilihat pada gambar 3.4 berikut.

Gambar 3. 4 *Wireframe Login Page*

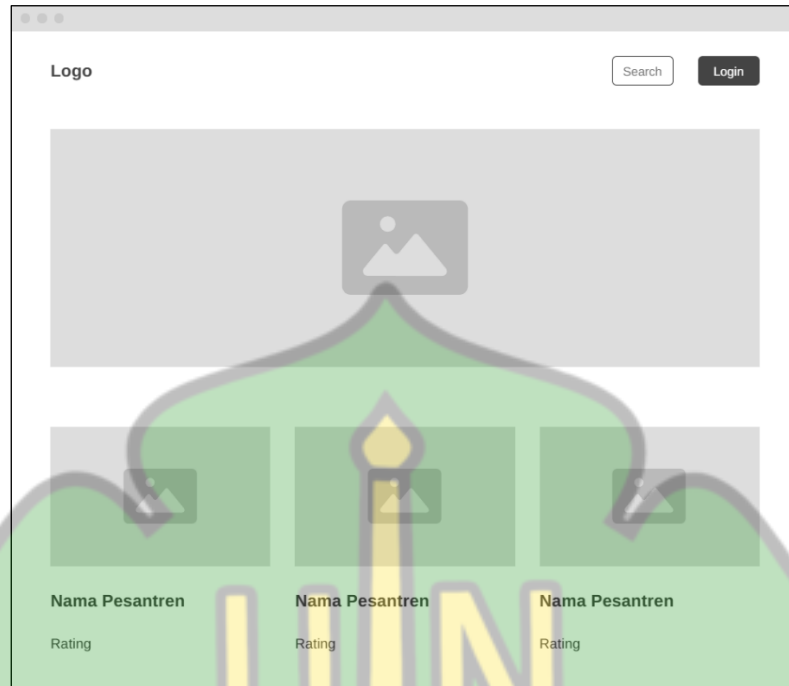
- *Wireframe Register Page*

Selanjutnya, pengguna akan diarahkan ke halaman *Register (Sign Up)* yang berfungsi untuk mendaftarkan akun baru. Pada halaman ini terdapat kolom pengisian seperti *username, email, dan password*, serta tombol “*Sign Up*” untuk menyimpan data pengguna baru ke sistem. Di bagian bawah juga terdapat tautan menuju halaman *login* bagi pengguna yang sudah memiliki akun. Setelah proses pendaftaran selesai, pengguna akan kembali diarahkan ke halaman *login* untuk melakukan masuk ke sistem. Untuk tampilan *Wireframe* nya dapat dilihat pada gambar 3.5 berikut.

Gambar 3. 5 *Wireframe Register Page*

- *Wireframe Home Page*

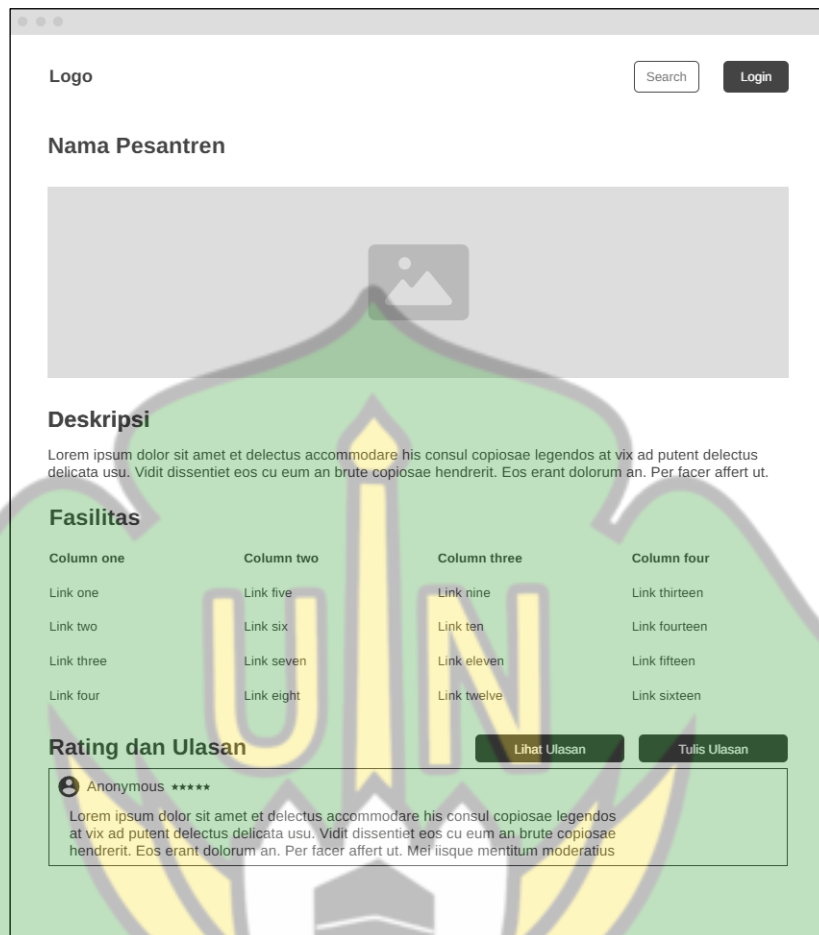
Setelah berhasil *login*, pengguna akan tiba di halaman *Home* (Beranda). Halaman ini menampilkan berbagai informasi pesantren dengan tampilan sederhana dan mudah diakses. Di bagian atas terdapat logo, kolom pencarian, dan tombol login atau profil pengguna. Terdapat pula banner utama di bagian tengah yang dapat menampilkan informasi penting, serta beberapa kartu pesantren di bawahnya yang berisi foto, nama pesantren, dan *rating*. Untuk tampilan *Wireframe* nya dapat dilihat pada gambar 3.6 berikut.



Gambar 3. 6 *Wireframe Home Page*

- *Wireframe Konten Page*

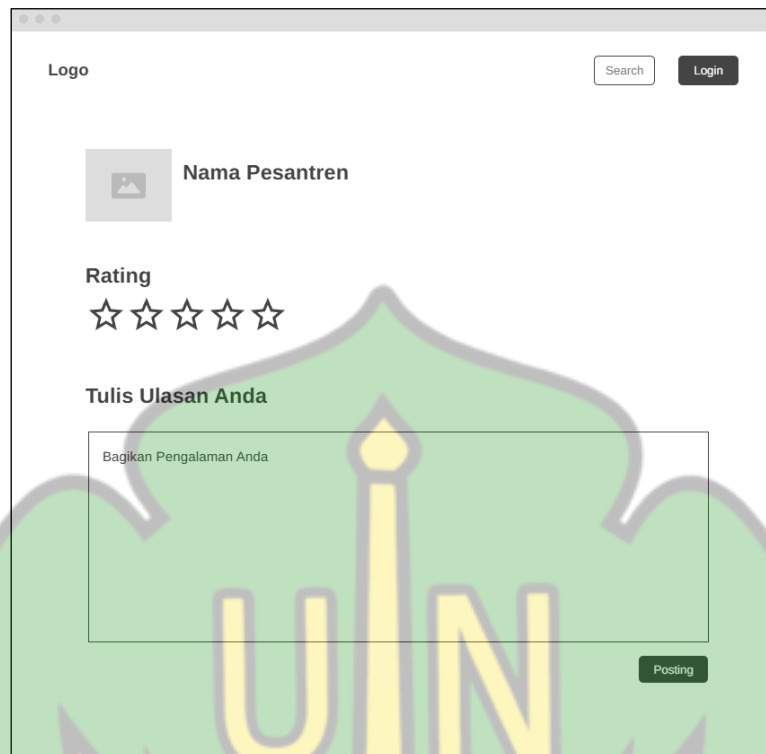
Ketika pengguna memilih salah satu pesantren, mereka akan diarahkan ke halaman Konten Pesantren, yang berisi detail lengkap seperti deskripsi, fasilitas, dan informasi umum pesantren. Bagian bawah halaman ini juga menampilkan fitur *Rating* dan Ulasan, di mana pengguna dapat melihat pengalaman pengguna lain atau menulis ulasan mereka sendiri. Untuk tampilan *Wireframe* nya dapat dilihat pada gambar 3.7 berikut.



Gambar 3. 7 Wireframe Content Page

- *Wireframe Reviews Page*

Terakhir, terdapat halaman *Review* atau *Ulasan*, di mana pengguna dapat memberikan penilaian berupa bintang dan menulis pengalaman mereka secara langsung melalui kolom teks yang disediakan. Setelah menekan tombol “Posting”, ulasan tersebut akan tersimpan dan dapat dilihat oleh pengguna lain di halaman profil pesantren tersebut. Dengan alur ini, pengguna dapat dengan mudah *login*, mendaftar, mencari pesantren, melihat detailnya, dan memberikan ulasan secara interaktif dalam satu sistem yang terintegrasi. Untuk tampilan *Wireframe* nya dapat dilihat pada gambar 3.8 berikut.



Gambar 3. 8 Wireframe Reviews Page

### 3.1.4 Implementasi Sistem

Tahapan implementasi dalam penelitian ini dilakukan secara bertahap untuk memastikan bahwa sistem *web* ulasan dan informasi pesantren yang dikembangkan dapat berjalan sesuai dengan kebutuhan pengguna. Tahap pertama adalah persiapan lingkungan pengembangan, di mana dilakukan instalasi dan konfigurasi perangkat lunak yang dibutuhkan, seperti *framework* Laravel sebagai fondasi pengembangan aplikasi, *server* lokal menggunakan XAMPP untuk menjalankan layanan Apache dan MySQL, serta *text editor* seperti Visual Studio Code sebagai lingkungan penulisan kode program. Selain itu, dilakukan konfigurasi basis data dan struktur folder proyek agar sistem dapat berjalan secara terorganisir dan mudah dikelola.

Tahap kedua adalah implementasi basis data (*database implementation*). Pada tahap ini dilakukan pembuatan struktur *database* yang berfungsi sebagai tempat penyimpanan seluruh data, seperti data pengguna, pesantren, ulasan, dan rating. *Database* dirancang berdasarkan *Entity Relationship Diagram* (ERD) yang sebelumnya telah dibuat pada tahap perancangan. Setiap entitas memiliki relasi

tertentu agar data antar tabel dapat saling terhubung, misalnya relasi antara tabel pesantren dengan ulasan yang bersifat *one-to-many*. Selanjutnya, digunakan fitur *migration* pada Laravel untuk membuat struktur tabel secara otomatis dan menjaga konsistensi data selama proses pengembangan berlangsung.

Tahap ketiga adalah implementasi fungsionalitas sistem, yaitu proses pengkodean (*coding*) untuk mengubah rancangan menjadi aplikasi *web* yang dapat dijalankan. Pada tahap ini, dibuat modul untuk *admin* dan pengguna. *Admin* dapat *login*, *logout*, serta mengelola data pesantren, *rating*, dan ulasan melalui fitur CRUD (*Create, Read, Update, Delete*). Sedangkan pengguna umum dapat melakukan registrasi, *login*, melihat informasi pesantren, serta memberikan ulasan dan *rating*.

Dengan pelaksanaan tahap implementasi yang terencana dan juga terstruktur, diharapkan sistem informasi dan ulasan *web* pesantren dapat dikembangkan secara efektif dan dapat memenuhi kebutuhan masyarakat.

### **3.1.5 Evaluasi**

Tahap evaluasi merupakan proses untuk menilai sejauh mana sistem *web* ulasan dan informasi pesantren yang telah dikembangkan dapat berfungsi dengan baik dan memenuhi kebutuhan pengguna. Evaluasi dilakukan setelah sistem selesai diimplementasikan agar dapat diketahui tingkat keberhasilan serta area yang masih perlu diperbaiki. Tujuan dari tahap ini adalah untuk memastikan bahwa seluruh komponen sistem bekerja sesuai dengan rancangan, mudah digunakan, dan memberikan hasil sesuai harapan pengguna.

Dalam proses evaluasi, dilakukan dua jenis penilaian, yaitu evaluasi teknis dan evaluasi pengguna. Evaluasi teknis berfokus pada aspek fungsional sistem seperti kelancaran proses *login*, pengelolaan data pesantren, pengiriman ulasan, serta tampilan informasi di halaman utama. Pengujian ini dilakukan menggunakan metode *Black Box Testing*, di mana setiap fitur diuji berdasarkan *input dan output* tanpa melihat kode program. Sementara itu, evaluasi pengguna dilakukan melalui uji coba langsung kepada beberapa pengguna, seperti santri, calon santri, dan pengelola

pesantren, untuk menilai kemudahan penggunaan (*usability*), kejelasan informasi, serta tampilan antarmuka sistem.

Hasil dari tahap evaluasi digunakan untuk memperbaiki kekurangan yang ditemukan selama pengujian. Umpan balik dari pengguna sangat berperan penting dalam penyempurnaan sistem, terutama untuk meningkatkan pengalaman pengguna (*user experience*) dan keakuratan informasi yang ditampilkan. Setelah proses perbaikan selesai, dilakukan evaluasi ulang secara singkat untuk memastikan perubahan yang diterapkan sudah berjalan dengan baik. Dengan adanya tahap evaluasi ini, sistem yang dikembangkan diharapkan dapat digunakan secara optimal, memiliki performa yang stabil, serta benar-benar bermanfaat bagi masyarakat dan pihak pesantren dalam penyebaran informasi serta pengumpulan ulasan secara digital.

### **3.2 Waktu dan Lokasi Penelitian**

Waktu penelitian ini dilaksanakan mulai dari bulan Agustus hingga Oktober 2025. Kegiatan meliputi pengumpulan data, perancangan, implementasi, pengujian, dan evaluasi sistem. Untuk saat ini fokus Lokasi penelitian dilakukan di Kota Banda Aceh, dengan objek penelitian beberapa pesantren yang menjadi sumber data dan uji coba sistem.

### **3.3 Alat dan Bahan**

Alat yang diperlukan dalam melaksanakan penelitian ini terdiri dari perangkat keras (*Hardware*) dan perangkat lunak (*Software*). Analisis mengenai alat dan spesifikasi sistem yang dibutuhkan untuk penelitian ini meliputi :

#### **3.3.1 Perangkat Keras (*Hardware*)**

Perangkat keras yang digunakan berupa satu unit Laptop HP Victus Gaming 15 dengan spesifikasi yang tertera pada tabel 3.1.

Tabel 3. 1 Spesifikasi Perangkat Keras

Komponen	Spesifikasi
----------	-------------

Processor	AMD Ryzen 5 5600H
Ram	16 GB DDR4
Storage	512 GB SSD NVME
Graphic Processing Unit	AMD Radeon Graphics

### 3.3.2 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam proses implementasi berupa sistem operasi Microsoft Windows Home Single Language 64- bit, dan menggunakan *tools* seperti, Visual Studio Code, XAMPP, Visual Paradigm, draw.io, Figma, Chrome. *Tools* tersebut akan digunakan di dalam penelitian yang akan penulis pakai, dapat dilihat pada tabel 3.2.

