



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**RANCANG BANGUN SISTEM INFORMASI MONITORING
HAFALAN AL QUR'AN SANTRI (BE-HAFIZH) BERBASIS *WEB*
MENGUNAKAN *FRAMEWORK LARAVEL*:**

**STUDI KASUS DI PESANTREN TAKHASSUS TAHFIDZ AL
MA'SHUM MARDIYAH**

TUGAS AKHIR

Muhammad Rifqi Robbani

0110220100

PROGRAM STUDI TEKNIK INFORMATIKA

AGUSTUS 2024



**STT TERPADU
NURUL FIKRI**

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**RANCANG BANGUN SISTEM INFORMASI MONITORING
HAFALAN AL QUR'AN SANTRI (BE-HAFIZH) BERBASIS *WEB*
MENGUNAKAN *FRAMEWORK LARAVEL*:**

**STUDI KASUS DI PESANTREN TAKHASSUS TAHFIDZ AL
MA'SHUM MARDIYAH**

TUGAS AKHIR

STT - NF

Muhammad Rifqi Robbani

0110220100

PROGRAM STUDI TEKNIK INFORMATIKA

AGUSTUS 2024

HALAMAN PERNYATAAN ORISINALITAS

Skripsi / Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Muhammad Rifqi Robbani

NIM : 0110220100

Depok, 29 Juli 2024

Tanda Tangan



Muhammad Rifqi Robbani

STT - NF

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Muhammad Rifqi Robbani

NIM : 0110220100

Program Studi : Teknik Informatika

Judul Skripsi : Rancang Bangun Sistem Informasi Monitoring Hafalan
Al-Qur'an Santri (Be-Hafizh) Berbasis Web
Menggunakan *Framework Laravel*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJI

Pembimbing

Penguji



(Nasrul, S.Pd.I., S.Kom., M.Kom.)



(Bambang Harie Wiyono, S.T., M.M.)

Ditetapkan di : Depok

Tanggal : 29 Juli 2024

KATA PENGANTAR

Dengan penuh rasa syukur, penulis mengungkapkan rasa terima kasih kehadirat Allah Subhanahu Wata'ala, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

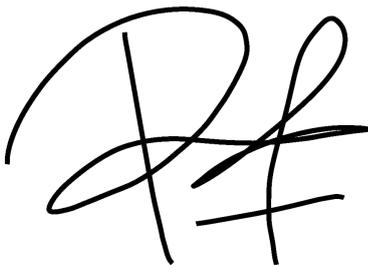
1. Allah Subhanahu Wata'ala yang telah memberikan rahmat dan petunjuk sehingga penulis berhasil menyelesaikan tugas akhir ini.
2. Bapak Yulianto dan Ibu Eti Setiowati selaku orang tua serta semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun material dalam penyelesaian tugas akhir ini.
3. Bapak Dr. Lukman Rosyidi, S.T., M.M., M.T., selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Tiffany Nabarian S.kom, M.T.I selaku Ketua Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Bapak Zaki Imaduddin S.T, M.Kom selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama berkuliah di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Bapak Nasrul, S.Kom., M.Kom selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
7. Ibu Nurul Janah S.IIP. M.Hum selaku Dosen Pembimbing Mata Kuliah Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.

8. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
9. Yayasan Al Ma'shum Mardiyah Ustadz Taufiq Hidayatullah S.pdi beserta karyawan yang telah meluangkan waktunya untuk memberikan data yang diperlukan bagi penulisan ilmiah ini.

Dalam penulisan karya ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Pada akhirnya, penulis mengharapkan Allah Subahanahu Wata'ala berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 29 Juli 2024



Muhammad Rifqi Robbani

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda di bawah ini:

Nama : Muhammad Rifqi Robbani

NIM : 0110220100

Program Studi : Teknik Informatika

Jenis karya : Skripsi / Tugas Akhir

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty - Free Right*)** atas karya ilmiah saya yang berjudul :

Rancang Bangun Sistem Informasi Monitoring Hafalan Al Qur'an Santri (Be-Hafizh)
Berbasis *Web* Menggunakan *Framework Laravel*

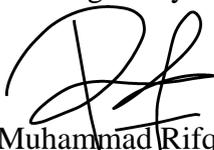
beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya

Dibuat di : Cianjur

Pada tanggal : 29 Juli 2024

Yang Menyatakan



(Muhammad Rifqi Robbani)

ABSTRAK

Nama : Muhammad Rifqi Robbani
NIM : 0110220100
Program Studi : Teknik Informatika
Judul : Rancang Bangun Sistem Informasi Monitoring Hafalan Al
Qur'an Santri(Be-Hafizh) Berbasis *Web* Menggunakan
Framework Laravel

Aplikasi Be-Hafizh merupakan salah satu aplikasi penginputan data setoran hafalan Al-Qur'an. Dalam konteks penelitian ini penulis terdapat kendala yang dihadapi dilapangan dalam melengkapi data-data. Kendala pertama adalah pengisian secara manual dalam penginputan data setoran hafalan Al-Qur'an. Dari permasalahan tersebut dibutuhkan digitalisasi dalam penginputan data setoran hafalan Al-Qur'an. Dalam membangun aplikasi Be-Hafizh ini menggunakan *framework Laravel*. Penelitian ini dari hasil observasi yang dilakukan dengan menggunakan metode kualitatif dan didapatkan score persentasi yang diberikan memiliki interpretasi yang baik sebesar 100%. Harapannya, dengan adanya hasil penelitian ini mampu memberikan solusi dalam penginputan data setoran hafalan Al-Qur'an yang diusulkan dalam mengatasi kendala pengisian manual, mempercepat proses penginputannya. Kesimpulan dari penelitian ini adalah dengan pendekatan *Extreme programming (XP)*, *filament*, dan memanfaatkan *framework laravel*, pengembangan aplikasi Be-hafizh ini diharapkan dapat memberikan solusi yang efektif dalam mengatasi tantangan yang dihadapi dalam penginputan data setoran hafalan Al-Qur'an, sehingga mempermudah pihak manajemen Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah untuk mengelola penginputan data setoran dan memudahkan dalam melakukan penginputan data setoran hafalan Al-Qur'an bulanan.

Kata kunci: *framework Laravel*, aplikasi be-hafizh, setoran hafalan al-quran, *extreme progamming (XP)*, *filament*

ABSTRACT

Name : Muhammad Rifqi Robbani

NIM : 0110220100

Study Program : Informatics Engineering

Title : *Design and Development of a Web-Based Student Memorization Al-Qur'an Monitoring Information System Using the Laravel Framework(Be-Hafizh)*

The Be-Hafizh application is an application for inputting data for memorizing the Al-Qur'an. In the context of this research, the author faced obstacles in the field in completing the data. The first obstacle is manually filling in data for memorizing Al-Qur'an deposits. Due to this problem, digitalization is needed in inputting deposit data for memorizing the Koran. In building the Be-Hafizh application using the Laravel framework. This research is based on the results of observations carried out using qualitative methods and the percentage score given has a good interpretation of 100%. The hope is that the results of this research will be able to provide a solution for inputting the proposed Al-Qur'an memorization deposit data to overcome the obstacles to manual filling, speeding up the input process. The conclusion of this research is that with the Extreme programming (XP) approach, filament, and utilizing the Laravel framework, the development of the Be-hafizh application is expected to provide an effective solution in overcoming the challenges faced in inputting Al-Qur'an memorization deposit data, making it easier the management of the Takhasuss Tahfidz Al-Ma'shum Mardiyah Islamic Boarding School to manage the input of deposit data and make it easier to input monthly Al-Qur'an memorization deposit data.

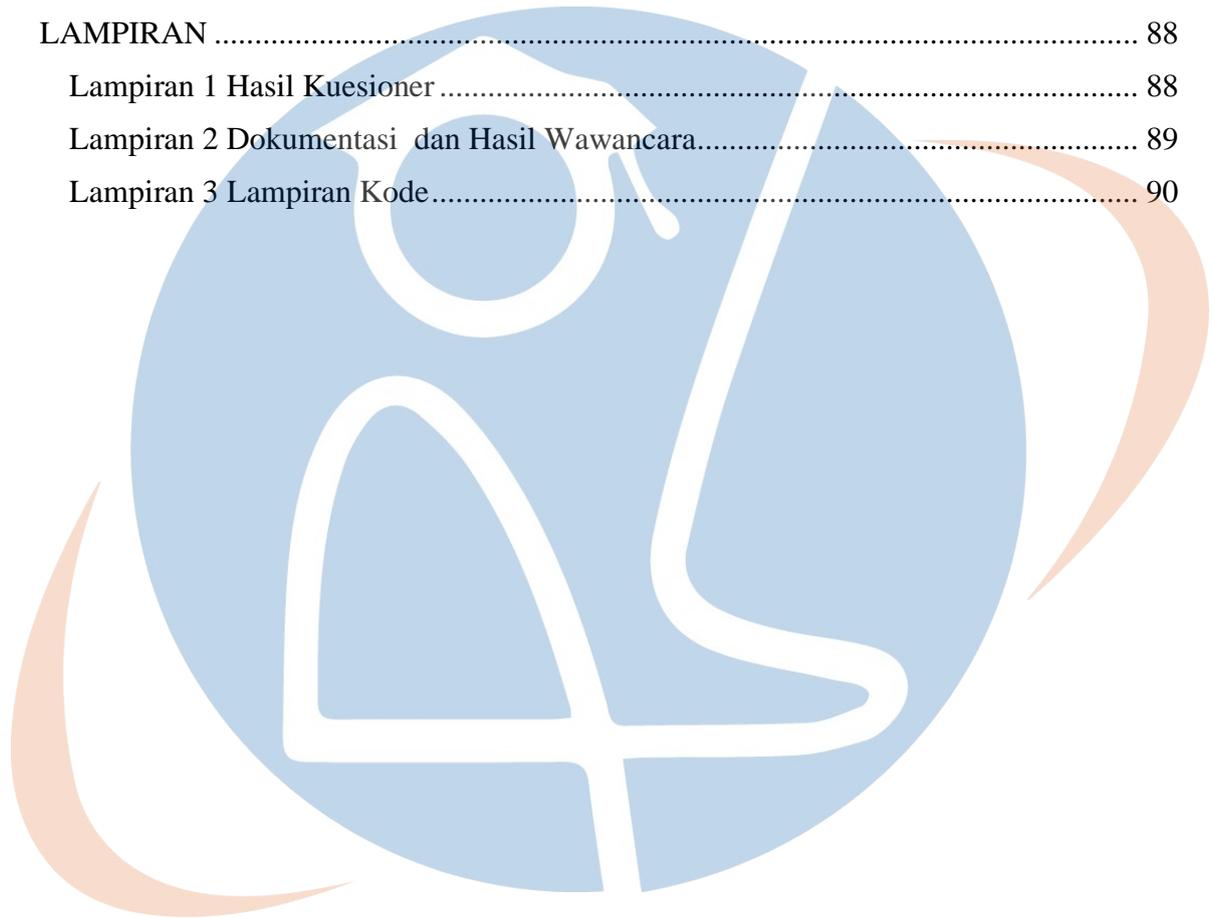
Key words: Laravel framework, be-hafizh application, al-Quran memorization deposit, extreme programming (XP), filament

DAFTAR ISI

| | |
|---|------|
| HALAMAN PERNYATAAN ORISINALITAS | I |
| HALAMAN PENGESAHAN | II |
| KATA PENGANTAR | III |
| HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI | V |
| TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS | V |
| ABSTRAK..... | VI |
| <i>ABSTRACT</i> | VII |
| DAFTAR ISI | VIII |
| DAFTAR TABEL | XI |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 2 |
| 1.3 Tujuan Penelitian | 2 |
| 1.4 Manfaat Penelitian | 2 |
| 1.5 Batasan Masalah | 3 |
| 1.6 Sistematika Penulisan | 3 |
| BAB II KAJIAN LITERATUR..... | 5 |
| 2.1 Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah | 5 |
| 2.2 Sistem Informasi | 5 |
| 2.3 <i>Framework</i> | 6 |
| 2.4 <i>Laravel</i> | 6 |
| 2.5 <i>Filament</i> | 6 |
| 2.6 <i>Model, View, Controller (MVC)</i> | 7 |
| 2.7 <i>Database MariaDB</i> | 7 |
| 2.8 <i>Extreme Progamming</i> | 8 |
| 2.9 <i>Unified Modeling Language</i> | 8 |
| 2.10 <i>Class Diagram</i> | 9 |
| 2.11 <i>Use Case Diagram</i> | 9 |
| 2.12 <i>Activity Diagram</i> | 10 |

| | | |
|---|---|-----------|
| 2.13 | <i>User Acceptance Testing</i> | 10 |
| 2.14 | Metode Implementasi dan Evaluasi | 11 |
| 2.15 | Penelitian Terkait | 11 |
| 2.16 | Posisi Penelitian | 13 |
| BAB III METODOLOGI PENELITIAN | | 15 |
| 3.1 | Tahapan Penelitian..... | 15 |
| 3.2 | Studi Pendahuluan | 16 |
| 3.3 | Analisa Perancangan..... | 16 |
| 3.4 | Implementasi..... | 16 |
| 3.5 | Pengujian..... | 16 |
| 3.6 | Kesimpulan | 17 |
| 3.7 | Rancangan Penelitian..... | 17 |
| 3.7.1 | Kualitatif..... | 17 |
| 3.8 | Jenis Penelitian..... | 18 |
| 3.9 | Metode Analisis Data..... | 18 |
| 3.10 | Metode Pengumpulan Data..... | 19 |
| 3.10.1 | Wawancara | 19 |
| 3.10.2 | Observasi..... | 19 |
| 3.10.3 | Studi Kepustakaan | 19 |
| 3.11 | Lingkungan Pengembangan..... | 20 |
| 3.12 | Lokasi Penelitian..... | 20 |
| 3.14 | Metode Pengujian | 21 |
| 3.15 | <i>Blackbox Testing</i> | 21 |
| 3.16 | Metode Implementasi dan Evaluasi | 22 |
| BAB IV IMPLEMENTASI DAN EVALUASI | | 23 |
| 4.1 | Cara Kerja Website | 23 |
| 4.2 | <i>Use Case Diagram</i> | 24 |
| 4.3 | <i>Class Diagram</i> | 26 |
| 4.4 | <i>Activity Diagram Admin</i> | 30 |
| 4.5 | <i>Activity Diagram Orang Tua</i> | 31 |
| 4.6 | <i>Activity Diagram Ustadz/Guru</i> | 32 |
| 4.7 | Implementasi..... | 33 |

| | |
|--|----|
| 4.5.4 Pengujian <i>User Acceptance Test</i> (UAT)..... | 72 |
| BAB V KESIMPULAN DAN SARAN | 82 |
| 5.1 Kesimpulan | 82 |
| 5.2 Saran | 82 |
| DAFTAR PUSTAKA..... | 84 |
| LAMPIRAN | 88 |
| Lampiran 1 Hasil Kuesioner | 88 |
| Lampiran 2 Dokumentasi dan Hasil Wawancara..... | 89 |
| Lampiran 3 Lampiran Kode..... | 90 |

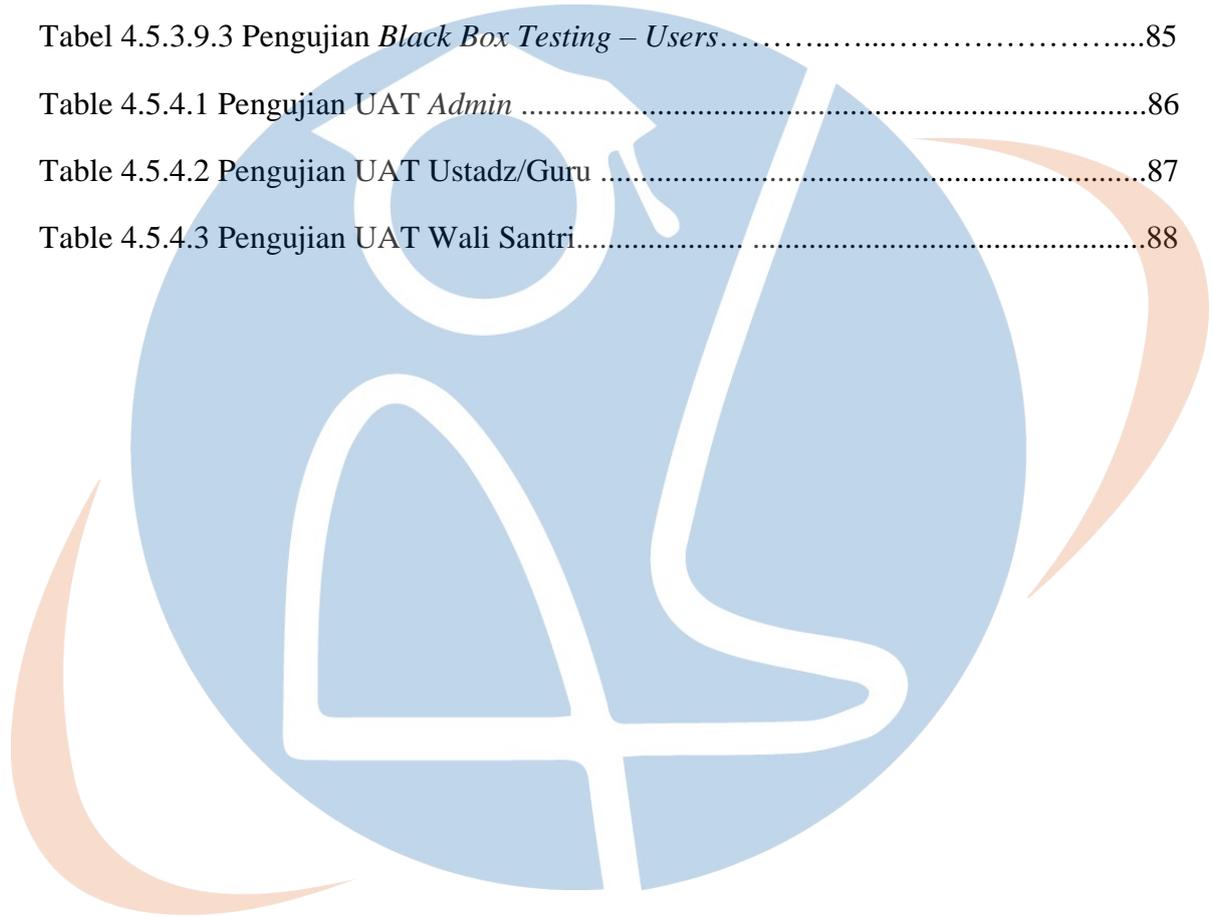


STT - NF

DAFTAR TABEL

| | |
|---|----|
| Tabel 2.1 Penelitian Terkait..... | 26 |
| Tabel 2.2 Posisi Penelitian..... | 28 |
| Tabel 4.5.2.1 Pengujian Halaman Daftar | 51 |
| Tabel 4.5.2.2 Pengujian Halaman <i>Login</i> | 52 |
| Tabel 4.5.2.3 Pengujian <i>Admin</i> Ustadz..... | 52 |
| Tabel 4.5.2.4 Pengujian <i>Form</i> Nilais..... | 54 |
| Tabel 4.5.2.5 Pengujian Hafalans..... | 54 |
| Tabel 4.5.2.6 Pengujian Kategoris..... | 56 |
| Tabel 4.5.2.7 Pengujian Kehadirans..... | 57 |
| Tabel 4.5.2.8 Pengujian Kelompok..... | 58 |
| Tabel 4.5.2.9 Pengujian Nilais..... | 59 |
| Tabel 4.5.2.9.1 Pengujian Santris..... | 60 |
| Tabel 4.5.2.9.2 Pengujian Surats..... | 61 |
| Table 4.5.2.9.3 Pengujian Ujians..... | 62 |
| Table 4.5.2.9.4 Pengujian <i>Users</i> | 63 |
| Table 4.5.2.9.5 Perkembangan Santri..... | 70 |
| Tabel 4.5.3.1 Pengujian <i>Black Box Testing</i> – <i>Login Admin</i> | 71 |
| Tabel 4.5.3.2 Pengujian <i>Black Box Testing</i> – Ustadz..... | 72 |
| Tabel 4.5.3.3 Pengujian <i>Black Box Testing</i> – <i>Form</i> Nilais..... | 74 |
| Tabel 4.5.3.4 Pengujian <i>Black Box Testing</i> – Hafalans..... | 75 |
| Tabel 4.5.3.5 Pengujian <i>Black Box Testing</i> – Kategoris..... | 76 |
| Tabel 4.5.3.6 Pengujian <i>Black Box Testing</i> – Kehadirans..... | 77 |
| Tabel 4.5.3.7 Pengujian <i>Black Box Testing</i> – Kelompok..... | 78 |

| | |
|--|----|
| Tabel 4.5.3.8 Pengujian <i>Black Box Testing</i> – Nilais..... | 79 |
| Tabel 4.5.3.9 Pengujian <i>Black Box Testing</i> – Santris..... | 82 |
| Tabel 4.5.3.9.1 Pengujian <i>Black Box Testing</i> – Surats..... | 83 |
| Tabel 4.5.3.9.2 Pengujian <i>Black Box Testing</i> – Ujians..... | 84 |
| Tabel 4.5.3.9.3 Pengujian <i>Black Box Testing</i> – Users..... | 85 |
| Table 4.5.4.1 Pengujian UAT Admin | 86 |
| Table 4.5.4.2 Pengujian UAT Ustadz/Guru | 87 |
| Table 4.5.4.3 Pengujian UAT Wali Santri..... | 88 |