Tabel 3. 2 Spesifikasi Perangkat Lunak

Perangkat Lunak	Version
Microsoft Windows Home Single Language 64- bit	10.0 Build 26100
XAMPP	V3.3.0
Visual Paradigm	17.2
draw.io	V28.2.5
Figma	125.4.9
Chrome	141.0.7390.54
Laravel	11.0
Filament	4.5

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Implementasi

Tahap Tahap implementasi sistem merupakan tahap penerapan seluruh hasil perancangan yang telah dilakukan pada bab sebelumnya ke dalam bentuk aplikasi website yang dapat dijalankan secara nyata. Implementasi ini bertujuan untuk menghasilkan sistem *web* ulasan dan informasi pesantren yang bersifat fungsional, mudah digunakan, serta mampu memenuhi kebutuhan pengguna.

Lingkungan implementasi sistem dikembangkan menggunakan *framework* Laravel versi 11 sebagai fondasi utama aplikasi. Laravel dipilih karena memiliki struktur pengembangan yang terorganisir, mendukung arsitektur *Model View Controller (MVC)*, serta menyediakan berbagai fitur bawaan yang mempermudah pengembangan sistem informasi berbasis *web*. Untuk pengelolaan data, sistem menggunakan basis data MySQL yang dioperasikan melalui server lokal XAMPP selama tahap pengembangan.

Selain itu, Filament versi 4.5 digunakan sebagai *admin* panel untuk mempermudah pengelolaan data pesantren dan ulasan. Penggunaan Filament memungkinkan *admin* mengelola data melalui antarmuka grafis yang intuitif tanpa harus berinteraksi langsung dengan basis data atau kode program. Arsitektur sistem secara keseluruhan dirancang agar mudah dikembangkan dan dipelihara di masa mendatang.

##### 4.1.1 Implementasi Sistem

Implementasi sistem merupakan tahap penerapan hasil perancangan ke dalam bentuk aplikasi yang dapat dijalankan dan digunakan oleh pengguna. Pada penelitian ini, sistem Pesantrenly diimplementasikan sebagai aplikasi web menggunakan *framework* Laravel dengan arsitektur *Model-View-Controller (MVC)* untuk memisahkan logika bisnis, pengolahan data, dan tampilan antarmuka. Proses

implementasi mencakup pengembangan sisi *backend* untuk pengelolaan data pesantren, ulasan, dan hak akses pengguna, serta sisi *frontend* untuk menyajikan informasi kepada pengguna secara interaktif.

Selain itu, sistem dilengkapi dengan Filament sebagai *admin* panel yang memudahkan *administrator* dalam melakukan manajemen data pesantren dan moderasi ulasan secara terstruktur dan efisien. Seluruh fitur utama, seperti pencarian pesantren, penampilan detail pesantren, pemberian ulasan dan *rating*, serta pengelolaan data oleh *admin*, diimplementasikan sesuai dengan kebutuhan sistem. Dengan integrasi komponen *backend* dan *frontend* yang baik, sistem Pesantrenly mampu berjalan secara stabil, aman, dan mendukung tujuan penelitian dalam menyediakan informasi pesantren yang terpusat dan mudah diakses.

## 1. Routing

Routing merupakan mekanisme pada aplikasi *web* yang berfungsi untuk mengatur alur permintaan pengguna menuju fungsi tertentu di dalam sistem. Pada Pesantrenly, *routing* digunakan untuk menghubungkan *URL* yang diakses pengguna dengan *controller* yang memproses logika aplikasi, seperti menampilkan halaman beranda, melakukan pencarian pesantren, dan menampilkan detail pesantren. Dengan penerapan *routing* yang terstruktur, sistem dapat mengelola navigasi secara rapi, aman, dan sesuai dengan alur kerja yang telah dirancang.

```
Route::get(uri: '/', action: [PesantrenController::class, 'index'])
->name(name: 'welcome');

Route::get(uri: '/search', action: [PesantrenController::class, 'search'])
->name(name: 'pesantren.search');

Route::get(uri: '/pesantren/{pesantren}', action: [PesantrenController::class, 'show'])
->name(name: 'pesantren.show');
```

Gambar 4. 1 Kode Program Routing web.php

Kode *program routing* pada gambar 4.1 berikut merupakan bagian dari implementasi sistem yang berfungsi untuk mengatur alur navigasi pengguna pada aplikasi Pesantrenly. *Routing* ini menghubungkan setiap *URL* yang diakses pengguna dengan fungsi tertentu pada *controller*, sehingga sistem

dapat menampilkan halaman yang sesuai dan memproses permintaan pengguna secara terstruktur. Penjelasan tiap bagian kode dijabarkan sebagai berikut:

- `Route::get('/', [PesantrenController::class, 'index'])` Digunakan untuk mengatur route halaman beranda yang akan menampilkan daftar pesantren melalui *method index* pada *PesantrenController*.  
`->name('welcome')` Memberikan nama pada *route* beranda agar mudah dipanggil kembali dalam aplikasi, misalnya pada *link* atau *redirect*.
- `Route::get('/search', [PesantrenController::class, 'search'])` Digunakan untuk menangani permintaan pencarian pesantren berdasarkan kata kunci yang dimasukkan oleh pengguna.  
`->name('pesantren.search')` Memberikan nama *route* pencarian pesantren untuk mempermudah pemanggilan dalam sistem.
- `Route::get('/pesantren/{pesantren}', [PesantrenController::class, 'show'])` Digunakan untuk menampilkan halaman *detail* pesantren berdasarkan data pesantren yang dipilih.  
`->name('pesantren.show')` Memberikan nama *route detail* pesantren agar dapat diakses secara dinamis dan terstruktur.

## 2. Controller (Logika Fitur)

*Controller* merupakan komponen pada aplikasi Pesantrenly yang berfungsi untuk mengelola logika aplikasi dan menjembatani interaksi antara *request* pengguna, model (*database*), dan *view* (tampilan). *Controller* menerima permintaan dari pengguna, memproses data yang diperlukan, kemudian meneruskannya ke tampilan yang sesuai sehingga sistem dapat berjalan secara terstruktur sesuai arsitektur *Model–View–Controller* (MVC).

### a. Method index pesantren controller

```
public function index(): View
{
    $pesantrens = Pesantren::latest()->get();
    return view(view: 'welcome', data: compact(var_name: 'pesantrens'));
}
```

Gambar 4. 2 Kode Program method index *PesantrenController*

Kode routing Pada gambar 4.2 merupakan bagian dari *PesantrenController* yang berfungsi untuk menampilkan halaman utama aplikasi. Method ini bertanggung jawab dalam mengambil dan menampilkan daftar pesantren kepada pengguna. Penjelasan tiap bagian kode adalah sebagai berikut :

- `public function index(): View` Mendefinisikan *method index* yang mengembalikan tampilan (*view*). *Method* ini menjadi titik awal untuk menampilkan halaman beranda.
- `$pesantrens = Pesantren::latest()->get();` Digunakan untuk mengambil seluruh data pesantren dari *database* dan mengurutkannya berdasarkan data terbaru.
- `return view('welcome', compact('pesantrens'));` Mengirimkan data pesantren ke tampilan *welcome.blade.php* agar dapat ditampilkan dalam bentuk daftar pada halaman utama aplikasi.

#### b. *Method index show pesantrencontroller*

```
public function show(Pesantren $pesantren): View
{
    $pesantren->load(relations: ['reviews.user']);
    return view(view: 'pesantren.show', data: compact(var_name: 'pesantren'));
}
```

Gambar 4. 3 Kode *Program method show PesantrenController*

Kode berikut merupakan bagian dari implementasi *PesantrenController* yang berfungsi untuk menampilkan halaman detail pesantren. *Method* ini menangani permintaan pengguna ketika memilih salah satu pesantren, sekaligus memastikan data yang ditampilkan sudah lengkap beserta ulasan terkait. Penjelasan tiap bagian kode dijabarkan sebagai berikut :

- `public function show(Pesantren $pesantren): View` *Method show* menerima parameter berupa objek *Pesantren* yang diperoleh melalui *route model binding*. *Method* ini digunakan untuk menampilkan *detail* data satu pesantren.

- `$pesantren->load(['reviews.user']);` Digunakan untuk memuat relasi data ulasan (*reviews*) beserta data pengguna (*user*) yang memberikan ulasan. Hal ini bertujuan agar seluruh informasi terkait pesantren dapat ditampilkan secara lengkap dalam satu halaman.
- `return view('pesantren.show', compact('pesantren'));` Mengirimkan data pesantren ke tampilan *pesantren.show* agar dapat ditampilkan dalam bentuk antarmuka detail kepada pengguna.

### c. Method index search pesantrencontroller

```
// Normalisasi input
$q = preg_replace(pattern: '/[^\a-z0-9\s]/i', replacement: ' ', subject: $q);
$q = preg_replace(pattern: '/\s+/', replacement: ' ', subject: $q);
```

Gambar 4. 4 Kode Program method search PesantrenController

Potongan kode pada gambar 4.4 di atas merupakan bagian inti dari implementasi fitur pencarian pesantren pada sisi *backend*. Kode ini berfungsi untuk memastikan input pengguna diproses secara aman serta melakukan pencarian data pesantren secara fleksibel dan relevan. Penjelasannya adalah sebagai berikut :

- `$q = preg_replace('/[^\a-z0-9\s]/i', ' ', $q);` Berfungsi untuk menghapus karakter yang tidak termasuk huruf, angka, dan spasi dari *input* pencarian. Hal ini dilakukan agar kata kunci yang diproses hanya berisi karakter yang valid dan tidak mengganggu proses pencarian data.
- `$q = preg_replace('/\s+/', ' ', $q);` Digunakan untuk merapikan spasi berlebih dengan menggantinya menjadi satu spasi saja, sehingga format input menjadi lebih konsisten dan mudah diproses oleh sistem.

```
$pesantren = Pesantren::where(column: function (Pesantren $query) use ($keywords): void {
    foreach ($keywords as $word) {
        $query->where(column: function (Pesantren $q2) use ($word): void {
            $q2->where(column: 'nama_pesantren', operator: 'LIKE', value: "%{$word}%")
                ->orWhere(column: 'alamat', operator: 'LIKE', value: "%{$word}%");
        });
    }
})->get();
```

Gambar 4. 5 Kode Program method search PesantrenController

Potongan kode pada gambar 4.5 merupakan bagian dari implementasi logika pencarian pesantren pada sisi *backend* aplikasi. Kode ini dirancang untuk menelusuri data pesantren berdasarkan kata kunci yang dimasukkan pengguna, sehingga hasil pencarian yang ditampilkan bersifat fleksibel dan relevan. Penjelasan tiap bagian kode dijabarkan sebagai berikut :

- `Pesantren::where(function ($query) use ($keywords) { ... })` Digunakan untuk membangun *query* pencarian data pesantren dengan kondisi dinamis berdasarkan kata kunci yang dimasukkan pengguna.  
`foreach ($keywords as $word)`  
Melakukan perulangan terhadap setiap kata kunci hasil pemecahan *input* pencarian, sehingga sistem dapat melakukan pencarian berbasis multi-kata.
- `$query->where(function ($q2) use ($word) { ... })` Digunakan untuk mengelompokkan kondisi pencarian agar setiap kata kunci dicocokkan secara terstruktur.
- `$q2->where('nama_pesantren', 'LIKE', "%{$word}%")` Mencari data pesantren yang memiliki kecocokan kata kunci pada kolom nama pesantren.  
`->orWhere('alamat', 'LIKE', "%{$word}%")` Menambahkan kondisi alternatif pencarian pada kolom alamat, sehingga hasil pencarian menjadi lebih luas dan relevan.  
`->get()` Menjalankan *query* dan mengambil seluruh data pesantren yang sesuai dengan kriteria pencarian.

#### d. Method index store reviewcontroller

```
public function store(Request $request, Pesantren $pesantren): RedirectResponse
{
    $request->validate(rules: [
        'rating' => 'required|integer|min:1|max:5',
        'ulasan' => 'required|string|min:5',
    ]);
}
```

Gambar 4. 6 Kode Program method store ReviewController

Potongan kode pada gambar 4.6 merupakan bagian dari *method store* yang digunakan untuk memproses penyimpanan ulasan dan *rating* pengguna pada sistem. Penjelasan tiap bagian kode adalah sebagai berikut:

- `public function store(Request $request, Pesantren $pesantren): RedirectResponse` *Method store* berfungsi untuk menerima data ulasan dari pengguna. *Method* ini menerima input dari *request* serta data pesantren yang diulas, dan mengembalikan respon berupa pengalihan halaman (*redirect*).
- `$request->validate([...])` Digunakan untuk melakukan validasi data sebelum disimpan ke *database* agar sesuai dengan aturan yang ditetapkan.
- `'rating' => 'required|integer|min:1|max:5'` Menentukan bahwa nilai *rating* wajib diisi, harus berupa angka, dan berada pada rentang 1 sampai 5.
- `'ulasan' => 'required|string|min:5'` Menentukan bahwa ulasan wajib diisi dalam bentuk teks dengan panjang minimal 5 karakter.

```
Review::create(attributes: [  
    'user_id' => auth()->id(),  
    'pesantren_id' => $pesantren->id,  
    'rating' => $request->rating,  
    'ulasan' => $request->ulasan,  
    'tanggal' => now(),  
]);
```

Gambar 4. 7 Kode Program *method create ReviewController*

Potongan kode pada gambar 4.7 merupakan bagian dari proses penyimpanan data ulasan dan *rating* pengguna pada sistem. Kode ini digunakan untuk membuat data ulasan baru yang terhubung dengan pengguna dan pesantren yang diulas, sehingga relasi data tersimpan secara konsisten di *database*. Penjelasan tiap bagian kode adalah sebagai berikut:

- `Review::create([...])` Digunakan untuk menyimpan data ulasan baru ke dalam tabel *reviews* menggunakan metode *mass assignment*.
- `'user_id' => auth()->id()` Menyimpan ID pengguna yang sedang *login* sebagai penulis ulasan, sehingga setiap ulasan memiliki keterkaitan langsung dengan akun pengguna.
- `'pesantren_id' => $pesantren->id` Menyimpan ID pesantren yang diulas agar data ulasan terhubung dengan pesantren terkait.
- `'rating' => $request->rating` Menyimpan nilai *rating* yang diberikan oleh pengguna terhadap pesantren.
- `'ulasan' => $request->ulasan` Menyimpan teks ulasan yang dituliskan oleh pengguna.
- `'tanggal' => now()` Mencatat waktu saat ulasan dibuat untuk keperluan informasi dan pengurutan data.

**e. Method index store, reply reviewreplycontroller**

```
public function store(Request $request, Review $review): RedirectResponse
{
    $request->validate(rules: [
        'balasan' => 'required|string|min:3',
    ]);

    ReviewReply::create(attributes: [
        'review_id' => $review->id,
        'admin_id' => auth()->id(),
        'balasan' => $request->balasan,
    ]);

    return back()->with(key: 'success', value: 'Balasan berhasil dikirim');
}
```

Gambar 4. 8 Kode Program *index store, reply reviewreplycontroller*

Potongan kode pada gambar 4.8 merupakan bagian dari proses penyimpanan balasan ulasan yang dilakukan oleh *admin* pada sistem. Kode ini memastikan bahwa balasan yang dikirim valid serta tersimpan dengan relasi yang tepat ke data ulasan dan *admin*. Penjelasan tiap bagian kode adalah sebagai berikut:

- `public function store(Request $request, Review $review): RedirectResponse Method store` digunakan untuk memproses balasan ulasan. *Method* ini menerima data input dari *admin* melalui *request* serta objek ulasan yang akan dibalas, dan mengembalikan respon berupa pengalihan halaman.
- `$request->validate([...])` Digunakan untuk melakukan validasi data balasan sebelum disimpan ke *database*.
- `'balasan' => 'required|string|min:3'` Menentukan bahwa balasan wajib diisi dalam bentuk teks dengan panjang minimal 3 karakter.
- `ReviewReply::create([...])` Digunakan untuk menyimpan data balasan ulasan ke dalam tabel *review\_replies*.
- `'review_id' => $review->id` Menyimpan ID ulasan yang dibalas agar balasan terhubung langsung dengan ulasan terkait.
- `'admin_id' => auth()->id()` Menyimpan ID *admin* yang memberikan balasan sebagai penanggung jawab balasan.
- `'balasan' => $request->balasan` Menyimpan isi balasan yang dituliskan oleh *admin*.
- `return back()->with('success', 'Balasan berhasil dikirim');` Mengembalikan *admin* ke halaman sebelumnya sekaligus menampilkan pesan notifikasi bahwa balasan berhasil dikirim.

### 3. Models (Relasi Database)

*Model* dalam aplikasi Pesantrenly berperan sebagai representasi struktur data dan penghubung langsung antara sistem dengan *database*. *Model* digunakan untuk mengelola data utama seperti pesantren, pengguna, ulasan, dan balasan ulasan, serta mengatur relasi antar data tersebut. Dengan memanfaatkan fitur *Eloquent ORM* pada Laravel, setiap model mempermudah proses pengambilan, penyimpanan, pembaruan, dan penghapusan data tanpa harus menulis *query SQL* secara manual.

Selain itu, *model* juga berfungsi untuk mendefinisikan relasi antar tabel seperti hubungan antara pesantren dengan ulasan, ulasan dengan pengguna, serta ulasan dengan balasan *admin*. Pendekatan ini membantu menjaga integritas data, membuat struktur kode lebih terorganisir, dan mendukung penerapan arsitektur *Model-View-Controller* (MVC) secara konsisten dalam pengembangan sistem.

**a. Models pesantren.php**

```
class Pesantren extends Model
{
    use HasFactory;
    0 references
    protected $table = 'pesantrens';
    0 references
    protected $fillable = [
        'nama_pesantren',
        'alamat',
        'deskripsi',
        'fasilitas',
        'gambar',
    ];
    0 references
    protected $casts = [
        'gambar' => 'array',
    ];
    1 reference | 0 overrides
    public function reviews(): HasMany
    {
        return $this->hasMany(related: Review::class);
    }
}
```

Gambar 4. 9 Kode Program models Pesantren.php

Kode pada gambar 4.9 merupakan bagian dari *model* Pesantren yang berfungsi untuk merepresentasikan struktur data pesantren serta mengatur relasi antara pesantren dan data ulasan pada sistem. *Model* ini digunakan sebagai penghubung antara aplikasi dengan tabel pesantren di *database*,

sehingga pengelolaan data dapat dilakukan secara terstruktur dan aman. Adapun penjelasan dari bagian kode yang penting adalah sebagai berikut:

- `class Pesantren extends Model` Menandakan bahwa kelas Pesantren merupakan *model Eloquent* yang merepresentasikan tabel pesantren dalam *database*.
- `protected $table = 'pesantrens';` Menentukan secara eksplisit nama tabel *database* yang digunakan oleh model Pesantren.
- `protected $fillable = [...]` Mendefinisikan atribut yang boleh diisi secara massal (*mass assignment*), seperti nama pesantren, alamat, deskripsi, fasilitas, dan gambar. Hal ini berfungsi untuk menjaga keamanan data saat proses penyimpanan.
- `protected $casts = ['gambar' => 'array'];` Mengatur agar kolom gambar diperlakukan sebagai *array*, sehingga memudahkan pengelolaan data gambar yang berjumlah lebih dari satu.
- `public function reviews(): HasMany` Mendefinisikan relasi satu ke banyak (*one-to-many*) antara pesantren dan ulasan, yang berarti satu pesantren dapat memiliki banyak ulasan dari pengguna.
- `return $this->hasMany(Review::class);` Menghubungkan *model* Pesantren dengan *model Review* sehingga data ulasan dapat diakses dengan mudah melalui *model* Pesantren.

## b. Models Review.php

```
protected $fillable = [  
    'user_id',  
    'pesantren_id',  
    'ulasan',  
    'rating',  
    'tanggal',  
];  
  
public function user(): BelongsTo  
{  
    return $this->belongsTo(related: User::class);  
}  
  
0 references | 0 overrides  
public function pesantren(): BelongsTo  
{  
    return $this->belongsTo(related: Pesantren::class);  
}  
  
0 references | 0 overrides  
public function replies(): HasMany  
{  
    return $this->hasMany(related: ReviewReply::class)->latest();  
}
```

Gambar 4. 10 Kode Program *models Review.php*

Potongan kode pada gambar 4.10 merupakan bagian dari *model Review* yang berfungsi untuk mengelola data ulasan pengguna terhadap pesantren. *Model* ini tidak hanya mengatur atribut data yang dapat disimpan, tetapi juga mendefinisikan relasi antara ulasan dengan pengguna, pesantren, serta balasan ulasan dari *admin*. Dengan adanya pengaturan ini, integritas dan keterkaitan data dalam sistem dapat terjaga dengan baik. Penjelasan bagian penting kode dijabarkan sebagai berikut:

- **Protected \$fillable = [...]** Menentukan atribut yang diizinkan untuk diisi secara massal, yaitu *user\_id*, *pesantren\_id*, *ulasan*, *rating*, dan *tanggal*. Pengaturan ini bertujuan untuk menjaga keamanan data saat proses penyimpanan ulasan.

- `public function user(): BelongsTo` Mendefinisikan relasi bahwa setiap ulasan dimiliki oleh satu pengguna, sehingga sistem dapat mengetahui siapa yang memberikan ulasan.
  - `public function pesantren(): BelongsTo` Menunjukkan bahwa setiap ulasan terhubung dengan satu pesantren tertentu yang menjadi objek ulasan.
  - `public function replies(): HasMany` Mendefinisikan relasi satu ke banyak antara ulasan dan balasan ulasan, di mana satu ulasan dapat memiliki beberapa balasan dari admin.
- `->latest()` Digunakan untuk mengurutkan balasan ulasan berdasarkan data terbaru agar informasi yang ditampilkan lebih relevan.

### c. *Models Reviewreply.php*

```
public function review(): BelongsTo
{
    return $this->belongsTo(related: Review::class);
}
```

Gambar 4. 11 Kode Program models ReviewReply.php

Potongan kode berikut merupakan bagian dari *model ReviewReply* yang berfungsi untuk mengatur relasi antara balasan ulasan dengan ulasan utama pada sistem. Relasi ini memastikan bahwa setiap balasan yang diberikan oleh *admin* terhubung secara langsung dengan ulasan pengguna yang sesuai, sehingga struktur data tetap konsisten dan mudah ditelusuri. Penjelasan bagian kode adalah sebagai berikut:

- `public function review(): BelongsTo` Mendefinisikan *method review* yang menunjukkan bahwa satu data balasan ulasan hanya dimiliki oleh satu ulasan utama.
- `return $this->belongsTo(Review::class);` Menetapkan relasi *many-to-one* antara *model ReviewReply* dan *model Review*, sehingga sistem dapat mengetahui ulasan mana yang sedang dibalas oleh *admin*.

#### 4. Migrations (struktur tabel utama)

*Migration* pada aplikasi Pesantrenly digunakan untuk mengelola struktur *database* secara terkontrol dan terstruktur. *Migration* berfungsi sebagai mekanisme *version control* pada *database*, sehingga setiap perubahan skema seperti pembuatan tabel, penambahan kolom, atau perubahan tipe data dapat dikelola secara konsisten melalui kode.

Dengan *migration*, proses pembuatan dan pembaruan *database* dapat dilakukan secara otomatis menggunakan perintah Laravel tanpa harus menulis SQL manual. Pendekatan ini membantu menjaga keseragaman struktur *database*, memudahkan pengembangan, serta mendukung integritas data dalam implementasi sistem.

##### a. Migrations tabel pesantren

```
public function up(): void
{
    Schema::create(table: 'pesantrens', callback: function (Blueprint $table): void {
        $table->id();
        $table->string(column: 'nama_pesantren');
        $table->string(column: 'alamat');
        $table->text(column: 'deskripsi');
        $table->text(column: 'fasilitas')->nullable();
        $table->string(column: 'gambar')->nullable();
        $table->timestamps();
    });
};
```

Gambar 4. 12 Kode *Program Migration* tabel pesantren

Potongan kode pada gambar 4.12 merupakan bagian dari *file migration* yang digunakan untuk membentuk struktur tabel pesantrens pada *database*. *Migration* ini dijalankan saat proses pembuatan atau pembaruan skema *database* agar sesuai dengan kebutuhan sistem. Penjelasan tiap bagian kode adalah sebagai berikut:

- `public function up(): void` Method `up` digunakan untuk menjalankan proses pembuatan tabel ketika *migration* dieksekusi.
- `Schema::create('pesantrens', function (Blueprint $table) { ... })` Digunakan untuk membuat tabel baru bernama pesantrens di dalam *database*.

- `$table->id();` Membuat kolom id sebagai *primary key* yang bersifat *auto increment* untuk mengidentifikasi setiap data pesantren secara unik.
- `$table->string('nama_pesantren');` Menyimpan nama pesantren dalam bentuk teks singkat.
- `$table->string('alamat');` Digunakan untuk menyimpan alamat pesantren.
- `$table->text('deskripsi');` Menyimpan deskripsi lengkap mengenai pesantren
- `.$table->text('fasilitas')->nullable();` Menyimpan informasi fasilitas pesantren dan bersifat opsional (boleh kosong).
- `$table->string('gambar')->nullable();` Digunakan untuk menyimpan data gambar pesantren dan bersifat opsional.
- `$table->timestamps();` Menambahkan kolom *created\_at* dan *updated\_at* secara otomatis untuk mencatat waktu pembuatan dan pembaruan data.

#### b. Migrations tabel review

```
public function up(): void
{
    Schema::create(table: 'reviews', callback: function (Blueprint $table): void {
        $table->id();
        $table->foreignId(column: 'user_id')->constrained()->cascadeOnDelete();
        $table->foreignId(column: 'pesantren_id')->constrained()->cascadeOnDelete();
        $table->text(column: 'ulasan');
        $table->unsignedTinyInteger(column: 'rating'); // 1-5
        $table->date(column: 'tanggal');
        $table->timestamps();
    });
}
```

Gambar 4. 13 Kode Program Migration tabel review

Potongan kode pada gambar 4.13 merupakan bagian dari *file migration* yang digunakan untuk membentuk struktur tabel *reviews* pada *database*. Tabel ini berfungsi untuk menyimpan data ulasan dan *rating* pengguna terhadap pesantren. Penjelasan tiap bagian kode adalah sebagai berikut:

- `public function up(): void` *Method up* digunakan untuk menjalankan proses pembuatan tabel saat *migration* dieksekusi.

- `Schema::create('reviews', function (Blueprint $table) { ... })` Digunakan untuk membuat tabel baru bernama *reviews* di dalam *database*.
- `$table->id();` Membuat kolom *id* sebagai *primary key* untuk mengidentifikasi setiap data ulasan secara unik.
- `$table->foreignId('user_id')->constrained()->cascadeOnDelete();` Menyimpan relasi ke tabel pengguna (*users*) sebagai penulis ulasan. Opsi *cascadeOnDelete* memastikan bahwa ulasan akan otomatis terhapus jika data pengguna dihapus.
- `$table->foreignId('pesantren_id')->constrained()->cascadeOnDelete();` Menyimpan relasi ke tabel pesantren (*pesantrens*) agar setiap ulasan terhubung dengan pesantren yang diulas. Relasi ini juga menerapkan penghapusan berantai untuk menjaga konsistensi data.
- `$table->text('ulasan');` Digunakan untuk menyimpan isi ulasan dalam bentuk teks.
- `$table->unsignedTinyInteger('rating');` Menyimpan nilai *rating* dengan rentang 1 sampai 5 sebagai penilaian pengguna terhadap pesantren.
- `$table->date('tanggal');` Menyimpan tanggal pembuatan ulasan.
- `$table->timestamps();` Menambahkan kolom *created\_at* dan *updated\_at* untuk mencatat waktu pembuatan dan pembaruan data.

## 5. Filament Admin Panel جامعة الرانري

Filament *Admin Panel* pada aplikasi Pesantrenly digunakan sebagai antarmuka pengelolaan data bagi *administrator*. Filament berfungsi untuk menyediakan panel *admin* yang terintegrasi dengan Laravel sehingga proses manajemen data dapat dilakukan secara cepat, aman, dan terstruktur tanpa perlu membangun antarmuka dari awal.

Melalui Filament, *admin* dapat melakukan operasi CRUD (*Create, Read, Update, Delete*) terhadap data pesantren, ulasan, dan pengguna dengan dukungan fitur seperti validasi otomatis, manajemen hak akses, serta tampilan *dashboard* yang informatif. Penggunaan Filament membantu meningkatkan

efisiensi pengelolaan sistem, menjaga konsistensi data, dan mendukung penerapan arsitektur *backend* yang rapi serta mudah dikembangkan.

**a. Filament *form* dan *table* Pesantren *Resource***

```
public static function form(Schema $schema): Schema
{
    return PesantrenForm::configure(schema: $schema);
}

public static function table(Table $table): Table
{
    return PesantrenTable::configure(table: $table);
}
```

Gambar 4. 14 Filament *form* dan *table* Pesantren *Resource*

Potongan kode pada gambar 4.14 merupakan bagian dari Filament *Resource* yang digunakan untuk mengatur *form input* dan tabel data pada modul Pesantren di panel *admin*. Kode ini berfungsi memisahkan konfigurasi tampilan agar lebih terstruktur dan mudah dikelola. Penjelasannya adalah sebagai berikut:

- `public static function form(Schema $schema): Schema` *Method* ini digunakan untuk mendefinisikan struktur *form* pada halaman tambah atau edit data pesantren di *admin* panel. Konfigurasi *form* didelegasikan ke kelas Pesantren *Form* agar pengaturan *field* lebih rapi dan terpisah.
- `return PesantrenForm::configure($schema);` Memanggil konfigurasi *form* pesantren yang berisi *field-field input* seperti nama, alamat, deskripsi, fasilitas, dan gambar.
- `public static function table(Table $table): Table` *Method* ini digunakan untuk mendefinisikan struktur tabel yang menampilkan daftar data pesantren pada halaman *admin*.
- `return PesantrenTable::configure($table);` Memanggil konfigurasi tabel pesantren yang mengatur kolom data, aksi edit/hapus, serta tampilan daftar pesantren.