STT - NF

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Tahap-tahap <i>Extreme Programming (XP)</i> | 24 |
| Gambar 3.1 Tahapan Penelitian..... | 31 |
| Gambar 4.1 <i>Flowchart Admin</i> | 40 |
| Gambar 4.2 <i>Use Case Diagram</i> | 41 |
| Gambar 4.3 <i>Class Diagram</i> | 43 |
| Gambar 4.4 <i>Activity Diagram Admin</i> | 45 |
| Gambar 4.5 <i>Activity Diagram Orang Tua</i> | 46 |
| Gambar 4.6 <i>Activity Diagram Ustadz</i> | 47 |
| Gambar 4.5.1 Halaman <i>Login Laravel</i> | 48 |
| Gambar 4.5.2 Halaman Beranda <i>Laravel</i> | 49 |
| Gambar 4.5.3 Halaman Monitoring Hafalan Al Qur'an | 49 |
| Gambar 4.5.4 Halaman Profil Pondok Pesantren Al – Ma'shum Mardiyah..... | 50 |
| Gambar 4.5.5 Halaman Informasi Kontak Pondok Pesantren Al – Ma'shum Mardiyah..... | 50 |
| Gambar 4.5.6 Halaman <i>Admin Ustadz</i> | 64 |
| Gambar 4.5.7 Halaman <i>Dashboard Admin</i> | 65 |
| Gambar 4.5.8 Halaman Menambahkan Ustadz atau Guru | 65 |
| Gambar 4.5.9 Halaman Menambahkan Hafalan Al-Quran | 66 |
| Gambar 4.6.0 Halaman Menambahkan Kategori | 66 |
| Gambar 4.6.1 Halaman Menambahkan Kehadiran Santri..... | 67 |
| Gambar 4.6.2 Halaman Menambahkan Kelompok Setoran..... | 67 |
| Gambar 4.6.3 Halaman Menambahkan Nilai..... | 68 |
| Gambar 4.6.4 Halaman Menambahkan Surat Hafalan Al-Quran..... | 68 |

Gambar 4.6.5 Halaman Menambahkan Ujian..... 69

Gambar 4.6.6 Halaman Menambahkan *User*..... 69



BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi yang cepat memungkinkan kita untuk melakukan banyak hal dengan lebih mudah dalam kehidupan modern. Salah satu contohnya adalah perkembangan teknologi dalam bidang Pendidikan. Teknologi informasi termasuk salah satu contoh nyata dari kemajuan teknologi yang telah dimanfaatkan secara luas dalam mengelola berbagai macam data, seperti pengumpulan, penyimpanan, pengiriman, dan manipulasi data untuk menghasilkan informasi yang bermanfaat bagi pengguna.

Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah merupakan Lembaga yang bergerak dalam bidang khusus para penghafal Al Qur'an. Setiap harinya tidak luput dengan bacaan Al Quran dan diajarkan Akhlak Al Quran dan di pesantren tersebut mempunyai target yang harus dicapai oleh para santrinya selama kurang waktu 6 tahun 30 juz.

Dalam penyelenggaranya di pesantren tersebut setiap bulannya guru-guru melakukan rekapitulasi hafalan dengan menggunakan *Microsoft Word*, *Microsoft Excel* secara manual sehingga data-data rekapitulasinya masih sering berantakan sehingga membutuhkan sebuah solusi yang bisa menampung rekapitulasi setiap bulan.

Dikarenakan itu guna mengukur tingkat keberhasilan be-hafizh ini maka yang perlu dilakukan yaitu, melihat penghubung antar aplikasi be-hafizh ini berjalan dengan baik sesuai dengan kebutuhan pengguna, seperti halaman beranda aplikasi be-hafizh halaman monitoring , halaman tentang kami, halaman info kami, halaman *login*, halaman *dashboard*, halaman pengujian pengisian santri, halaman pengujian pengisian ustadz, serta halaman pengujian lainnya.

Oleh karena itu, peneliti ini diarahkan untuk merancang Sistem Informasi Monitoring Hafalan Al Qur'an Santri guna memudahkan guru-guru, wali santri

melihat perkembangan santrinya atau anaknya. Solusi ini diharapkan akan memberikan kemudahan, fleksibilitas, serta meningkatkan rekapitulasi kedepannya. Maka penulis memutuskan untuk memberi judul tugas akhir ini dengan judul “ **Rancang Bangun Sistem Informasi Monitoring Hafalan Al Qur’an Santri (Be-Hafizh) Berbasis Web Menggunakan *Framework Laravel*** ”

1.2 Rumusan Masalah

1. Bagaimana rancang bangun be-hafizh berbasis *web* menggunakan *framework laravel* ?
2. Apakah Sistem Informasi yang dikembangkan dapat berjalan dengan baik dan sesuai dengan kebutuhan pengguna?

1.3 Tujuan Penelitian

1. Mengembangkan rancang bangun be-hafizh berbasis *web* menggunakan *framework laravel*.
2. Menguji sistem informasi yang telah dibuat apakah berjalan dengan baik dan sesuai dengan kebutuhan pengguna.

1.4 Manfaat Penelitian

1. Pengguna akan memperoleh pengalaman praktis dalam rancang bangun be-hafizh berbasis *web* menggunakan *framework laravel*.
2. Sebagai persyaratan untuk mendapatkan gelar sarjana.
3. Memberikan solusi kepada guru, wali santri melihat perkembangan hafalan Al Qur’an setiap bulanan.

1.5 Batasan Masalah

Penelitian ini menekankan pada pengembangan aplikasi Be-Hafizh. Batasan berikut dibuat untuk menentukan ruang lingkup penelitian:

1. Sistem informasi yang dikembangkan akan fokus pada merekapitulasi setiap bulanan dalam bentuk *Framework Laravel*.
2. Sistem Informasi akan berfokus pada Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah berupa data hafalan Al Qur'an setiap bulanan, data kelompok setoran hafalan Al Qur'an, hasil keseluruhan hafalan setiap bulan, nilai-nilai akhlak serta informasi lain yang berkaitan dengan pondok pesantren.
3. Sistem informasi yang dikembangkan ini berbentuk *web* agar memudahkan wali santri, guru, yayasan melihat perkembangan santri setiap bulanan.
4. *Tools-tools* yang digunakan dalam penelitian ini adalah *composer* serta artisan

1.6 Sistematika Penulisan

Dalam penelitian ini, metode penulisan berikut digunakan untuk mempermudah pemahaman tugas akhir secara keseluruhan:

BAB I PENDAHULUAN

Pada bab ini merupakan bab pembuka yang memberikan penjelasan tentang latar belakang, rumusan masalah, tujuan dan manfaat penelitian untuk menjawab rumusan masalah, batasan masalah, serta sistematika penulisan yang merupakan bagian untuk menjelaskan struktur kerangka dalam penyajian laporan penelitian ini.

BAB II KAJIAN LITERATUR

Pada bab ini menguraikan konsep dan teori yang berkaitan dengan penelitian, mengevaluasi temuan dari penelitian terkait, dan mengidentifikasi perbedaan penelitian ini dengan studi sebelumnya yang telah dilakukan oleh para ahli.

BAB III METODOLOGI PENELITIAN

Pada bab ini akan membahas tahapan penelitian, mulai dari alur penelitian, rancangan penelitian, hingga analisis sistem yang didasarkan pada *User* dan *Admin*. Sebagai langkah akhir, rancangan pengujian akan menggunakan metode *black box*.

BAB IV IMPLEMENTASI DAN EVALUASI

Pada bab ini akan membahas perancangan *laravel* untuk be-hafizh: studi kasus di Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah. Bab ini mencakup hasil dari perancangan *laravel* yang dilakukan oleh penulis, serta pembahasan hasil pengujian dan evaluasi dari *laravel* tersebut.

BAB V KESIMPULAN DAN SARAN

Pada bab ini berisi informasi terkait kesimpulan yang menjawab tujuan penelitian, serta saran dari penulis yang sesuai untuk penelitian selanjutnya

STT - NF

BAB II

KAJIAN LITERATUR

2.1 Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah

Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah merupakan Lembaga yang bergerak dalam bidang khusus para penghafal Al Qur'an. Setiap harinya tidak luput dengan bacaan Al Quran dan diajarkan Akhlak Al Quran dan di pesantren tersebut mempunyai target yang harus dicapai oleh para santrinya selama kurang waktu 6 tahun 30 juz. Visi utamanya adalah menjadi pesantren unggulan pencetak generasi qurani. Dalam visinya, Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah melibatkan orang tua/wali santri/orang tua/wali santri dalam mewujudkan visi Yayasan, melaksanakan kurikulum pendidikan terpadu dengan sistem berasrama selama 6 tahun (tingkat menengah pertama dan atas), menanamkan akhlak karimah (qur'ani) dalam kehidupan warga pesantren, mengembangkan minat dan bakat santri, membudayakan kerja inovatif, mencapai kemandirian finansial, menjadikan lembaga lebih dikenal masyarakat. Selain itu, Pesantren Takhassus Tahfidz Al Ma'shum Mardiyah dengan fokus utama mendukung masyarakat dalam menghafal Al Quran serta keinginan untuk mencapai target hafalan Al Quran 30 juz mutqin[1].