### b. Filament *Table Review Resource*

```
public static function table(Table $table): Table
{
    return ReviewsTable::configure(table: $table);
}
```

Gambar 4. 15 *Table Review Resource*

Potongan kode tersebut merupakan bagian dari *Filament Resource* yang berfungsi untuk mengatur tampilan tabel data ulasan pada panel *admin*. Kode ini menghubungkan *resource* dengan konfigurasi tabel yang telah didefinisikan secara terpisah. Penjelasan nya adalah sebagai berikut:

- `public static function table(Table $table): Table` *Method* ini digunakan untuk mendefinisikan struktur tabel yang akan ditampilkan pada halaman manajemen ulasan di *admin* panel.
- `return ReviewsTable::configure($table);` Memanggil kelas *ReviewsTable* yang berisi pengaturan kolom tabel, seperti nama pengguna, pesantren, rating, ulasan, dan tanggal, serta aksi pengelolaan data.

## 6. *Blade Views (Frontend User)*

*Blade Views* merupakan komponen tampilan (*frontend*) dalam *framework* Laravel yang berfungsi untuk menyajikan data dari sistem kepada pengguna dalam bentuk antarmuka *web*. *Blade* menggunakan sintaks khusus yang sederhana untuk menampilkan variabel, melakukan perulangan, serta mengatur kondisi tampilan, sehingga memudahkan pengembangan halaman *web* yang dinamis dan terstruktur.

Pada aplikasi *Pesantrenly*, *Blade Views* digunakan untuk menampilkan informasi pesantren, hasil pencarian, detail pesantren, serta ulasan dan *rating* pengguna. Data yang ditampilkan pada *Blade* berasal dari *controller* dan *model*, sehingga setiap perubahan data di *database* dapat langsung tercermin pada tampilan. Dengan pemisahan logika dan tampilan ini, sistem menjadi

lebih rapi, mudah dipelihara, dan mendukung penerapan arsitektur MVC (*Model–View–Controller*) secara optimal.

**a. Show blade**

```
<h1 class="text-2xl md:text-3xl font-bold text-green-800">
|   {{ $pesantren->nama_pesantren }}
<span class="font-bold text-green-700 text-lg">
|   {{ $pesantren->averageRating() ?? '0.0' }}
<span class="text-gray-500">
|   ({{ $pesantren->reviews->count() }} ulasan)
<p class="text-gray-600 mt-1">
|   {{ $pesantren->alamat }}
```

Gambar 4. 16 *Blade Views Show*

Kode *Blade View* pada gambar 4.16 tersebut merupakan bagian dari tampilan *detail* informasi pesantren yang berfungsi menampilkan identitas utama pesantren kepada pengguna. Informasi yang ditampilkan meliputi nama pesantren, nilai rata-rata *rating*, jumlah ulasan, serta alamat pesantren. Data ditampilkan secara dinamis dari *database* menggunakan sintaks *Blade* Laravel sehingga selalu sesuai dengan data terbaru yang tersimpan dalam sistem. Penjelasan bagian kode adalah sebagai berikut:

- `{{ $pesantren->nama_pesantren }}` Digunakan untuk menampilkan nama pesantren sebagai judul utama halaman detail pesantren.
- `{{ $pesantren->averageRating() ?? '0.0' }}` Berfungsi untuk menampilkan nilai rata-rata *rating* pesantren yang dihitung dari data ulasan pengguna. Jika belum ada ulasan, sistem menampilkan nilai default 0.0.
- `{{ $pesantren->reviews->count() }}` Digunakan untuk menampilkan jumlah total ulasan yang telah diberikan pengguna terhadap pesantren tersebut.

- `{{ $pesantren->alamat }}` Menampilkan alamat lengkap pesantren sebagai informasi lokasi yang membantu pengguna mengenali letak pesantren secara jelas.

#### b. *Search blade*

```

{{-- JUDUL --}}
<h1 class="text-2xl font-bold text-green-800 mt-10">
| Hasil pencarian: "{{ $q }}"
</h1>

@forelse ($pesantren as $pesantren)
|
| {{ $pesantren->nama_pesantren }}

<p class="text-sm text-gray-500 mt-1">
| {{ Str::limit(value: $pesantren->alamat, limit: 50) }}
</p>

```

Gambar 4. 17 *Blade Views Search*

Kode *Blade View* pada Gambar 4.17 merupakan bagian dari implementasi antarmuka pengguna yang berfungsi untuk menampilkan hasil pencarian pesantren berdasarkan kata kunci yang dimasukkan oleh pengguna. *View* ini menerima data dari *controller* berupa variabel kata kunci pencarian (`$q`) dan kumpulan data pesantren (`$pesantren`), kemudian menampilkannya secara dinamis menggunakan sintaks *Blade* Laravel. Dengan mekanisme ini, sistem mampu menyajikan hasil pencarian secara informatif, terstruktur, dan mudah dipahami oleh pengguna. Penjelasan bagian kode adalah sebagai berikut:

- `{{ $q }}` Digunakan untuk menampilkan kata kunci pencarian yang dimasukkan oleh pengguna pada judul halaman, sehingga pengguna mengetahui parameter pencarian yang sedang digunakan.
- `@forelse ($pesantren as $pesantren)` Berfungsi untuk melakukan perulangan data pesantren hasil pencarian. Struktur `@forelse`

memastikan tampilan tetap aman dan terkontrol meskipun data pencarian tidak menghasilkan hasil.

- `asset('storage/' . ($pesantren->gambar[0] ?? 'pesantren/default.jpg'))`  
Digunakan untuk menampilkan gambar pesantren dari *folder storage*. Jika data gambar tidak tersedia, sistem secara otomatis menampilkan gambar *default* sebagai *fallback*.
- `{{ $pesantren->nama_pesantren }}` Menampilkan nama pesantren sebagai informasi utama pada setiap hasil pencarian yang ditampilkan kepada pengguna.
- `Str::limit($pesantren->alamat, 50)` Digunakan untuk membatasi panjang teks alamat pesantren agar tampilan tetap rapi dan konsisten tanpa menghilangkan informasi penting.

### c. Ulasan Blade

```


<div>
  <h1 class="text-xl font-bold text-green-800">
    {{ $pesantren->nama_pesantren }}
  </h1>
  <p class="text-sm text-gray-600">
    {{ $pesantren->alamat }}
  </p>

  <form method="POST" action="{{ route('review.store', parameters: $pesantren) }}" class="space-y-5">
    @csrf

    {{-- Rating --}}
    <div>
      <label class="block text-sm font-medium mb-1">
        Berikan rating anda
      </label>
    </div>
  </form>
```

Gambar 4. 18 *Blade Views* Ulasan

Kode *Blade View* pada Gambar 4.18 merupakan bagian dari implementasi antarmuka pengguna yang berfungsi untuk menampilkan *detail* pesantren sekaligus menyediakan *form* pengisian ulasan dan *rating* oleh pengguna. *View* ini menerima data pesantren dari *controller* dan memanfaatkannya untuk menampilkan informasi utama pesantren serta memfasilitasi interaksi pengguna dalam memberikan penilaian. Dengan mekanisme ini, sistem mampu mengelola proses pemberian ulasan secara

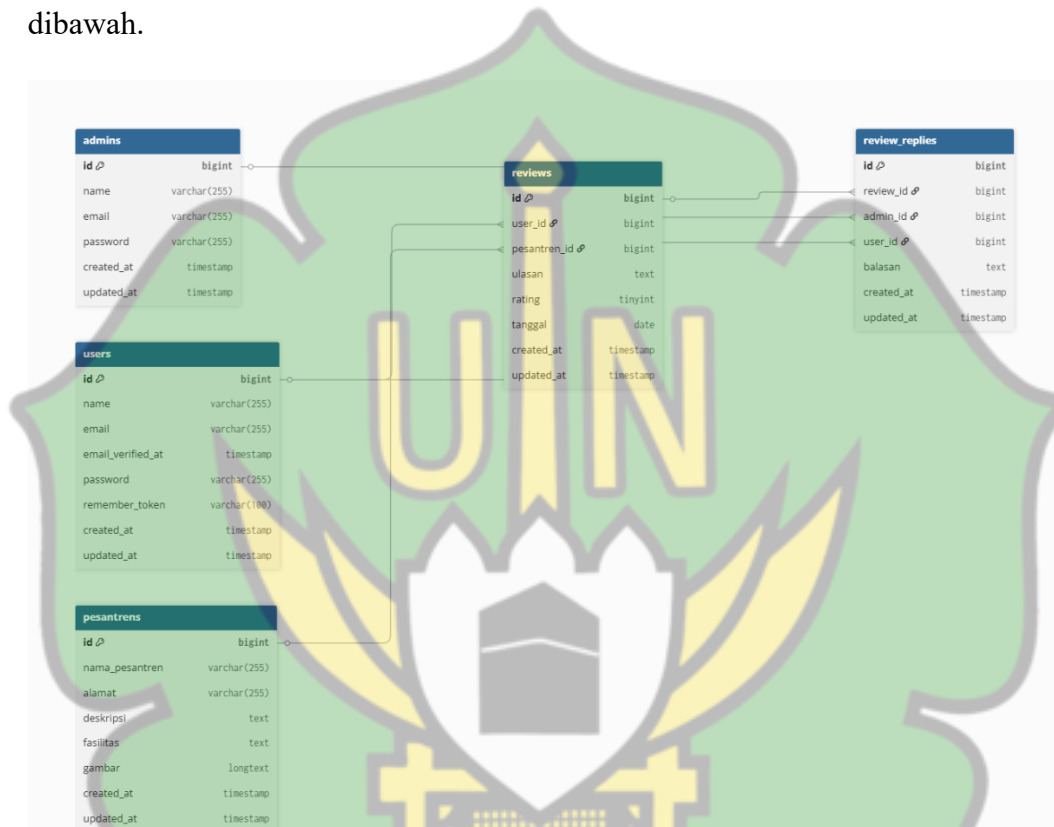
terstruktur dan terintegrasi dengan *backend*. Penjelasan bagian kode adalah sebagai berikut :

- `asset('storage/' . ($pesantren->gambar[0] ?? 'pesantren/default.jpg'))`  
Digunakan untuk menampilkan gambar pesantren yang tersimpan pada *folder storage*. Apabila data gambar tidak tersedia, sistem secara otomatis menampilkan gambar *default* sebagai *fallback*.
- `{{ $pesantren->nama_pesantren }}` Menampilkan nama pesantren sebagai identitas utama pada halaman ulasan, sehingga pengguna mengetahui pesantren yang sedang diberi penilaian.
- `{{ $pesantren->alamat }}` Digunakan untuk menampilkan alamat pesantren sebagai informasi pendukung agar pengguna memiliki konteks lokasi pesantren yang diulas.
- `<form method="POST" action="{{ route('review.store', $pesantren) }}">` Berfungsi untuk mengirimkan data ulasan dan rating pengguna ke sistem melalui *route review.store*, yang selanjutnya diproses oleh *controller*.
- `@csrf` Digunakan sebagai mekanisme keamanan Laravel untuk melindungi *form* dari serangan *Cross-Site Request Forgery* (CSRF).
- `<label>Berikan rating anda</label>` Menampilkan label input *rating* sebagai petunjuk bagi pengguna dalam memberikan penilaian terhadap pesantren.

#### 4.1.2 Implementasi Basis data

Implementasi basis data dilakukan berdasarkan rancangan *Entity Relationship Diagram (ERD)* yang telah disusun pada tahap perancangan sistem. ERD tersebut berfungsi sebagai acuan utama dalam menentukan struktur tabel, atribut, serta hubungan antar entitas yang terdapat dalam sistem *web* ulasan dan informasi pesantren. Dengan berpedoman pada ERD, proses implementasi basis data dapat dilakukan secara sistematis dan terarah sehingga mampu merepresentasikan kebutuhan data sistem secara akurat.

Rancangan ERD ini kemudian diwujudkan dalam bentuk *database schema*, yaitu rancangan struktur basis data yang menggambarkan susunan tabel dan kolom, tipe data, serta hubungan antar tabel dalam *database*. *Database schema* digunakan sebagai pedoman dalam penyimpanan dan pengelolaan data agar tersusun rapi dan mudah diakses oleh sistem. Untuk *database schema* dapat dilihat pada gambar 4.19 dibawah.



Gambar 4. 19 *Schema Database Pesantrenly*

Dari hasil perancangan *database schema* didapat lima tabel dengan rincian yang dapat dilihat pada tabel dibawah.

- **Tabel Database admins**

Tabel *admins* merupakan tabel yang digunakan untuk menyimpan data *administrator* sistem pada *website* ulasan dan informasi pesantren. *Administrator* memiliki peran penting dalam sistem, yaitu mengelola data pesantren, mengelola konten, serta memastikan informasi yang ditampilkan kepada pengguna tetap akurat dan terverifikasi. Seluruh data *admin* disimpan

dalam *database* pesantrenly pada tabel admins. Untuk tabel dapat dilihat pada tabel 4.1 dibawah.

Tabel 4. 1 *Database Admins*

No	Field Name	Type	Description
1	id	bigint	<i>Primary key</i>
2	name	varchar	
3	email	varchar	
4	password	varchar	
5	created_at	timestamp	
6	update_at	timestamp	

- Tabel *Database users*

Tabel *users* digunakan untuk menyimpan data pengguna umum pada sistem *website* ulasan dan informasi pesantren. Pengguna yang tersimpan dalam tabel ini merupakan masyarakat atau calon santri yang dapat mengakses informasi pesantren serta memberikan ulasan dan penilaian terhadap pesantren yang tersedia pada sistem. Seluruh data pengguna disimpan dalam *database* pesantrenly. Struktur tabel *users* dirancang untuk mendukung proses autentikasi, pengelolaan akun pengguna, serta pencatatan aktivitas pengguna di dalam sistem. Untuk tabel dapat dilihat pada tabel 4.2 dibawah.

Tabel 4. 2 *Database users*

No	Field Name	Type	Description
1	id	bigint	<i>Primary key</i>
2	name	varchar	
3	email	varchar	
4	email_verified_at	timestamp	
5	password	varchar	

6	remember_token	varchar	
7	created_at	timestamp	
8	updated_at	timestamp	

- Tabel *Database* pesantrens

Tabel pesantrens digunakan untuk menyimpan data utama pesantren yang ditampilkan pada sistem. Seperti dapat dilihat pada tabel di bawah, data yang disimpan meliputi id pesantren, nama pesantren, alamat, deskripsi, fasilitas, serta gambar pesantren. Selain itu, terdapat *field created\_at* dan *updated\_at* yang berfungsi untuk mencatat waktu pembuatan dan pembaruan data. Struktur tabel ini memungkinkan pengelolaan data pesantren dilakukan secara terstruktur dan efisien oleh sistem. Untuk tabel dapat dilihat pada tabel 4.3 dibawah.

Tabel 4. 3 *Database* pesantrens

No	Field Name	Type	Description
1	id	bigint	<i>Primary key</i>
2	nama_pesantren	varchar	
3	alamat	varchar	
4	deskripsi	text	
5	fasilitas	text	
6	gambar	longtext	
7	created_at	timestamp	
8	updated_at	timestamp	

- Tabel *Database* reviews

Tabel *reviews* digunakan untuk menyimpan data ulasan dan penilaian yang diberikan oleh pengguna terhadap pesantren. Tabel ini mencatat hubungan antara pengguna dan pesantren melalui *user\_id* dan *pesantren\_id*, serta menyimpan isi ulasan, *rating* (1–5), dan tanggal ulasan. Selain itu, *field created\_at* dan *updated\_at* digunakan untuk mencatat waktu pembuatan dan

pembaruan data, sehingga ulasan dapat dikelola dan ditampilkan secara terstruktur oleh sistem. Untuk tabel dapat dilihat pada tabel 4.4 dibawah.