2.2 Sistem Informasi

Sistem informasi dapat dipahami sebagai suatu himpunan atau kumpulan dari kelompok orang-orang yang bekerja, prosedur-prosedur, dan sumber daya peralatan yang mengumpulkan data dan mengolahnya menjadi informasi, serta merawat dan menyebarkan informasi dalam suatu organisasi[2].

Menyederhanakan pemahaman terhadap sistem informasi sebagai komponen-komponen dalam organisasi atau perusahaan yang berhubungan dengan proses penciptaan dan pengaliran informasi yang akan digunakan oleh satu atau lebih pemakai (*users*). Para pemakai biasanya tergabung dalam suatu entitas organisasi formal, seperti departemen atau lembaga suatu instansi pemerintahan yang dapat

dijabarkan menjadi direktorat, bidang, bagian sampai pada unit terkecil di bawahnya[3].

Sistem informasi memuat berbagai informasi penting mengenai orang, tempat, dan segala sesuatu yang ada di dalam atau di lingkungan sekitar organisasi. Informasi menjelaskan mengenai organisasi atau salah satu sistem utamanya mengenai apa yang telah terjadi pada masa lalu, apa yang sedang terjadi sekarang, dan apa yang mungkin akan terjadi pada masa yang akan datang tentang organisasi tersebut[4].

2.3 *Framework*

Framework merupakan sebuah *software* atau aplikasi yang bisa dibayangkan seperti kerangka kerja yang fungsinya untuk memudahkan *developer* dalam mengembangkan aplikasi website yang ada [5].

2.4 *Laravel*

Laravel adalah sebuah *framework PHP* yang dirilis di bawah lisensi MIT, dibangun dengan konsep *MVC (model view controller)*. *Laravel* adalah pengembangan website berbasis *MVC* yang ditulis dalam *PHP* yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu[6].

Laravel merupakan *framework* dengan versi *PHP* yang up-to-date, karena *Laravel* mensyaratkan *PHP* versi 5.3 keatas. *Laravel* merupakan *framework PHP* yang menekankan pada kesederhanaan dan fleksibilitas pada desain[7]

2.5 *Filament*

Laravel Filament terdengar seperti toolkit canggih yang dirancang untuk mempercepat pengembangan aplikasi tumpukan *TALL*, yang memanfaatkan *Tailwind CSS*, *Alpine.js*, *Laravel*, dan *Livewire*[8]. Ini mencakup fitur-fitur seperti panel *admin*, pembuat formulir, dan pembuat tabel yang dirancang khusus untuk *Laravel*. Dengan *Filament*, pengembang dapat secara efisien membuat dan mengelola

sumber daya seperti pengguna dan produk hanya dengan beberapa perintah, menyederhanakan proses pengembangan, dan memungkinkan terciptanya aplikasi yang menarik secara estetika [9].

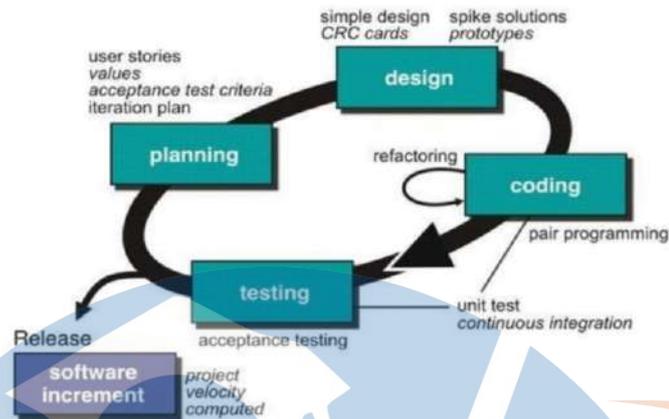
2.6 Model, View, Controller (MVC)

Model View Controller atau *MVC* merupakan cara membuat aplikasi dengan membagi data (model) dari tampilan (*view*) dan cara menanganinya (*controller*) [10]. Dari segi implementasinya, sebagian besar *framework* pada aplikasi web didasarkan pada arsitektur *MVC* (Ilmi et al., 2020). *MVC* memisahkan pengembangan aplikasi dari segi komponen utama yang menjadi dasar aplikasi dibangun, seperti manipulasi data, antarmuka pengguna, dan apa yang menjadi kontrol dalam aplikasi web.

2.7 Database MariaDB

MariaDB adalah sistem manajemen *database* relasional yang dikembangkan dari *MySQL* yang dimaksudkan untuk tetap bebas di bawah *General Public License (GPL)* [11]. Pengembangan dipimpin oleh beberapa orang yang sebelumnya berkontribusi untuk *MySQL*, karena kekhawatiran atas akuisisi oleh *Oracle Corporation*. Kontributor diwajibkan untuk berbagi hak cipta mereka dengan Yayasan *MariaDB*. *MariaDB* bermaksud untuk mempertahankan kompatibilitas yang tinggi dengan *MySQL*, memastikan kemampuan penggantian *drop-in* dengan kesetaraan biner perpustakaan dan cocok dengan *Application Programming Interface (API) MySQL* dan perintahnya. Ini termasuk mesin penyimpanan *XtraDB* untuk mengganti *InnoDB*. Serta mesin penyimpanan baru, *Aria*, yang bermaksud menjadi mesin transaksional dan non-transaksional bahkan mungkin disertakan dalam versi *MySQL* di masa mendatang [12].

2.8 Extreme Programming



Gambar 2.1 Tahap-tahap *Extreme Programming (XP)*[13]

"*Extreme Programming (XP)* adalah salah satu metodologi dalam pendekatan *Agile Software Development* yang dikenal karena pendekatannya yang berfokus pada praktik-praktik pengkodean (*coding*) ekstensif dan berulang"[14]. Pendekatan ini bertujuan untuk meningkatkan responsifitas terhadap perubahan kebutuhan pelanggan serta membangun perangkat lunak dengan kualitas yang lebih baik.

2.9 Unified Modeling Language

"*UML (Unified Modeling Language)* adalah sebuah bahasa standar yang digunakan dalam pemodelan perangkat lunak"[15]. Dengan menggunakan diagram dan notasi visual, *UML* memungkinkan para pengembang perangkat lunak untuk secara jelas menggambarkan struktur, perilaku, interaksi, dan arsitektur sistem yang mereka rancang.

Sejak diciptakannya pada tahun 1997 oleh *Object Management Group (OMG)*, *UML* telah menjadi standar *de facto* dalam dunia rekayasa perangkat lunak. *UML* memberikan sejumlah diagram yang dapat digunakan untuk merepresentasikan berbagai aspek sistem, seperti diagram kelas, *diagram sekuen*, *diagram aktivitas*, *diagram use case*, dan banyak lagi.

Penggunaan *UML* membantu para pengembang untuk memahami, berkomunikasi, dan merancang sistem perangkat lunak secara lebih efektif, terutama dalam proyek-

proyek yang kompleks atau melibatkan tim yang besar. Selain itu, *UML* juga berguna dalam dokumentasi sistem, memungkinkan untuk mempertahankan pemahaman yang konsisten tentang sistem seiring waktu.

2.10 *Class Diagram*

"*Class diagram* memang sangat penting dalam pemodelan sistem berorientasi objek karena memberikan gambaran yang jelas tentang struktur kelas, atribut, metode, dan hubungan antara objek-objek dalam sistem"[14]. Seperti yang telah disebutkan, diagram kelas bersifat statis dan berfokus pada representasi struktur sistem pada satu titik waktu tertentu.

Pembagian desain model diagram kelas menjadi dua bagian yang telah disebutkan juga sangat relevan. Bagian pertama yang berkaitan dengan penjabaran dari database memungkinkan pengembang untuk merancang basis data yang sesuai dengan struktur kelas dalam sistem. Sedangkan bagian kedua yang terkait dengan modul *MVC* (*Model-View-Controller*) menunjukkan bagaimana kelas-kelas dalam sistem akan dibagi ke dalam tiga komponen utama: model (entitas dan logika bisnis), *view* (antarmuka pengguna), dan *controller* (logika pengontrol)[15].

Dengan memahami dan menggunakan diagram kelas dengan baik, tim pengembang dapat memiliki pandangan yang jelas tentang struktur sistem mereka dan dapat memastikan bahwa desain sistem mereka sesuai dengan konsep-konsep yang mendasari pemrograman berorientasi objek[16].

2.11 *Use Case Diagram*

"*Use case diagram* merupakan salah satu jenis diagram *UML* yang sangat berguna dalam memodelkan interaksi antara sistem yang sedang dikembangkan dengan aktor-aktor eksternal yang terlibat dalam penggunaannya"[17].

Dalam *use case diagram*, aktor-aktor tersebut direpresentasikan sebagai entitas luar yang berinteraksi dengan sistem, sedangkan *use case* sendiri merupakan aksi atau fungsi yang dapat dilakukan oleh aktor tersebut. *Use case diagram* memberikan gambaran yang jelas tentang fungsionalitas sistem dari sudut pandang pengguna atau aktor[18].

Use case diagram memang merupakan langkah awal yang baik dalam proses pemodelan karena memberikan pemahaman yang mudah dipahami tentang interaksi antara sistem dan penggunaannya. Dengan menggunakan *use case diagram*, tim pengembang dapat mengidentifikasi dan mendefinisikan kebutuhan fungsional sistem dengan jelas sebelum melangkah lebih jauh dalam proses pengembangan perangkat lunak[19].

2.12 *Activity Diagram*

"*Activity diagram* atau diagram aktivitas adalah salah satu jenis diagram dalam UML yang digunakan untuk memodelkan proses-proses atau aktivitas-aktivitas yang terjadi dalam suatu sistem"[20]. Diagram ini menyajikan aliran kerja atau alur aktivitas dari suatu proses dalam sistem secara *visual*, dengan menggunakan simbol-simbol dan panah-panah untuk menggambarkan urutan dan hubungan antara aktivitas-aktivitas tersebut.

Activity diagram biasanya digunakan untuk menggambarkan alur kerja dari suatu proses atau aktivitas dalam sistem, seperti urutan menu-menu atau proses bisnis yang terjadi. Dengan *activity diagram*, pengembang perangkat lunak dapat memahami dengan jelas bagaimana proses-proses dalam sistem berjalan dan berinteraksi satu sama lain[21].

Seperti yang telah disebutkan, *activity diagram* memang fokus pada alur kerja atau aktivitas dalam sistem dan tidak menjelaskan kelakuan aktor secara detail. Dalam pemodelan dengan *activity diagram*, aktor-aktor biasanya tidak diperhatikan secara langsung, tetapi difokuskan pada urutan aktivitas atau langkah-langkah yang harus dilakukan dalam suatu proses[22].

2.13 *User Acceptance Testing*

"UAT (*User Acceptance Testing*) merupakan tahap pengujian yang dilakukan oleh pengguna akhir atau *end-user* dari suatu sistem perangkat lunak"[23]. Tujuan dari UAT adalah untuk memastikan bahwa sistem yang telah dikembangkan memenuhi kebutuhan dan harapan pengguna serta dapat diterima untuk digunakan dalam lingkungan produksi.

Kebutuhan yang dimaksud dalam konteks UAT adalah kebutuhan fungsional dan non - fungsional yang telah ditetapkan dalam spesifikasi perangkat lunak. Kebutuhan ini biasanya disepakati antara pihak pengembang dan pihak pengguna pada awal proyek pengembangan perangkat lunak. Kebutuhan fungsional mencakup fitur-fitur dan fungsi -fungsi yang diinginkan dari sistem, sedangkan kebutuhan non-fungsional mencakup atribut-atribut kualitas seperti performa, keamanan, dan ketersediaan.

Dalam UAT, pengguna melakukan pengujian untuk memverifikasi apakah sistem telah memenuhi kebutuhan fungsional dan non-fungsional yang telah disepakati sebelumnya. Pengujian ini bertujuan untuk menunjukkan bahwa sistem berperilaku sesuai dengan yang diharapkan dan dapat digunakan dengan efektif dalam lingkungan produksi.

2.14 Metode Implementasi dan Evaluasi

“Implementation” dari Grindle, sebagaimana model konseptual dan kerangka pemikiran implementasi sebagai proses politik dan proses administrasi. Kerangka pemikiran tersebut disusun atas jawaban dua pertanyaan pokok, khususnya di negara berkembang pada umumnya[24]. Pertanyaan pertama mengenai “Content” (isi) dari program itu sendiri. Pengaruh dapat terjadi karena isi program terhadap proses implementasi. Mungkin ada program yang pada dasarnya lebih sulit daripada program lain dalam proses implementasi. Pertanyaan kedua mengenai “Context” (kondisi lingkungan) yang memiliki kaitan pengaruh atau hubungan terhadap implementasi[25].

2.15 Penelitian Terkait

Tabel 1 Penelitian Terkait dengan *framework laravel*

Tabel 2.1 Penelitian Terkait

| No | Nama dan Tahun | Judul | Topik | Subjek | Hasil |
|----|----------------------|-----------------------|-----------------------|-------------------------------------|-----------------------|
| 1 | Anis Purnomo, Astria | Rancang Bangun Sistem | E- <i>Learning</i> | Santri, asatidz Taman Pendidikan | Rancang Bangun Sistem |

| | | | | | |
|---|---|--|------------------|--------------------------------------|--|
| | Hijriani, dan Ossy Dwi Endah Wulansari, 1 April 2021 | Informasi Manajemen Data Santri Pada Taman Pendidikan Al-Quran Masjid Ad-Du'a Berbasis Laravel | | Al-Quran Masjid Ad-Du'a | Informasi Manajemen Data Santri Pada Taman Pendidikan Al-Quran Masjid Ad-Du'a Berbasis Laravel |
| 2 | Sania Nur Agustina, Divi Galih Prasetyo P., S.Kom., M.Kom. 2017 | Sistem Informasi Berbasis Web Monitoring Perkembangan Hafalan Al- Qur'an Di Sma Ar- Rosyidah | Online Learning | Santri Al-Qur'an Di SMA Ar- Rosyidah | Monitoring Perkembangan Hafalan Al- Qur'an Di SMA Ar-Rosyidah |
| 3 | Muhammad Revan S, 2022 | Rancang Bangun Sistem Informasi Repository Materi Ajar Menggunakan Framework Laravel | Offline Learning | Siswa, guru, MAN INSAN CENDEKIA SIAK | Repository Materi Ajar Menggunakan Framework Laravel |

Dalam Tabel 2.1 Penelitian Terkait, dapat diketahui bahwa telah dilakukan beberapa penelitian terkait *framework laravel*. Penelitian ini merujuk pada studi-studi sebelumnya yang relevan dengan topik yang sedang diteliti, membantu mendukung, dan memberikan kontekstualisasi terhadap penelitian yang sedang dilakukan

2.16 Posisi Penelitian

Data penelitian berikut ini mencakup perbedaan dalam teknologi yang digunakan dan subjek penelitian, dibandingkan dengan penelitian sebelumnya

Tabel 2.2 Posisi Penelitian

| No | Monitoring Hafalan Santri Berbasis Web | Berbasis Web | Framework Laravel | Blackbox Testing | Extreme Progamming |
|----|---|--------------|-------------------|------------------|--------------------|
| 1. | Anis Purnomo, Astria Hijriani, dan Ossy Dwi Endah Wulansari, 1 April 2021 | | | | |
| 2. | Sania Nur Agustina, Divi Galih Prasetyo P., S.Kom., M.Kom. 2017 | | | | |
| 3. | Muhammad Reva S, 2022 | | | | |
| 4. | Muhammad Rifqi Robbani, 2024 | | | | |

Dalam Tabel 2.2 Posisi Penelitian menunjukkan bahwa beberapa penelitian mengadopsi teknologi dan metode pengembangan sistem informasi yang serupa.

Berikut adalah penjelasan lebih rinci mengenai setiap penelitian yang mencakup persamaan dalam pendekatan teknologi dan metodologi pengembangan yang digunakan.

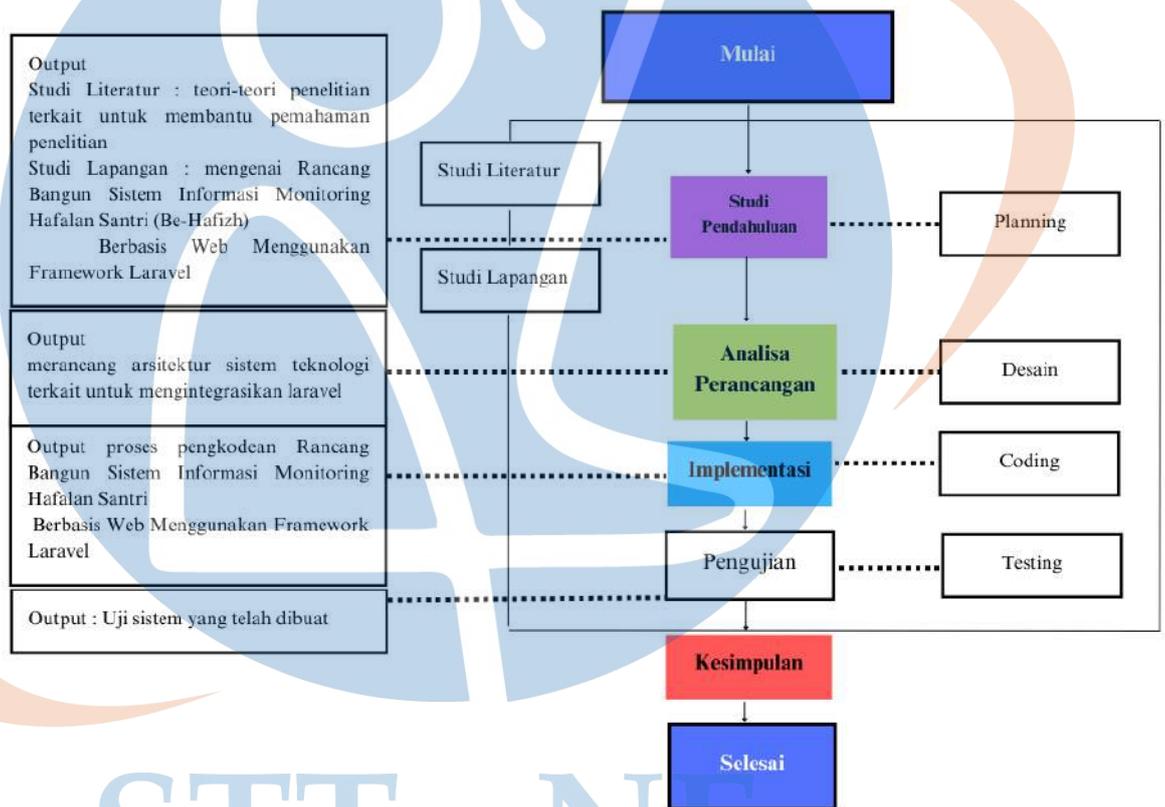
1. Anis Purnomo, Astria Hijriani, dan Ossy Dwi Endah Wulansari melakukan penelitian tentang penerapan sistem informasi manajemen data santri untuk mencapai indikator kinerja utama di Taman Pendidikan Al Quran Masjid Ad-Du'a. Mereka memilih *framework laravel* dalam pengembangan sistem ini. Fokus penelitian mereka adalah pada pengembangan *E-Learning* dengan menggunakan *framework Laravel*.
2. Sania Nur Agustina, Divi Galih Prasetyo P., S.Kom., M.Kom. melakukan penelitian tentang penerapan Sistem Informasi Berbasis Web Monitoring Perkembangan Hafalan Al-Qur'an untuk mencapai indikator kinerja utama di SMA Ar-Rosyidah. Mereka memilih menggunakan berbasis web monitoring dalam pengembangan sistem ini. Fokus penelitian mereka adalah pada pengembangan *online learning* dengan menggunakan berbasis web.
3. Muhammad Revan S, melakukan penelitian tentang penerapan Sistem Informasi *Repository* Materi Ajar untuk mencapai indikator kinerja utama di MAN INSAN CENDEKIA SIAK. Mereka memilih menggunakan *framework laravel* dan *extreme programming* tanpa menggunakan *blackbox testing*. Fokus penelitian mereka adalah pada pengembangan *offline learning* dengan menggunakan *framework laravel* dan *extreme programming*.