Tabel 4. 4 *Database reviews*

No	Field Name	Type	Description
1	id	bigint	<i>Primary key</i>
2	user_id	bigint	<i>Foreign key</i>
3	pesantren_id	bigint	<i>Foreign key</i>
4	ulasan	text	
5	rating	tinyint	
6	tanggal	date	
7	created_at	timestamp	
8	updated_at	timestamp	

- Tabel *Database reviews replies*

Tabel *reviews replies* digunakan untuk menyimpan data balasan terhadap ulasan yang diberikan oleh pengguna pada pesantren. Tabel ini menghubungkan ulasan dengan pihak yang memberikan balasan, baik *admin* maupun pengguna, sehingga interaksi dapat tercatat dengan jelas. Selain itu, tabel ini juga menyimpan isi balasan serta informasi waktu pembuatan dan pembaruan data untuk menjaga keteraturan dan riwayat interaksi dalam sistem. Untuk tabel dapat dilihat pada tabel 4.5 dibawah.

Tabel 4. 5 *Database reviews replies*

No	Field Name	Type	Description
1	id	bigint	<i>Primary key</i>
	review_id	bigint	<i>Foreign key</i>
	admin_id	bigint	<i>Foreign key</i>
	user_id	bigint	<i>Foreign key</i>

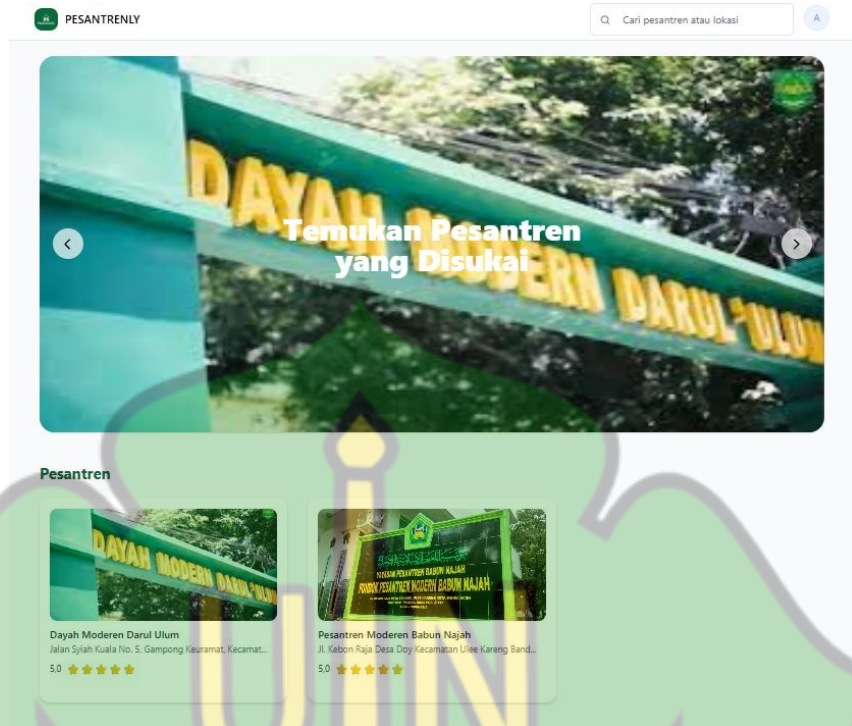
	balasan	text	
	created_at	timestamp	
	update_at	timestamp	

### 4.1.3 Implementasi antar muka Pengguna (UI)

Antar muka pengguna (*User Interface/UI*) adalah bagian sistem yang menjadi penghubung antara pengguna dan aplikasi atau *website*. UI dirancang agar tampilan mudah dipahami, navigasi jelas, dan fitur dapat digunakan dengan nyaman. Dengan antar muka yang baik, pengguna dapat mengakses informasi dan menggunakan sistem secara efektif tanpa kesulitan.

#### a. Tampilan Halaman Utama

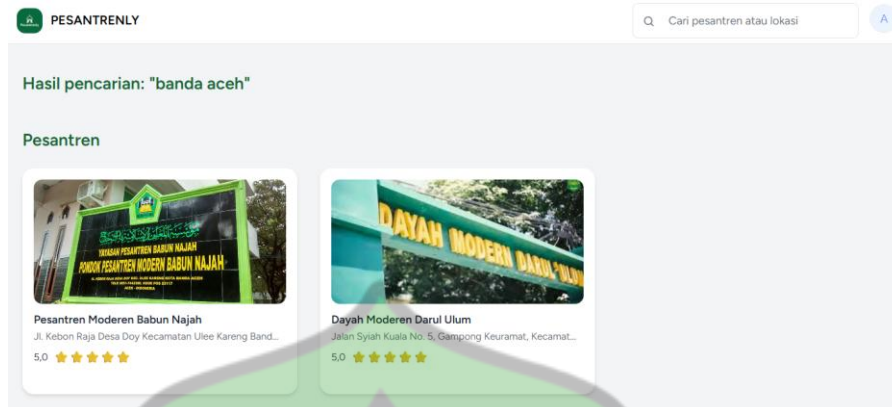
Halaman Utama (*Landing Page*) dirancang dengan pendekatan *User-Centered Design*, di mana tampilan dan fitur difokuskan pada kebutuhan pengguna. Fitur pencarian pesantren ditempatkan sebagai elemen utama untuk memudahkan pengguna dalam menemukan pesantren berdasarkan nama atau lokasi. Selain itu, halaman ini menampilkan daftar pesantren populer dalam bentuk kartu yang informatif sehingga pengguna dapat mengakses informasi secara cepat dan mudah. Untuk gambar dapat dilihat pada gambar 4.20 dibawah.....



Gambar 4. 20 Tampilan Halaman Utama

### b. Tampilan Fitur Pencarian

Tampilan fitur pencarian menampilkan hasil pencarian pesantren berdasarkan kata kunci yang dimasukkan pengguna, seperti nama daerah atau lokasi. Pada halaman ini, sistem menampilkan informasi “hasil pencarian” untuk memperjelas konteks pencarian yang dilakukan. Hasil ditampilkan dalam bentuk kartu pesantren yang berisi gambar, nama pesantren, alamat singkat, dan *rating*, sehingga pengguna dapat dengan mudah membandingkan dan memilih pesantren yang sesuai dengan kebutuhan. Untuk gambar dapat dilihat pada gambar 4.21 dibawah.



Gambar 4. 21 Tampilan Fitur Pencarian

### c. Tampilan Detail Konten

Tampilan detail konten menyajikan informasi lengkap mengenai pesantren yang dipilih oleh pengguna. Pada halaman ini ditampilkan nama pesantren, *rating* beserta jumlah ulasan, serta alamat lengkap sebagai informasi utama. Selain itu, tersedia galeri gambar untuk memberikan gambaran visual lingkungan pesantren. Konten juga dilengkapi dengan deskripsi dan fasilitas pesantren sehingga pengguna dapat memahami profil pesantren secara menyeluruh sebelum mengambil keputusan. Untuk gambar dapat dilihat pada gambar 4.22 dan 4.23 dibawah

## Dayah Modern Darul Ulum

5 ★★★★★ (1 ulasan)

Jalan Syiah Kuala No. 5, Gampong Keuramat, Kecamatan Kuta Alam, Kota Banda Aceh, Provinsi Aceh



### Deskripsi

#### Latar Belakang

Pesantren Modern Darul 'Ulum (DDU) di Banda Aceh adalah lembaga pendidikan Islam terpadu yang didirikan tahun 1990, menggabungkan sistem pesantren dan madrasah, bertujuan mencetak generasi Qurani berakhlak mulia dan berwawasan IPTEK, menawarkan jenjang MTs/SMP dan MA, serta fasilitas pendukung seperti perpustakaan dan laboratorium, dikelola oleh Yayasan Pembangunan Umat Islam (YPUI).

#### Visi

Menjadi pusat pengembangan ilmu pendidikan Islam yang beradab dan berakhlak, serta siap menjawab tantangan zaman.

#### Misi

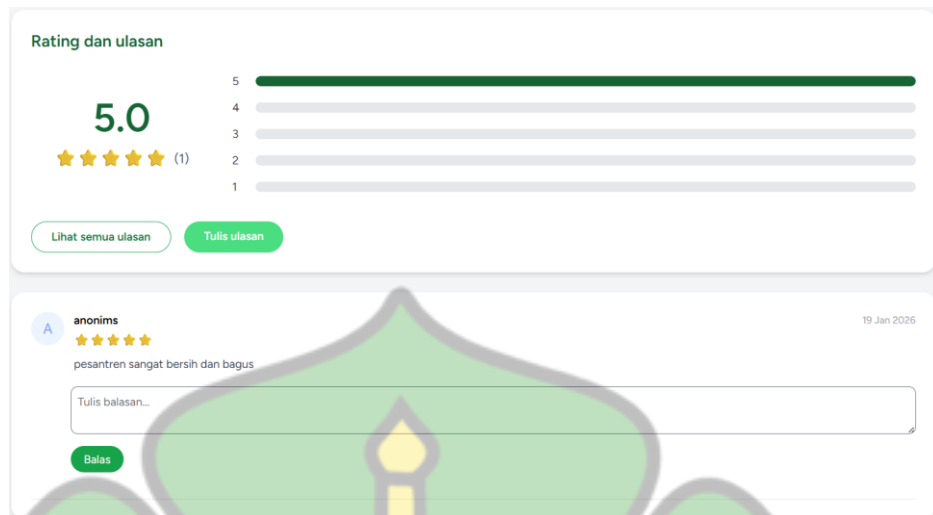
Membangun dasar aqidah Islamiyah dan akhlakul karimah.  
 Memperdalam pemahaman Al-Qur'an dan Hadis.  
 Menyediakan pendidikan ilmu-ilmu Islam yang menyeluruh.  
 Meningkatkan keterampilan sesuai kebutuhan zaman.  
 Membentuk generasi Islam yang tangguh menghadapi perubahan.

### Fasilitas Pesantren

Ruang Kelas  
 Masjid  
 Laboratorium Komputer

Asrama Putra - Putri  
 Perpustakaan  
 Toilet

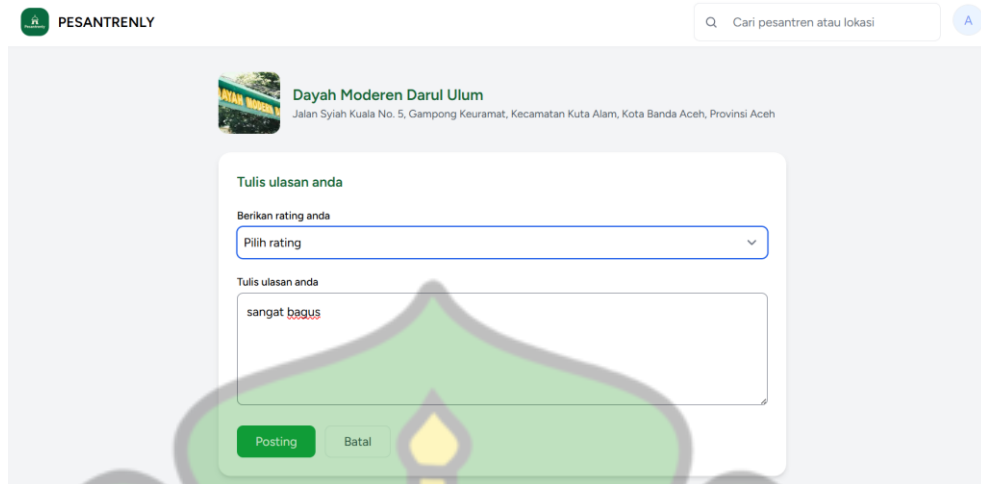
Gambar 4. 22 Tampilan *Detail Konten*



Gambar 4. 23 Tampilan *Detail* Konten

#### d. Tampilan halaman Ulasan dan Rating

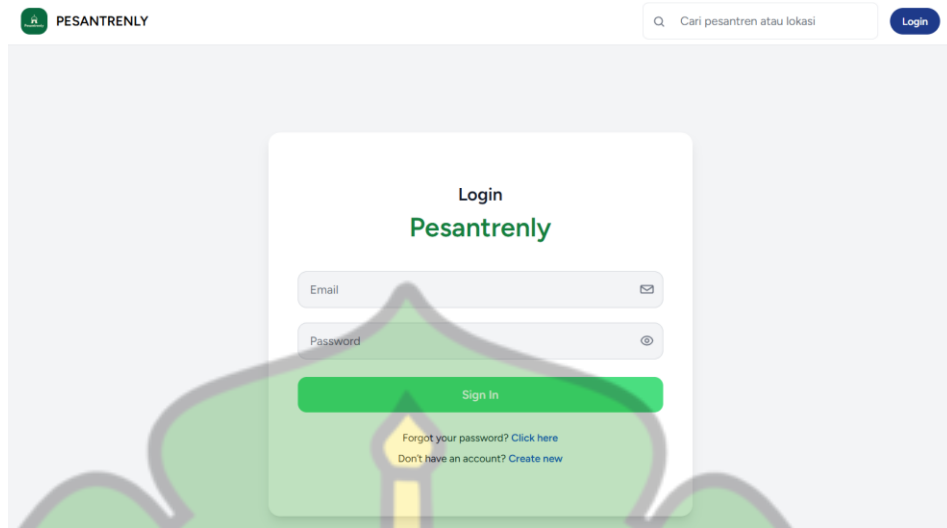
Tampilan ulasan dan *rating* berfungsi sebagai sarana bagi pengguna untuk memberikan penilaian terhadap pesantren. Pada halaman ini ditampilkan nama dan informasi singkat pesantren, diikuti dengan *form* pengisian *rating* dan ulasan. Pengguna dapat memilih nilai *rating* serta menuliskan pengalaman atau pendapatnya pada kolom ulasan yang tersedia. Tersedia tombol posting untuk mengirim ulasan dan tombol batal untuk membatalkan pengisian, sehingga proses pemberian ulasan menjadi mudah dan terkontrol. Untuk gambar dapat dilihat pada gambar 4.24 dibawah.