Dari ringkasan penelitian di atas, dapat disimpulkan bahwa belum ada penelitian yang membahas penmbangunan menggunakan *blackbox testing* untuk sistem informasi monitoring hafalan al qur'an santri dengan menggunakan *framework laravel*, metode *extreme progamming*, dan *blackbox testing*. Oleh karena itu, penelitian berjudul “ Rancang Bangun Sistem Informasi Monitoring Hafalan Al Qur'an Santri (Be-Hafizh) Berbasis Web Menggunakan *Framework Laravel*” akan mengisi kekosongan tersebut pada pengembangan *blackbox testing* untuk aplikasi Be-Hafizh dengan teknologi dan pendekatan tersebut

BAB III METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Dalam penelitian ini, penulis menjalani berbagai tahapan sesuai dengan tujuan penelitian. Proses tahapan penelitian ini melibatkan alur yang diuraikan secara rinci untuk setiap proses yang telah disusun, sehingga penelitian dapat dilaksanakan dengan jelas dan terstruktur.



Gambar 3.1 Tahapan Penelitian[28]

Pada Gambar 3.1 Tahapan Penelitian ini menerapkan metodologi *Extreme Programming (XP)* untuk pengembangan aplikasi be-hafizh. *Extreme Programming (XP)* adalah suatu proses rekayasa perangkat lunak yang umumnya mengadopsi pendekatan berorientasi objek. Metodologi ini cocok digunakan

Ketika tim dihadapkan pada kebutuhan yang tidak jelas atau perubahan kebutuhan yang cepat

3.2 Studi Pendahuluan

Dalam studi pendahuluan ini, dilakukan studi literatur untuk memahami teknologi dan pendekatan terkini dalam informasi dari resume secara langsung. Fokusnya adalah mengidentifikasi solusi terkait informasi website Yayasan secara langsung. Selain itu, dilakukan studi lapangan dengan mengamati langsung website Yayasan untuk mendapatkan wawasan praktis dalam konteks penelitian.

3.3 Analisa Perancangan

Dalam merancang, tahapan perancangan ini melibatkan rancangan arsitektur sistem untuk pengembangan aplikasi Be-Hafizh dari resume pengguna, dengan indikator capaian seperti pemilihan penggunaan menggunakan *Laravel*, pilihan dan justifikasi penggunaan kerangka kerja *controller*, *middleware* dan *request* dalam pengembangannya, serta perancangan hasil dalam bentuk publik setelah pengguna menjalankan *laravel*.

3.4 Implementasi

Dalam tahap ini, peneliti akan memulai dengan menulis kode dalam Bahasa pemrograman PHP secara dinamis dan fleksibel. Indikator capaian yang menjadi fokus pada tahap ini mencakup konfigurasi *MVC(Model-View- Controller)* sesuai dengan kebutuhan, implementasi layanan informasi dari resume, pengembangan algoritma nya yang mampu cepat, relatif mudah bagi pengembangan PHP, Skalabilitas yang baik untuk aplikasi menengah, kinerja sedang, ekosistem *laravel* yang besar dan aktif, mendukung banyak *database* melalui *ORM(Eloquent)*, aktif dan mendukung pengguna, lebih populer dengan *CMS* dan sistem pengembangan web.

3.5 Pengujian

Pengujian dilakukan terhadap sistem yang telah dikembangkan menggunakan *MVC(Model-View-Controller)* untuk memastikan akurasi yang tepat. Pengujian melibatkan Bahasa pemrograman *PHP*, *database*, direktori berfungsi dengan baik. Aplikasi Be-Hafizh memulai serangkaian aktivitas untuk memverifikasi

bahwa tujuan penelitian telah tercapai melalui pengujian fungsional.

3.6 Kesimpulan

Langkah terakhir melibatkan penyusunan kesimpulan dari hasil penelitian dan pengembangan. Peneliti menjelaskan solusi yang telah diimplementasikan untuk mengatasi masalah yang ada. Kesimpulan ini mencakup evaluasi keberhasilan dalam menangani masalah, pencapaian tujuan yang telah ditetapkan

3.7 Rancangan Penelitian

Rancangan penelitian menjelaskan tentang rencana yang akan dilakukan dalam penelitian ini. Rancangan penelitian memaparkan tentang pendekatan untuk mengatasi tantangan yang dihadapi, jenis penelitian yang dipilih, dan metode pengumpulan data. Maka dari itu rancangan penelitian yang akan digunakan dalam penelitian sebagai berikut:

3.7.1 Kualitatif

Penelitian Kualitatif adalah metode penelitian yang menggunakan data deskriptif berupa Bahasa tertulis atau lisan dari orang dan pelaku yang dapat diamati. Penelitian Kualitatif memiliki rancangan penelitiannya di antaranya : 1. Tahap pra-lapangan, 2 tahap kegiatan lapangan, 3 tahap pasca-lapangan. Berikut penjelasannya:

1. Tahap Pra-lapangan

Peneliti menaruh minat dan kepedulian terhadap Pondok Pesantren Takhassus Tahfizd Al Ma'shum Mardiyah terkait masih manual menggunakan *Microsoft word*, *Microsoft Excel* ketika melakukan pengisian data-data pondok pesantren diantaranya: merekapitulasi hafalan setiap bulanan, merekapitulasi nilai-nilai setiap bulanan.

Pengamatan sepintas sudah dilakukan jauh sebelum rancangan penelitian disusun dan diajukan sebagai topik penelitian.

Berbekal pengamatan awal dan telaah pustaka, peneliti mengajukan usulan penelitian tentang membuat aplikasi Be-Hafizh.

2. Tahap Kegiatan Lapangan

Sepanjang pelaksanaan penelitian ketika kegiatan dilapangan sedang berlangsung banyak data-data hafalan al quran santri yang berserakan tidak tersusun rapih

Dikarenakan oleh itu maka peneliti muncul ide membuat sebuah aplikasi be-hafizh guna merapihkan data-data hafalan al quran santri.

3. Tahap Pasca-lapangan

Telah disinggung bahwa penelitian ini menerapkan metode kualitatif yang dimana peneliti mengambil data secara langsung dilapangan ketika pasca-lapangan telah berlangsung.

4. Hasil *Kuesioner*

Telah disinggung bahwa penelitian ini menerapkan metode kualitatif yang dimana peneliti mengambil data secara langsung dilapangan ketika pasca-lapangan telah berlangsung dan menggunakan kuesioner.

3.8 Jenis Penelitian

Jenis penelitian yang akan dilakukan dalam penelitian ini sebagai penelitian pengembangan (*development research*). Penelitian pengembangan fokus pada pengembangan suatu produk, sistem, atau teknologi dengan tujuan memberikan solusi konkret terhadap tantangan yang dihadapi, dalam konteks ini yaitu aplikasi Be-Hafizh . Penelitian ini mencakup perancangan, pengembangan, dan pengujian suatu teknologi yang mampu memberikan solusi terhadap tantangan yang dihadapi oleh *Platform* Yayasan Al Ma'shum.

3.9 Metode Analisis Data

Metode analisis data penelitian yang diterapkan dalam penelitian ini adalah metode kualitatif yang dipilih dengan pertimbangan yang cermat. Keputusan ini diambil karena penelitian ini melibatkan tempat dimana mengajar ,memungkinkan para responden untuk memberikan tanggapan secara langsung melalui wawancara.

Selama pelaksanaan survei, data statistik secara sistematis dikumpulkan,

diolah, dan dianalisis. Penggunaan metode kualitatif memberikan keunggulan dalam proses pengumpulan data bersifat fleksibel sesuai keadaan di lapangan, memungkinkan peneliti untuk mendapatkan gambaran yang lebih jelas tentang persepsi pengguna terhadap kualitas dan kinerja sistem informasi yang menjadi fokus penelitian.

3.10 Metode Pengumpulan Data

Pada penelitian ini, tahapan pengumpulan data dan informasi melalui sejumlah metode, seperti:

3.10.1 Wawancara

Metode wawancara memang menjadi salah satu teknik pengumpulan data yang penting dalam penelitian kualitatif. Dengan melakukan wawancara, peneliti dapat mendapatkan pemahaman yang lebih dalam tentang sudut pandang, pengalaman, dan pemikiran responden terkait dengan topik penelitian yang sedang diteliti. Kelebihan dari metode wawancara adalah fleksibilitasnya yang memungkinkan peneliti untuk menyesuaikan pertanyaan-pertanyaan sesuai dengan respons dan arah percakapan yang berkembang. Selain itu, melalui wawancara, peneliti juga dapat mengeksplorasi konteks sosial dan budaya yang memengaruhi pandangan responden terhadap topik yang diteliti.

3.10.2 Observasi

Observasi dilakukan dengan menganalisis langsung *platform* Yayasan Al Ma'shum yang melibatkan pengamatan langsung terhadap objek penelitian terkait permasalahan yang akan dibahas. Ini melibatkan identifikasi proses pada *platform* Yayasan Al Ma'shum sebagai acuan untuk menentukan data yang harus digunakan dari *resume*, sehingga dapat digunakan dalam aplikasi be-hafizh .

3.10.3 Studi Kepustakaan

Dalam pengembangan layanan mencari informasi langsung di *platform* Yayasan Al Ma'shum, studi kepustakaan memegang peran penting. Studi kepustakaan dilakukan dengan cermat untuk mengidentifikasi dan

mengakses sumber-sumber pustaka yang mendukung penelitian ini. Sumber-sumber literatur yang relevan termasuk buku, jurnal, artikel, dan makalah dalam berbagai jenis.

3.11 Lingkungan Pengembangan

Lingkungan Pengembangan merujuk pada lokasi penelitian, rangkaian alat, perangkat lunak, dan sumber daya lain yang diterapkan oleh pengembang perangkat lunak atau aplikasi untuk membuat, menguji, dan menyusun kode. Lingkungan pengembangan mencakup berbagai elemen yang mendukung proses pengembangan perangkat lunak dari awal hingga selesai. Maka dari itu berikut penjelasannya :

1 Lokasi Penelitian

Jl. Simpang Galudra No.76, Galudra, Kec. Cugenang, Kabupaten Cianjur, Jawa Barat 43252 (Pondok Pesantren Takhassus Tahfizh)

3.12 Lokasi Penelitian

Penelitian ini dilakukan dengan menjadikan website Yayasan Al Ma'shum sebagai rujukan yang akan diamati secara *online* dan di tempat langsung saya mengajar. Pengamatan dilakukan dengan mengakses website Yayasan Al Ma'shum langsung secara *online* dan *offline* sebagai pengguna baru. Temuan utama menunjukkan bahwa saat ini belum ada fitur *framework laravel*, dan ini menjadi latar belakang yang relevan untuk pengembangan penelitian.

3.13 Alat dan Bahan

Berikut adalah penjelasan mengenai alat-alat yang digunakan dalam penelitian ini, serta uraian tentang bahan-bahan yang diperlukan untuk pengumpulan data dengan cermat:

1. Kebutuhan Alat

Dalam lingkup penelitian tugas akhir ini, perangkat keras yang dipakai adalah sebuah laptop dengan rincian spesifikasi sebagai berikut:

- a HP *EliteBook Folio 1040 G2 Type 23H2*
- b Intel® Core™ i5-5200U CPU @2.20GHz 2.20 GHz

- c Ram 8GB
- d *System Type 64-bit operating system, x64-based processor*

2. Kebutuhan Bahan

Dalam lingkup penelitian tugas akhir ini, melibatkan penggunaan berbagai perangkat lunak dengan spesifikasi sebagai berikut:

- a Editor : *Visual Studio Code*
- b Database : *PHP , Mysql*
- c Framework : *Laravel*

3.14 Metode Pengujian

Pada pengujian ini merupakan suatu mekanisme yang digunakan untuk mengidentifikasi data uji yang mampu menguji perangkat lunak yang dikembangkan. Pengujian dilakukan dengan menggunakan *User Acceptance Testing (UAT)*. UAT akan melibatkan pengujian kemampuan *framework* dalam mengidentifikasi dan mengekstrak data kunci, seperti informasi dasar, riwayat pendidikan, dan keterampilan. Sebelum ditahap UAT, peneliti menggunakan beberapa metode pengujian untuk mendapatkan hasil sesuai dengan penelitian. Adapun metode pengujian yang melibatkan Teknik *Black Box testing* dan *kuesioner*.

3.15 Blackbox Testing

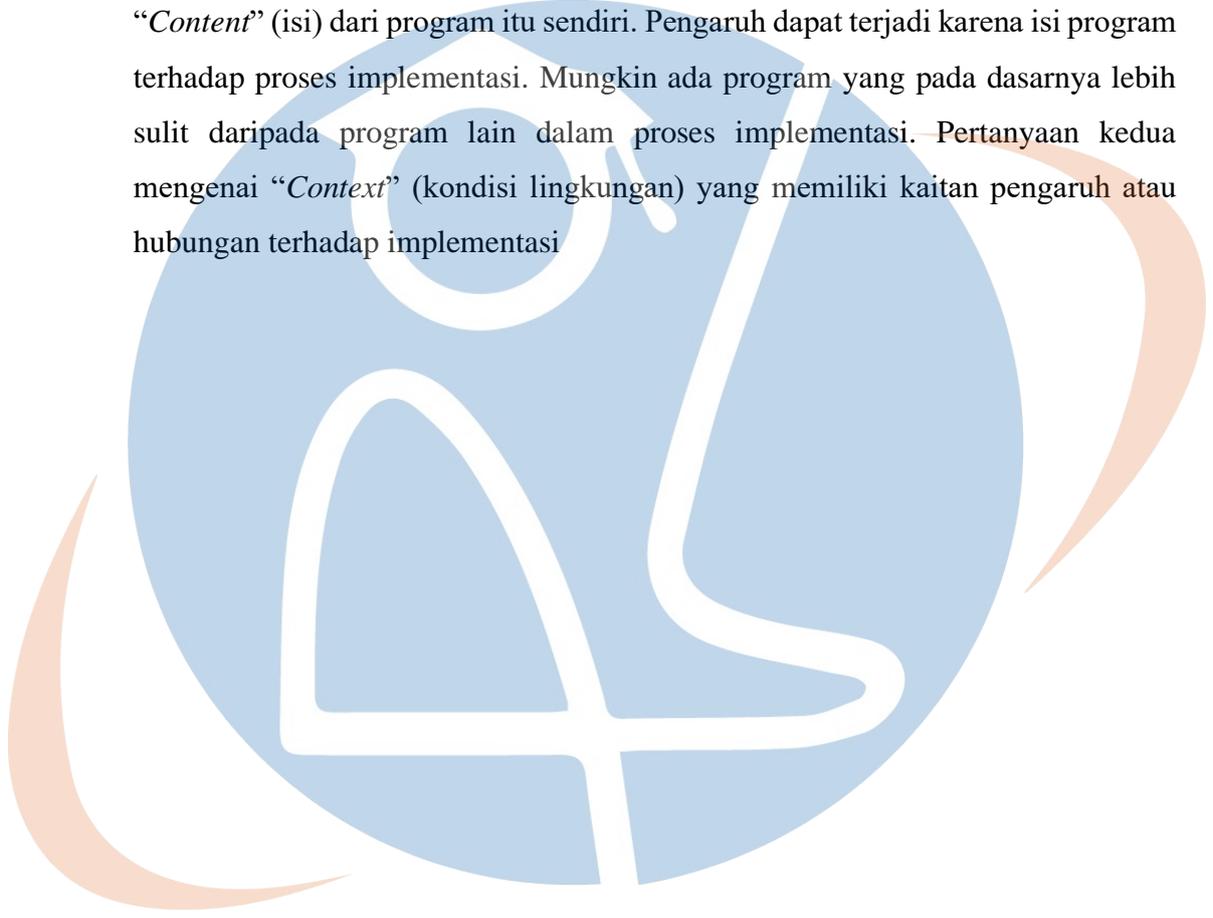
Dalam penelitian ini, pendekatan yang diterapkan untuk menguji sistem be-hafizh yang telah dirancang adalah melalui metode *Black Box Testing*. Pengujian dengan metode *Black Box* difokuskan pada aspek persyaratan fungsional dari *framework laravel*. Dengan menerapkan pendekatan *Black Box testing*, aplikasi Be-Hafizh dapat menghadapi serangkaian input yang sepenuhnya sesuai dengan persyaratan fungsional yang telah ditetapkan.

Pengujian *black box* untuk pengembangan aplikasi Be-Hafizh bertujuan untuk memastikan fungsional yang benar, kehandalan, keamanan, kemudahan pengguna, serta kompatibilitas dan interoperabilitas dengan hasil yang diharapkan agar aplikasi be-hafizh dapat beroperasi sesuai spesifikasi, memberikan respons yang konsisten, menjaga keamanan, menawarkan antarmuka yang intuitif, dan

dapat berintegrasi dengan berbagai sistem *eksternal*.

3.16 Metode Implementasi dan Evaluasi

“*Implementation*” dari *Grindle*, sebagaimana model konseptual dan kerangka pemikiran implementasi sebagai proses politik dan proses administrasi. Kerangka pemikiran tersebut disusun atas jawaban dua pertanyaan pokok, khususnya di negara berkembang pada umumnya. Pertanyaan pertama mengenai “*Content*” (isi) dari program itu sendiri. Pengaruh dapat terjadi karena isi program terhadap proses implementasi. Mungkin ada program yang pada dasarnya lebih sulit daripada program lain dalam proses implementasi. Pertanyaan kedua mengenai “*Context*” (kondisi lingkungan) yang memiliki kaitan pengaruh atau hubungan terhadap implementasi



STT - NF

BAB IV

IMPLEMENTASI DAN EVALUASI

Pada bab 4 terdapat langkah-langkah implementasi serta evaluasi kinerja dalam pengembangan aplikasi be-hafizh. Ini mencakup aspek implementasi yang mendalam guna menyempurnakan aplikasi be-hafizh. Evaluasi hasil implementasi menjadi fokus utama dalam memastikan keefektifan solusi yang dikembangkan.

4.1 Cara Kerja Website

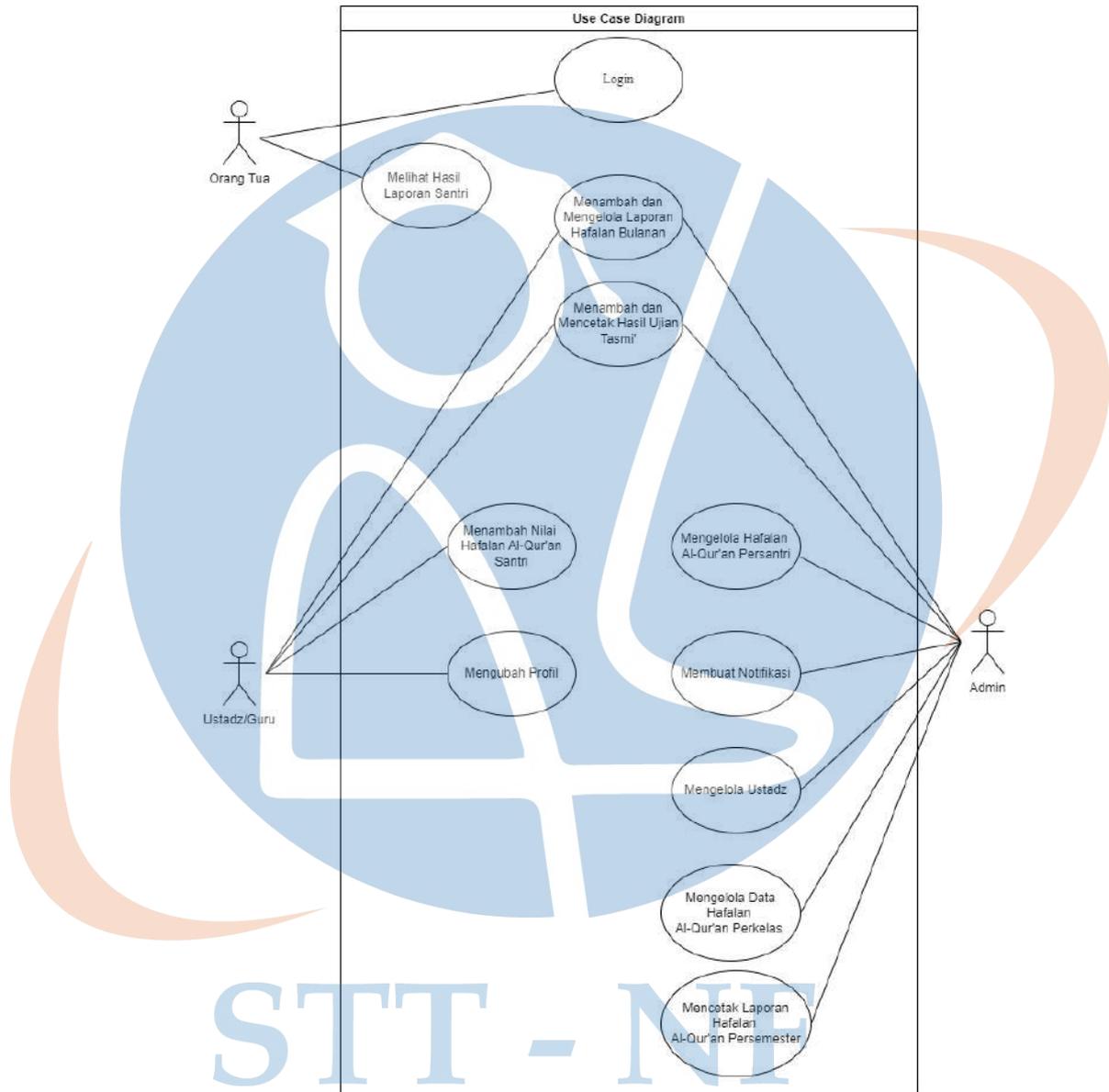


Gambar 4.1 Flowchart Admin

Flowchart Admin. Ketika Admin mengakses website dan melakukan login, setelah login sistem akan mengidentifikasi akun untuk hak akses website, jika gagal login maka admin akan kembali ke form login, Jika berhasil maka akan tampil form admin. Admin dapat melakukan pengolahan data atribut jumlah hafalan santri jumlah hafalan al qur'an santri untuk data menambah data kategori