Gambar 4. 24 Tampilan halaman Ulasan dan *Rating*

#### e. Tampilan halaman *Login User*

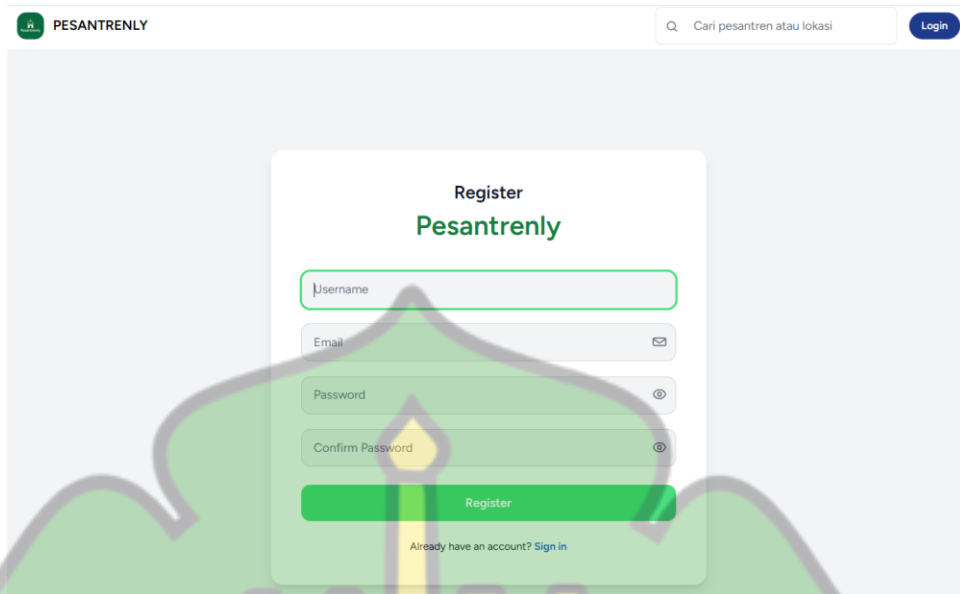
Tampilan halaman *login user* berfungsi sebagai gerbang autentikasi bagi pengguna sebelum mengakses fitur tertentu pada sistem. Halaman ini menyediakan *form login* yang terdiri dari kolom *email* dan *password* untuk memastikan keamanan akun pengguna. Selain itu, terdapat tombol *Sign In* untuk masuk ke sistem, serta opsi *Forgot your password* bagi pengguna yang lupa kata sandi dan *Create new* untuk pendaftaran akun baru. Desain halaman dibuat sederhana dan jelas agar proses login dapat dilakukan dengan mudah dan cepat. Untuk gambar dapat dilihat pada gambar 4.25 dibawah.



Gambar 4. 25 Tampilan halaman *Login User*

#### f. Tampilan halaman *register*

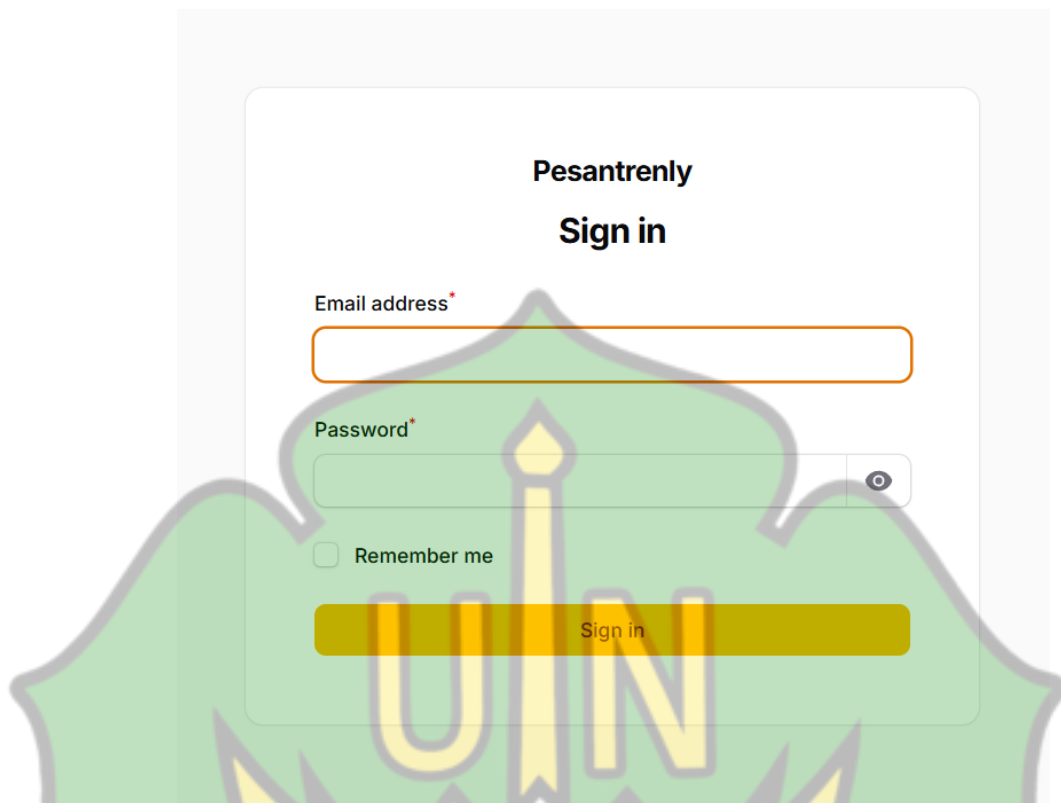
Tampilan halaman *registrasi user* berfungsi untuk memungkinkan pengguna membuat akun baru pada sistem. Pada halaman ini disediakan form pendaftaran yang terdiri dari kolom *username*, *email*, *password*, dan konfirmasi *password* untuk memastikan data yang dimasukkan sesuai. Terdapat tombol *Register* untuk mengirim data pendaftaran, serta tautan *Sign in* bagi pengguna yang sudah memiliki akun. Desain halaman dibuat sederhana dan terfokus agar proses *registrasi* dapat dilakukan dengan mudah dan aman. Untuk gambar dapat dilihat pada gambar 4.26 dibawah.



Gambar 4. 26 Tampilan halaman *register*

**g. Tampilan halaman *Login Admin***

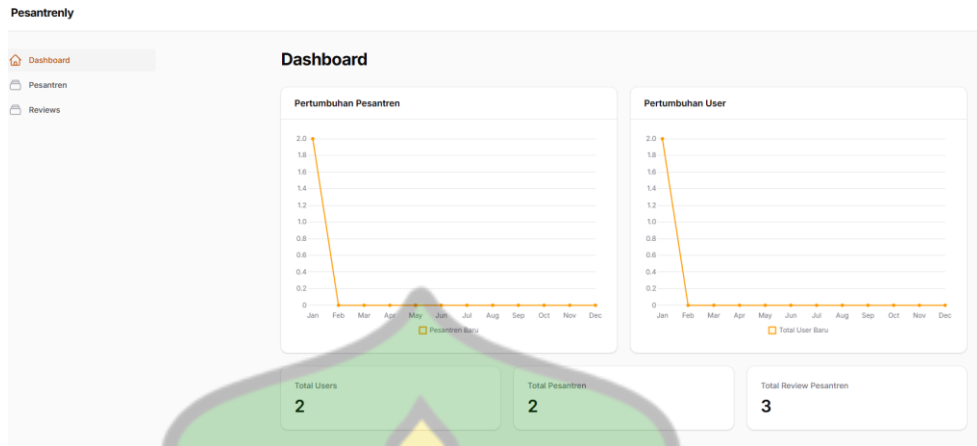
Tampilan halaman sign in ini berfungsi sebagai antarmuka autentikasi pengguna untuk masuk ke dalam sistem Pesantrenly. Halaman menampilkan *form login* yang terdiri dari kolom email dan password yang wajib diisi oleh pengguna. Selain itu, terdapat fitur *Remember me* untuk menyimpan sesi *login* serta ikon untuk menampilkan atau menyembunyikan password. Tombol *Sign in* disediakan sebagai aksi utama untuk mengakses sistem, dengan desain gelap yang sederhana dan fokus pada kenyamanan serta keamanan pengguna. Untuk gambar dapat dilihat pada gambar 4.27 dibawah.



Gambar 4. 27 Tampilan halaman *Login Admin*

#### **h. Tampilan halaman Dashboard Admin**

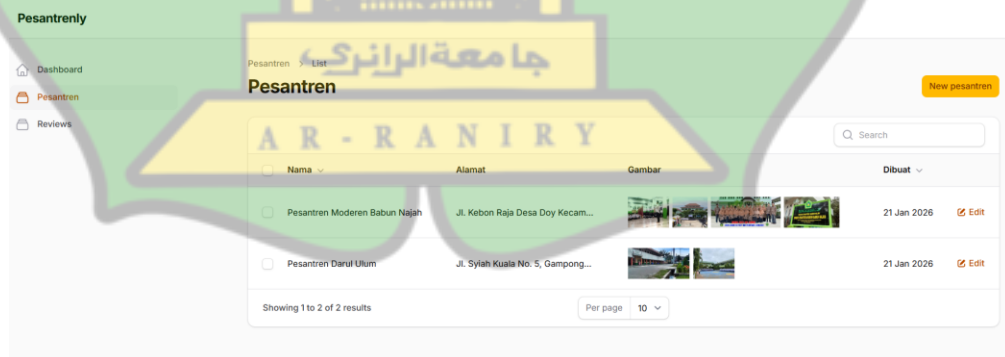
Tampilan dashboard berfungsi sebagai halaman utama bagi *admin* untuk memantau aktivitas dan perkembangan sistem Pesantrenly. Pada halaman ini ditampilkan grafik pertumbuhan pesantren dan pertumbuhan pengguna berdasarkan data bulanan, sehingga memudahkan *admin* dalam melihat tren perkembangan sistem. Selain itu, *dashboard* juga menampilkan ringkasan data berupa total pengguna, total pesantren, dan total ulasan pesantren dalam bentuk kartu informasi. Desain *dashboard* dibuat informatif dan ringkas agar *admin* dapat memperoleh gambaran kondisi sistem secara cepat dan jelas. Untuk gambar dapat dilihat pada gambar 4.28 dibawah.



Gambar 4. 28 Tampilan halaman *Dashboard Admin*

**i. Tampilan halaman manajemen data**

Halaman ini berfungsi untuk menampilkan daftar pesantren yang tersimpan dalam sistem dalam bentuk tabel. Informasi yang ditampilkan meliputi nama pesantren, alamat, gambar, serta tanggal pembuatan data. Selain itu, tersedia fitur pencarian, pengaturan jumlah data per halaman, tombol *New* pesantren untuk menambahkan data baru, serta aksi *Edit* untuk mengubah data pesantren yang sudah ada. Halaman ini dirancang untuk memudahkan admin dalam mengelola data pesantren secara terstruktur dan efisien. Untuk gambar dapat dilihat pada gambar 4.29 dibawah.



Gambar 4. 29 Tampilan halaman manajemen data

#### j. Tampilan halaman tambah data

Pada halaman ini tersedia *form* pengisian yang meliputi nama pesantren dan alamat sebagai data utama, serta kolom deskripsi yang dilengkapi *editor* teks untuk menuliskan informasi pesantren secara *detail*. Selain itu, terdapat kolom fasilitas untuk mencantumkan sarana pendukung pesantren dan fitur unggah gambar untuk menambahkan dokumentasi visual. Di bagian bawah disediakan tombol *Create*, *Create & create another*, dan *Cancel* untuk mengatur proses penyimpanan data sesuai kebutuhan *admin*. Untuk gambar dapat dilihat pada gambar 4.30 dibawah.



Gambar 4. 30 Tampilan halaman tambah data

#### k. Tampilan halaman manajemen ulasan

Tampilan ini merupakan halaman manajemen ulasan (*Reviews List*) yang digunakan oleh *admin* untuk mengelola data ulasan pengguna terhadap pesantren. Pada halaman ini ditampilkan daftar ulasan dalam bentuk tabel yang berisi informasi nama pengguna, pesantren yang diulas, nilai *rating*, isi ulasan, serta tanggal ulasan dibuat. Selain itu, tersedia aksi Hapus untuk menghapus ulasan yang tidak sesuai. Halaman ini dirancang untuk membantu *admin* memantau kualitas ulasan dan menjaga kredibilitas informasi pada sistem. Untuk gambar dapat dilihat pada gambar 4.31 dibawah.

Pesantrenly

Dashboard Pesantren Reviews

Reviews > List New review

<input type="checkbox"/>	User	Pesantren	Rating	Ulasan	Tanggal	
<input type="checkbox"/>	anonims	Pesantren Moderen Babun Najah	5	pesantren sangat bersih dan nyaman	Jan 21, 2026	Hapus
<input type="checkbox"/>	bjorka	Pesantren Moderen Babun Najah	4	ingin berkunjung kesini	Jan 21, 2026	Hapus
<input type="checkbox"/>	bjorka	Pesantren Darul Ulum	5	pesantrennya sangat indah dan dekat dari pusat kot...	Jan 21, 2026	Hapus

Showing 1 to 3 of 3 results Per page 10

Gambar 4. 31 Tampilan halaman manajemen ulasan

## 4.2 Pengujian sistem

Pengujian sistem pada penelitian ini menggunakan metode *Black Box Testing*, yaitu metode pengujian yang berfokus pada fungsi sistem tanpa melihat struktur atau kode *program* di dalamnya. Pengujian dilakukan dengan cara memberikan input pada setiap fitur sistem dan mengamati *output* yang dihasilkan. Tujuan dari pengujian ini adalah untuk memastikan bahwa seluruh fungsi, seperti *login*, *registrasi*, pencarian pesantren, pengelolaan data, serta fitur ulasan dan *rating*, berjalan sesuai dengan kebutuhan pengguna. Dengan pengujian *black box*, dapat diketahui apakah sistem telah berfungsi dengan baik dan sesuai dengan spesifikasi yang telah dirancang.

### 4.2.1 Pengujian fungsional

Pengujian fungsional secara umum merupakan proses pengujian yang bertujuan untuk memastikan bahwa setiap fungsi dalam sistem dapat berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan. Pengujian ini dilakukan dengan mengamati kesesuaian antara input yang diberikan dan *output* yang dihasilkan oleh sistem. Melalui pengujian fungsional, dapat diketahui apakah sistem sudah bekerja dengan benar, stabil, dan siap digunakan oleh pengguna. Hasil pengujian dapat dilihat pada tabel 4.6 dibawah.

Tabel 4. 6 Pengujian Fungsional

No	Fitur	Skenario Pengujian	Hasil yang Diharapkan	Keterangan
1	Registrasi User	Mengisi form registrasi dengan data valid	Akun berhasil dibuat	valid
2	Login User	Login dengan email dan password benar	Berhasil masuk ke sistem	valid
3	Login User	Login dengan password salah	Pesan kesalahan ditampilkan	valid
4	Pencarian Pesantren	Mencari pesantren berdasarkan kata kunci	Data pesantren tampil	valid
5	Detail Pesantren	Membuka detail pesantren	Informasi detail tampil	valid
6	Ulasan & Rating	Mengirim rating dan ulasan	Ulasan tersimpan	valid
7	Balas Ulasan	Admin membalas ulasan	Balasan tampil	valid
8	Dashboard Admin	Admin membuka dashboard	Data statistik tampil	valid
9	Kelola Pesantren	Admin menambah data pesantren	Data tersimpan	valid
10	Kelola Pesantren	Admin mengedit data pesantren	Perubahan tersimpan	valid
11	Kelola Ulasan	Admin menghapus ulasan	Ulasan terhapus	valid

Berdasarkan hasil pengujian fungsional yang telah dipaparkan pada Tabel 4.6, dapat disimpulkan bahwa sistem informasi Pesantrenly telah memenuhi standar keberhasilan operasional sesuai dengan alur logika yang dirancang. Pengujian dilakukan menggunakan metode *Black Box Testing* untuk memvalidasi fungsi-fungsi utama sistem, baik pada sisi pengguna maupun *admin*, sehingga dapat dipastikan bahwa setiap fitur berjalan sesuai dengan kebutuhan yang telah ditetapkan.

1) Kesesuaian Logika Proses (*ProcessLogic*)

Seluruh alur pengelolaan data pesantren dan ulasan (CRUD) yang dijalankan oleh *admin* melalui panel sistem telah berfungsi sesuai dengan perancangan. Data pesantren, ulasan, dan balasan ulasan dapat disimpan, diperbarui, serta dihapus dengan baik, sehingga menjamin keteraturan dan keamanan data dari akses yang tidak sah.

2) Efektivitas Penelusuran Informasi

Fitur pencarian pesantren terbukti mampu membantu pengguna dalam menemukan informasi pesantren secara cepat dan tepat. Sistem berhasil menampilkan data pesantren yang relevan berdasarkan kata kunci pencarian, sehingga mendukung kemudahan akses informasi dan meningkatkan efisiensi penggunaan sistem.

3) Responsivitas Antarmuka Sistem

Antarmuka sistem Pesantrenly menunjukkan konsistensi tampilan dan kemudahan penggunaan pada berbagai perangkat. Hal ini membuktikan bahwa sistem dapat diakses secara fleksibel baik melalui komputer maupun perangkat *mobile* tanpa mengurangi kenyamanan dan kualitas pengalaman pengguna.

Secara keseluruhan, hasil pengujian fungsional menunjukkan bahwa sistem Pesantrenly telah berjalan dengan baik dan optimal. Seluruh fitur yang diuji dinyatakan valid, sehingga sistem dinilai layak untuk digunakan sebagai media penyedia informasi dan ulasan pesantren bagi masyarakat.

#### 4.2.2 Pengujian integritas data dan logika bisnis

Pengujian integritas data dan logika bisnis dilakukan untuk memastikan bahwa data yang tersimpan dalam sistem tetap konsisten, saling terhubung dengan benar, dan sesuai dengan aturan proses bisnis yang telah ditetapkan. Pengujian ini mencakup pemeriksaan relasi antar data pesantren, ulasan, dan pengguna, serta validasi hak akses dan alur penggunaan sistem. Hasil pengujian menunjukkan bahwa sistem mampu menjaga keutuhan data dan menerapkan logika bisnis dengan baik, sehingga seluruh proses berjalan sesuai perancangan dan sistem dinyatakan layak digunakan. Hasil pengujian dapat dilihat pada tabel 4.7 dibawah.

Tabel 4. 7 Pengujian integritas data dan logika bisnis

No	Logika yang Diuji	Skenario Uji	Hasil Aktual	Analisis
1	Integritas Data Pesantren	Admin menambahkan data pesantren	Berhasil	Data pesantren tersimpan dengan baik dan terstruktur sesuai relasi database.
2	Relasi Data Ulasan	User memberikan ulasan pada pesantren	Berhasil	Data ulasan terhubung dengan data pesantren dan pengguna melalui foreign key.
3	Logika Pencarian Pesantren	User melakukan pencarian pesantren	Berhasil	Sistem menampilkan data pesantren yang relevan sesuai kata kunci pencarian.
4	Hak Akses Sistem	User dan admin mengakses fitur sesuai perannya	Berhasil	Sistem membatasi akses sesuai hak pengguna sehingga logika bisnis berjalan benar.

5	Balasan Ulasan	Admin membalas ulasan pengguna	Berhasil	Balasan ulasan tersimpan dan terkait dengan data ulasan secara konsisten.
---	----------------	--------------------------------	----------	---

Berdasarkan hasil pengujian yang tertera pada Tabel 4.7, sistem terbukti mampu menangani logika bisnis yang kompleks guna menjaga integritas dan akurasi informasi yang disajikan. Rangkaian pengujian ini memberikan gambaran komprehensif mengenai keandalan logika backend sistem melalui poin-poin analisis sebagai berikut:

1. Integritas Data Pesantren

Pengujian menunjukkan bahwa proses penambahan data pesantren oleh admin pada sistem Pesantrenly berjalan dengan baik. Data pesantren berhasil tersimpan ke dalam database secara terstruktur dan konsisten sesuai dengan relasi tabel yang telah dirancang, sehingga informasi pesantren dapat dikelola dengan aman.

2. Relasi Data Ulasan

Pada saat pengguna memberikan ulasan terhadap pesantren, sistem berhasil menyimpan data ulasan dan menghubungkannya dengan data pesantren serta data pengguna melalui foreign key. Hal ini membuktikan bahwa relasi antar data berjalan dengan benar dan menjaga integritas data ulasan.

3. Logika Pencarian Pesantren

Fitur pencarian pesantren mampu menampilkan data pesantren yang relevan berdasarkan kata kunci yang dimasukkan oleh pengguna, baik berdasarkan nama maupun lokasi. Pengujian ini menunjukkan bahwa logika pencarian pada sistem berjalan efektif dan mendukung kemudahan akses informasi.

4. Hak Akses Sistem

Sistem Pesantrenly berhasil menerapkan pembatasan akses sesuai dengan peran pengguna dan *admin*. Pengguna hanya dapat mengakses fitur yang

diperbolehkan, sedangkan admin memiliki akses penuh untuk pengelolaan data, sehingga logika bisnis dan keamanan sistem dapat terjaga.

#### 5. Balasan Ulasan

Fitur balasan ulasan memungkinkan admin untuk menanggapi ulasan yang diberikan oleh pengguna. Hasil pengujian menunjukkan bahwa balasan ulasan tersimpan dengan baik dan terhubung secara konsisten dengan data ulasan terkait, sehingga interaksi antara pengguna dan pengelola pesantren dapat tercatat dengan rapi.

Secara keseluruhan, hasil pengujian integritas data dan logika bisnis mempertegas bahwa sistem Pesantrenly memiliki keunggulan teknis dalam pengelolaan dan keterkaitan data dibandingkan sistem informasi statis pada umumnya. Keberhasilan pengujian ini menunjukkan bahwa sistem tidak hanya unggul dari sisi tampilan antarmuka, tetapi juga didukung oleh fondasi *backend* yang kuat dalam menjaga konsistensi data, keamanan akses, serta keakuratan informasi pesantren dan ulasan. Dengan demikian, sistem Pesantrenly dinilai mampu menyajikan informasi yang valid, terstruktur, dan terpercaya bagi pengguna.

### 4.3 Pembahasan Hasil

Subbab ini menyajikan pembahasan mendalam mengenai hasil implementasi dan pengujian sistem yang telah dilakukan pada subbab sebelumnya. Analisis difokuskan pada penilaian kinerja sistem Pesantrenly dalam memenuhi tujuan penelitian, khususnya dalam menyediakan informasi pesantren yang terintegrasi dan mudah diakses. Selain itu, pembahasan ini juga meninjau kemampuan sistem dalam mengatasi permasalahan penyebaran informasi yang tidak terpusat serta keterbatasan akses informasi pesantren yang sebelumnya dialami oleh masyarakat.

#### 4.3.1 Analisis Implementasi Teknologi

Analisis ini menjelaskan bagaimana penerapan teknologi pada sisi *backend* dan *frontend* dalam sistem Pesantrenly saling terintegrasi untuk mendukung keamanan data, konsistensi informasi, serta kelancaran alur kerja sistem. Integrasi teknologi yang digunakan bertujuan untuk memastikan bahwa proses pengelolaan data pesantren, ulasan, dan interaksi pengguna dapat berjalan secara efisien dan sesuai

dengan kebutuhan pengguna. Efektivitas sistem ditinjau melalui beberapa aspek utama sebagai berikut:

1. **Keamanan dan Akurasi Data** : Berdasarkan hasil pengujian, sistem Pesantrenly terbukti mampu menyimpan data pesantren, ulasan, dan balasan ulasan secara aman dan akurat. Penerapan relasi basis data dan mekanisme pengelolaan data memastikan seluruh informasi, mulai dari profil pesantren hingga ulasan pengguna, tersimpan secara utuh tanpa risiko kerusakan atau kehilangan data. Selain itu, sistem mampu menjaga keterkaitan antar data secara konsisten untuk mendukung penyajian informasi yang tepat.
2. **Validasi Integritas Konten** : Mekanisme validasi yang diterapkan pada proses pengelolaan data pesantren dan ulasan (CRUD) berjalan dengan efektif. Sistem secara otomatis memastikan bahwa data yang disimpan telah memenuhi ketentuan yang ditetapkan, seperti kelengkapan informasi dan kesesuaian *format* data. Hal ini menjamin bahwa informasi pesantren dan ulasan yang ditampilkan kepada pengguna bersifat konsisten, valid, dan dapat dipercaya.
3. **Manajemen Hak Akses (*Role-Based Access Control*)** : Implementasi pembatasan hak akses berbasis peran pada sistem Pesantrenly berhasil memisahkan area publik dan area *administrasi*. Sistem memastikan bahwa hanya *admin* yang memiliki kewenangan untuk menambah, mengubah, atau menghapus data pesantren dan melakukan moderasi ulasan. Dengan demikian, keamanan sistem dapat terjaga dari potensi penyalahgunaan atau manipulasi data oleh pihak yang tidak berwenang.

#### 4.3.2 Pencapaian tujuan penelitian

Hasil implementasi dan pengujian sistem menunjukkan bahwa penelitian ini telah berhasil merealisasikan tujuan yang telah ditetapkan. Sistem Pesantrenly mampu menjawab permasalahan utama terkait keterbatasan dan fragmentasi informasi pesantren dengan menyediakan *platform digital* yang terintegrasi. Indikator pencapaian tujuan penelitian diuraikan sebagai berikut:

1. **Realisasi Platform Informasi Pesantren Terpadu** : Aplikasi Pesantrenly yang dibangun menggunakan *framework* Laravel dan dukungan Filament berhasil direalisasikan sebagai platform informasi terpusat mengenai pesantren. Sistem ini berfungsi sebagai *media digital* yang menyajikan informasi profil pesantren, lokasi, fasilitas, serta dokumentasi visual secara terstruktur. Dengan adanya basis data terpusat, sistem mampu mengatasi permasalahan penyebaran informasi pesantren yang sebelumnya tersebar di berbagai media.
2. **Validasi Fungsionalitas Sistem** : Seluruh fitur utama sistem, mulai dari autentikasi pengguna, pencarian pesantren, penampilan detail pesantren, hingga pengelolaan ulasan dan data oleh *admin*, telah diuji menggunakan metode *Black Box Testing* dan dinyatakan berfungsi dengan baik. Hasil pengujian menunjukkan bahwa sistem mampu memfasilitasi pengguna dalam menemukan dan mengakses informasi pesantren secara cepat dan mudah, sehingga mendukung tujuan penelitian dalam meningkatkan aksesibilitas dan kualitas informasi pesantren bagi masyarakat.

Secara menyeluruh, aplikasi Pesantrenly telah menunjukkan kinerja yang baik dan mampu berperan sebagai solusi *digital* terpadu dalam pengelolaan serta penyampaian informasi pesantren. Keberadaan sistem ini mendukung optimalisasi tata kelola data dan publikasi informasi pesantren secara modern, sehingga selaras dengan kebutuhan masyarakat di tengah perkembangan teknologi *digital*.

#### 4.3.3 Analisis Penelitian Studi Terkait

Tabel penelitian terdahulu disusun untuk memberikan gambaran perbandingan penelitian yang relevan berdasarkan metode, teknologi, serta hasil yang diperoleh, termasuk kelebihan dan kekurangannya. Analisis ini bertujuan untuk mengetahui posisi penelitian ini serta mengidentifikasi celah penelitian yang masih dapat dikembangkan. Uraian penelitian terdahulu tersebut dapat dilihat pada tabel 4.8 di bawah.

Tabel 4. 8 Perbandingan Metode dan Hasil Penelitian Terdahulu

NO	REFERENSI	METODE	HASIL	KELEBIHAN	KEKURANGAN
1	Zaharuddin, Pradana, & Brata (2020)	<i>Framework</i> Laravel (MVC) digunakan untuk merancang sistem penilaian santri berbasis web dengan pemisahan antara <i>model, view, dan controller</i> sehingga memudahkan pengelolaan data.	Hasil penelitian menunjukkan Laravel mampu menyediakan sistem yang terintegrasi, <i>user friendly</i> , dan dapat diakses oleh wali santri melalui internet, sehingga proses <i>monitoring</i> nilai santri menjadi lebih transparan dan efektif.	<ul style="list-style-type: none"> <li>Menggunakan <i>Framework</i> Laravel dengan arsitektur MVC sehingga sistem terstruktur dan mudah dikelola.</li> <li>Sistem terintegrasi dan mudah diakses secara <i>online</i>, meningkatkan transparansi <i>monitoring</i> nilai santri.</li> </ul>	<ul style="list-style-type: none"> <li>Fokus sistem masih terbatas pada penilaian santri, belum mencakup informasi lembaga secara menyeluruh.</li> <li>Tidak membahas metode pengembangan iteratif maupun evaluasi sistem lanjutan.</li> </ul>
2	Rahmawati & Sumarsono (2024)	Arsitektur MVC dengan <i>Framework</i> Laravel digunakan untuk	Penggunaan Laravel berbasis MVC menghasilkan performa pencarian data yang lebih	<ul style="list-style-type: none"> <li>Penerapan MVC dengan Laravel menghasilkan performa pencarian data</li> </ul>	<ul style="list-style-type: none"> <li>Sistem belum mengakomodasi kebutuhan pengguna yang dinamis seperti personalisasi</li> </ul>

		membangun desain <i>website</i> , dengan pemisahan logika aplikasi, antarmuka, dan basis data.	cepat, tampilan yang responsif, serta keaktualan informasi yang lebih baik dibandingkan pendekatan konvensional.	yang cepat dan stabil. <ul style="list-style-type: none"> <li>• Tampilan <i>website</i> responsif dan informatif, meningkatkan kualitas penyajian informasi.</li> </ul>	atau fitur interaktif. <ul style="list-style-type: none"> <li>• Evaluasi sistem belum melibatkan pengujian dari sisi pengguna akhir secara mendalam.</li> </ul>
3	Haris (2023)	Pendekatan studi literatur dan analisis fenomena digitalisasi pesantren yang menekankan pemanfaatan <i>website</i> dan media sosial dalam pendidikan.	Hasil menunjukkan bahwa digitalisasi melalui <i>website</i> dan media sosial penting untuk <i>branding</i> pesantren, administrasi, dan literasi digital, meskipun masih terkendala SDM teknis.	<ul style="list-style-type: none"> <li>• Memberikan kajian yang luas mengenai digitalisasi pesantren dari sisi konseptual</li> <li>• Menunjukkan peran penting <i>website</i> dan media sosial dalam pengelolaan dan citra lembaga.</li> </ul>	<ul style="list-style-type: none"> <li>• Penelitian tidak disertai dengan implementasi sistem atau aplikasi.</li> <li>• Pembahasan bersifat deskriptif tanpa pengujian solusi secara teknis.</li> </ul>
4	Hakam, Triayudi, &	Metode Agile (Scrum)	Integrasi Agile dan Laravel terbukti	<ul style="list-style-type: none"> <li>• Menggabungkan Metode Agile (Scrum)</li> </ul>	<ul style="list-style-type: none"> <li>• Studi kasus terbatas pada satu <i>domain</i></li> </ul>

	Hayati (2022)	dikombinasikan dengan <i>Framework</i> Laravel (MVC) dalam membangun sistem manajemen zakat berbasis <i>web</i> untuk mengelola data dan transaksi.	meningkatkan efisiensi input, penyimpanan, serta pengolahan data zakat, menghasilkan sistem yang lebih cepat, responsif, dan mampu menyesuaikan kebutuhan pengguna secara berkelanjutan.	dengan <i>Framework</i> Laravel dalam pengembangan sistem. <ul style="list-style-type: none"> <li>• Pendekatan iteratif mendukung sistem yang lebih adaptif terhadap perubahan kebutuhan.</li> </ul>	aplikasi tertentu. <ul style="list-style-type: none"> <li>• Pembahasan lebih berfokus pada fungsi sistem dibandingkan aspek nonfungsional .</li> </ul>
5	Faiz Ramadhana Surya (2025)	Metode Agile (Scrum) diterapkan dalam pengembangan <i>website</i> informasi dan ulasan pesantren berbasis <i>Framework</i> Laravel (MVC)	Hasil penelitian menunjukkan bahwa sistem mampu menyajikan informasi, pencarian, serta ulasan dan <i>rating</i> pesantren secara terstruktur, responsif, dan	<ul style="list-style-type: none"> <li>• Menggunakan <i>Framework</i> Laravel dengan arsitektur MVC dan metode Agile, sehingga pengembangan sistem lebih terstruktur dan fleksibel.</li> <li>• Sistem menyediakan</li> </ul>	<ul style="list-style-type: none"> <li>• Pengujian sistem masih terbatas pada <i>black box testing</i>, belum mencakup pengujian performa dan keamanan lanjutan.</li> <li>• Sistem hanya dikembangkan dalam bentuk <i>website</i>, belum</li> </ul>

		dengan MySQL sebagai basis data dan Filament sebagai <i>admin panel</i> .	mudah digunakan untuk membantu masyarakat memperoleh informasi yang akurat.	fitur informasi, pencarian, serta ulasan dan <i>rating</i> pengguna yang terintegrasi dalam satu platform <i>web</i> .	mendukung <i>platform mobile</i> .
--	--	---	---	--	------------------------------------



## BAB V

### KESIMPULAN

#### 5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem yang telah dilakukan, dengan menggunakan metode pengembangan perangkat lunak dan teknologi *framework* Laravel, dapat ditarik beberapa kesimpulan utama sebagai berikut:

1. Website Pesantrenly berhasil dikembangkan sebagai *platform* informasi dan ulasan pesantren berbasis *framework* Laravel (MVC) dengan dukungan Filament sebagai *admin panel*, sehingga mampu menyajikan informasi pesantren secara terstruktur, terpusat, dan mudah dikelola.
2. Hasil pengujian menunjukkan bahwa seluruh fitur sistem berjalan dengan baik, data tersimpan secara konsisten, dan alur proses sistem berfungsi stabil, sehingga Pesantrenly layak digunakan sebagai media informasi dan ulasan pesantren berbasis *web*.

#### 5.2 Saran

Berdasarkan hasil perancangan, implementasi, dan pengujian *website* informasi dan ulasan pesantren berbasis Laravel menggunakan metode Agile, terdapat beberapa saran yang dapat dijadikan acuan untuk pengembangan dan penelitian selanjutnya.

1. Pengembangan selanjutnya dapat mengarah pada pembuatan aplikasi *mobile* agar akses informasi dan ulasan pesantren menjadi lebih fleksibel dan menjangkau lebih banyak pengguna.
2. Pengujian sistem dapat ditingkatkan dengan menambahkan pengujian keamanan dan performa guna meningkatkan keandalan dan kualitas *website* informasi dan ulasan pesantren.

## DAFTAR PUSTAKA

- Amarulla, J. B. I., Jasmir, & Aryani, L. (2023). Perancangan Sistem Informasi Akademik Pada SD Xaverius 1 Kota Jambi Berbasis Web. *Jurnal Manajemen Teknologi Dan Sistem Informasi (JMS)*, 3(1), 405–416. <https://doi.org/10.33998/jms.2023.3.1.166>
- Ambara, M. P., Satya Dharma, I. G., & Suraja Antarjaya, I. N. (2025). Implementasi Metode Agile untuk Meningkatkan Efisiensi Pengembangan Sistem Informasi Tabungan Koperasi. *Jurnal Sistem Dan Informatika (Jsi)*, 42–52.
- Andy Soebroto, A. (2025). Pengembangan Sistem Informasi Profil Pondok Pesantren Berbasis Web. *Informatics and Digital Expert (INDEX)*, 7(1), 55–60. <https://doi.org/10.36423/index.v7i1.2157>
- Ariansyah, K., Prawiro, J., & Sanjaya, R. (2025). Pengaruh Ulasan Online Terhadap Keputusan Wisatawan dalam Memilih Hotel. *Jurnal Pariwisata Dan Perhotelan*, 2(2), 8. <https://doi.org/10.47134/pjpp.v2i2.3559>
- Asnawan, A. (2021). Pemberdayaan Pondok Pesantren Dan Pendidikan Formal Terhadap Santri Assunniyyah Kencong Jember Di Era Regulasi Pendidikan Nasional. *As-Sunniyyah*, 1(01), 134–158.
- Ernungtyas, N. F., Boer, R. F., & Qadrifa, S. S. (2024). The government website as user's information source: A model of user satisfaction, information, and system quality. *Informasi*, 53(2), 197–214. <https://doi.org/10.21831/informasi.v53i2.60579>
- Fauzi, C., Hamidan, R., Kacung, S., & Informatika, T. (2025). *Berbasis Web Untuk Meningkatkan Transparansi*. 9(2), 2262–2267.
- Gunawan, R., & StevanieBK, F. A. (2025). Analisis Penerapan Metode Agile dalam Pengembangan Sistem Informasi Berbasis Website: Studi Efektivitas dan Dampak Implementasi. *Dirgamaya: Jurnal Manajemen Dan Sistem Informasi*, 5(1), 9–14.
- Gunawan Utami, M., & Aghniarahma Junia, A. (2024). Perbaikan dan Perancangan

Fitur Review Produk Dalam Upaya Meningkatkan Aktivitas Pengguna Untuk Memberikan Ulasan di Aplikasi Zalora. *Jurnal Ilmiah Wahana Pendidikan*, 2024(5), 602–614. <https://doi.org/10.5281/zenodo.10544115>.

Hakam, M. A., Triayudi, A., & Hayati, N. (2022). Implementasi metode agile pada sistem manajemen zakat berbasis website dengan framework Laravel. *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, 6(1), 111–116.

Haris, M. A. (2023). Urgensi Digitalisasi Pendidikan Pesantren Di Era Society 5.0 (Peluang Dan Tantangannya Di Pondok Pesantren Al-Amin Indramayu). *Islamic Management: Jurnal Manajemen Pendidikan Islam*, 6(01), 49–64. <https://doi.org/10.30868/im.v4i02.3616>

Hartati, Y., Ipriadi, I., & Wijaya, A. H. (2023). Perancangan Sistem Informasi E-Learning pada SMA NEGERI 1 TIGO NAGARI Dengan Menggunakan Bahasa Pemrograman PHP Dan Database Mysql. *Jurnal Sistem Informasi Dan Informatika*, 1(2), 36–43. <https://doi.org/10.47233/jiska.v1i2.920>

Hermiati, R., Asnawati, A., & Kanedi, I. (2021). Pembuatan E-Commerce Pada Raja Komputer Menggunakan Bahasa Pemrograman Php Dan Database Mysql. *Jurnal Media Infotama*, 17(1), 54–66. <https://doi.org/10.37676/jmi.v17i1.1317>

Hilmi, H. ilhamsyah, M. Rafli Supiyan Ananda, & M. Rifki Firmansyah. (2025). Perancangan Sistem Informasi Company Profile Berbasis Website Pada PT. Bagus Jaya Sejahtera. *Jurnal Multidisiplin Ilmu Akademik*, 2(4), 380–386. <https://doi.org/10.61722/jmia.v2i4.5311>

Indah Melyani, R., Rosita, R., & Aji, S. (2023). Pengembangan Sistem Informasi Penggajian Berbasis Web Menggunakan Framework Laravel dengan Metode Agile Software Development. *Jurnal Sistem Informasi Akuntansi (JASIKA)*, 3(1), 31–36. <https://doi.org/10.31294/jasika.v3i01.2195>

Juriatun Nasirin, L., Jasmir, & Riyadi, W. (2023). Perancangan Sistem Informasi Penerimaan Siswa Baru Berbasis Web Pada MTS Al-Fajar Tanjung Jabung Timur. *Jurnal Manajemen Teknologi Dan Sistem Informasi (JMS)*, 3(1), 384–394.

<https://doi.org/10.33998/jms.2023.3.1.736>

Kalsum Siregar, U., Arbaim Sitakar, T., Haramain, S., Nur Salamah Lubis, Z., Nadhirah, U., & Sains dan Teknologi, F. (2024). Pengembangan database Management system menggunakan My SQL. *SAINTEK: Jurnal Sains, Teknologi & Komputer*, 1(1), 8–12.

Kurniawan, H., Fadhli, I., & Korti, S. (2025). PENERAPAN METODE AGILE DALAM PERANCANGAN WEBSITE SISTEM INFORMASI DATA POIN PELANGGARAN SISWA. *Insand Comtech: Information Science and Computer Technology Journal*, 10(1), 38–45.

Lucita, A. (2025). *SISTEM PENGELOLAAN DAN MONITORING SANTRI BERBASIS WEBSITE UNTUK PIHAK PENGASUHAN DI PONDOK PESANTREN BHRUL ULUM SUNGAILIAT*. Politeknik Manufaktur Negeri Bangka Belitung.

Mahardiansyah, A. (2024). *Sekolah Tinggi Teknologi Terpadu Nurul Fikri Rancang Bangun Sistem Informasi Tabungan Siswa Berbasis Web Menggunakan Php Framework Laravel Tugas Akhir*.

Maulidiyah, N. F. A., Singasatia, D., & Sunandar, M. A. (2022). Analisis Pengaruh User Experience Terhadap Kepuasan Pengguna Mobile Application VLive Menggunakan Model SCSL. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 2(2), 28–34. <https://doi.org/10.57152/malcom.v2i2.398>

Menrisal, M., Rezi, F., & Rahmadhani, P. (2022). Pengembangan E-Learning Menggunakan PHP Native pada SMK Muhammadiyah 1 Padang. *Jurnal Pti (Pendidikan Dan Teknologi Informasi) Fakultas Keguruan Ilmu Pendidikan Universita Putra Indonesia "Yptk" Padang*, 9, 25–30. <https://doi.org/10.35134/jpti.v9i1.106>

Pratiwi, A. D., & Heriadi, A. (2023). Sistem Informasi Manajemen Keuangan Berbasis Web Menggunakan Framework Laravel (Studi Kasus: Toko Bangunan Tiga "D" Jaya). *Jurnal Informatika Dan Multimedia*, 15(2), 1–5. <https://doi.org/10.33795/jtim.v15i2.4434>

PRAYOGA, A. R. (2024). *PERANCANGAN SISTEM INFORMASI PENJUALAN BERBASIS WEB PADA TOKO DOMORO MENGGUNAKAN PHP MYSQL*. Universitas Labuhanbatu.

Putri, A. Y., Elia Mariza, & Alimni. (2023). Pesantren Sebagai Lembaga Pendidikan Islam (Sejarah Lahirnya Pesantren Sebagai Lembaga Pendidikan Islam Di Indonesia, Sejarah Perkembangan Pesantren/ Sistem Pendidikan, Dan Perkembangannya Masa Kini. *Innovative: Journal Of Social Science Research*, 3(2), 83–96.

Putri, R., & Kurniawan, D. (2025). Transformasi Budaya Pesantren Di Era Globalisasi. *Jurnal Adab Dan Peradaban Islam*, 1(2), 53–65.

Rahayu, W. D. P., Hendriadi, A. A., & Ridwan, T. (2024). Perancangan Ui Ux Aplikasi Website Sistem Informasi Menggunakan Metode User Centered Dsign (Studi Kasus Desa Losari Kidul). *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3), 2952–2964. <https://doi.org/10.23960/jitet.v12i3.4994>

Rahma, A., & Perawironegoro, D. (2024). Kontribusi Lembaga Pendidikan Islam terhadap Pengembangan Akhlak Generasi Muda. *Jurnal Inovasi Global*, 2(11), 1687–1699. <https://doi.org/10.58344/jig.v2i11.208>

Rahmawati, L., & Sumarsono, S. (2024). Desain Pengembangan Website dengan Arsitektur Model View Controller pada Framework Laravel. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 6(4), 785–790. <https://doi.org/10.47233/jteksis.v6i4.1497>

Rakhmawati, H. (2024). *E-Commerce Sayur dan Buah dengan Metode User Centered Design & Agile Software Development*. Politeknik Negeri Bali.

Robbani, A. A., Wiyono, B. H., & Haromain, I. (2025). Rancang Bangun Aplikasi Sistem Gudang Berbasis Web Menggunakan Framework Laravel dengan Agile Development Studi Kasus pada PT XYZ. *Jurnal Informatika Terpadu*, 11(1), 63–71. <https://doi.org/10.54914/jit.v11i1.1770>

Sinaga, S. G., & Aprillia, A. (2025). Peran Online Customer Review dan Online Customer Rating terhadap Niat Beli. *JIIIP - Jurnal Ilmiah Ilmu Pendidikan*, 8(6), 5570–5578. <https://doi.org/10.54371/jiip.v8i6.7932>

Sinaga, W. M. B. B., & Firmansyah, A. (2024). Perubahan Paradigma Pendidikan di Era Digital. *Jurnal Teknologi Pendidikan*, 1(4), 10. <https://doi.org/10.47134/jtp.v1i4.492>

Sukardi, S., & Putra, T. I. (2022). Website E-Learning Menggunakan Web Responsive PHP dan Database MYSQL. *Jurnal Pedagogi Dan Pembelajaran*, 5(1), 42–49. <https://doi.org/10.23887/jp2.v5i1.42448>

Viani, D. O., No, S., & Prasetyo, J. A. (2024). Pengembangan Backend Menggunakan Metode Extreme Programming (Xp) Pada Aplikasi Reservasi Pesona Java Ijen Homestay. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3), 2799–2808. <https://doi.org/10.23960/jitet.v12i3.4913>

Wardani, S. K. (2013). Sistem informasi pengolahan data nilai siswa berbasis web pada sekolah menengah atas (SMA) Muhammadiyah Pacitan. *Indonesian Journal of Networking and Security (IJNS)*, 2(2).

Wijaya, A. F., Surya Mulyana, T. M., & Liunard, G. J. (2025). Desain User Experience (UX) Pada Website Smart City untuk Meningkatkan Aksesibilitas Layanan Publik. *Jurnal Teknologi Dan Manajemen Industri Terapan*, 4(2), 69–75. <https://doi.org/10.55826/jtmit.v4i2.582>

Wijayanti, T., Nugraha, F., & Utomo, A. P. (2022). Rancang Bangun Sistem Manajemen Pengelolaan Pengaduan Masyarakat Di Kabupaten Kudus. *Journal of Computer and Information Systems Ampera*, 3(1), 56–65. <https://doi.org/10.51519/journalcisa.v3i1.141>

Zaharuddin, M., Pradana, F., & Brata, K. C. (2020). Pengembangan Sistem Penilaian Santri Terpadu di Al-Izzah Leadership School berbasis Web. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 4(11), 3978–3986.