untuk pilihan guru serta melakukan pengolahan data jumlah hafalan santri dari nama, kelas, bulan, juz, halaman, serta rekap jumlah hafalan. *Admin* memiliki akses penuh dimana dapat mengelola hak akses *admin*.

4.2 Use Case Diagram



4.2 Use Case Diagram

Di *Use Case Diagram* terdapat 3 aktor yaitu *Admin*, *Ustadz/Guru*, dan *Orang Tua*, berikut penjelasannya:

1) Orang Tua

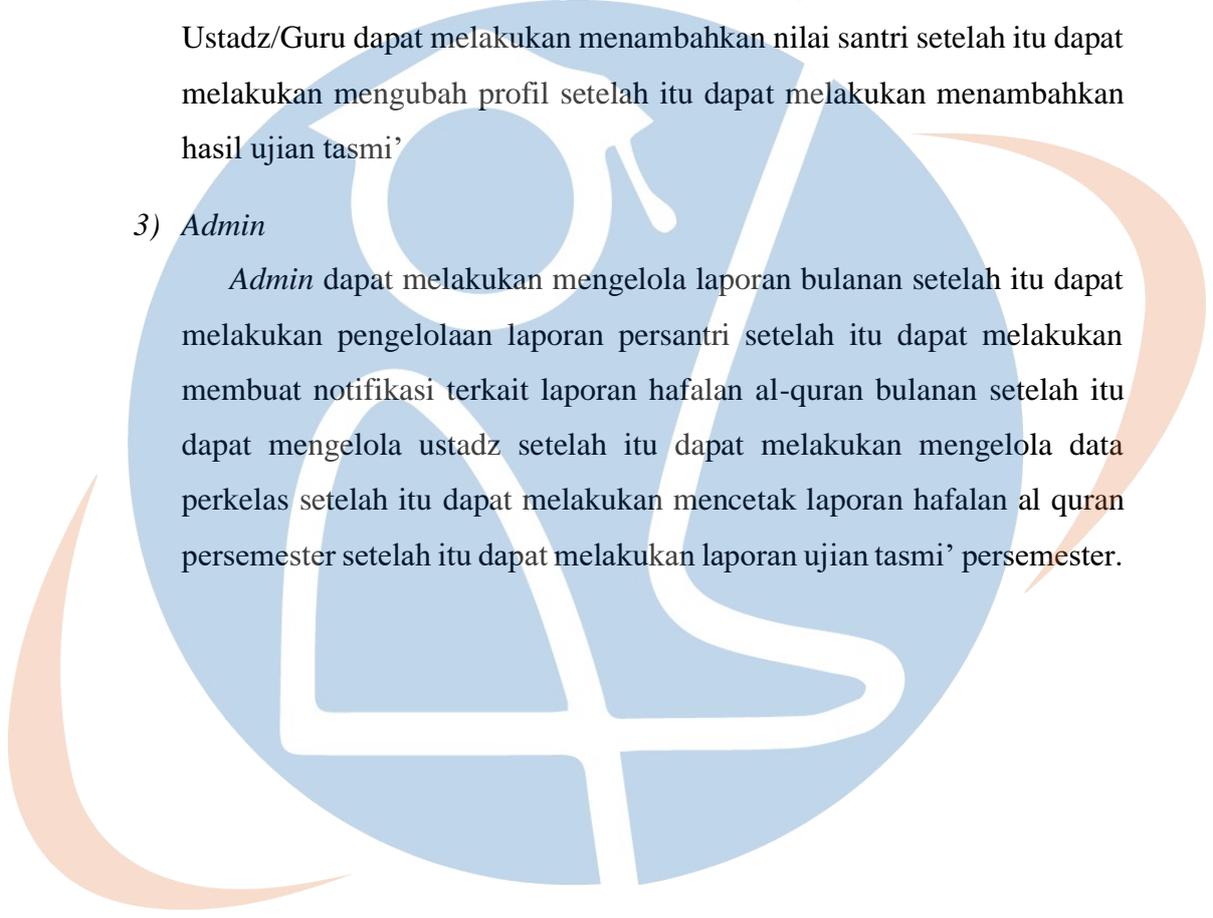
Orang Tua dapat melakukan *login* setelah itu orang tua dapat melihat hasil laporan perkembangan santri

2) Ustadz/Guru

Ustadz/Guru dapat melakukan *login* setelah itu Ustadz/Guru dapat melakukan menambahkan laporan hafalan al quran bulanan setelah itu Ustadz/Guru dapat melakukan menambahkan nilai santri setelah itu dapat melakukan mengubah profil setelah itu dapat melakukan menambahkan hasil ujian tasmi'

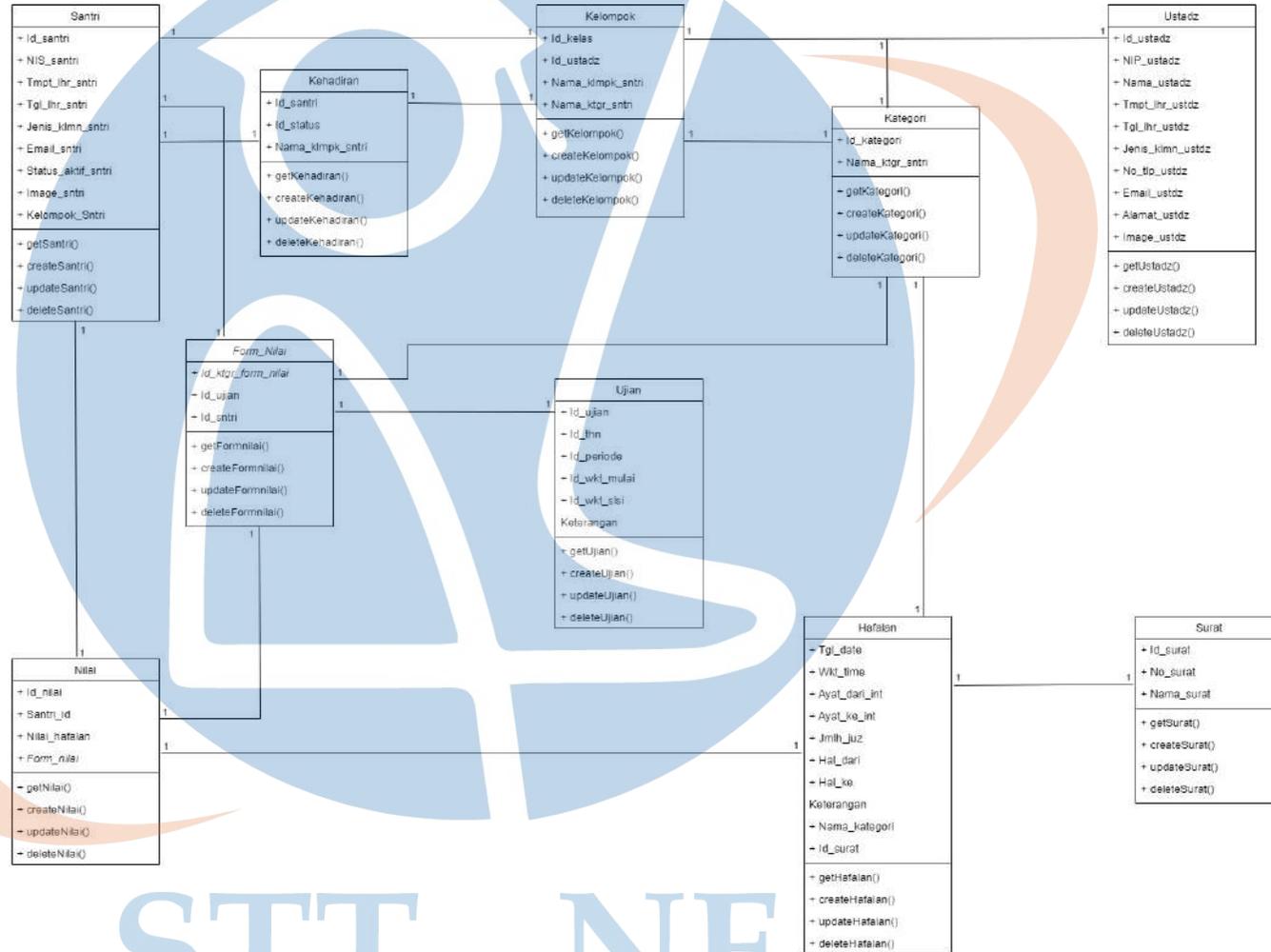
3) Admin

Admin dapat melakukan mengelola laporan bulanan setelah itu dapat melakukan pengelolaan laporan persantri setelah itu dapat melakukan membuat notifikasi terkait laporan hafalan al-quran bulanan setelah itu dapat mengelola ustadz setelah itu dapat melakukan mengelola data perkelas setelah itu dapat melakukan mencetak laporan hafalan al quran persemester setelah itu dapat melakukan laporan ujian tasmi' persemester.



STT - NF

4.3 Class Diagram



Gambar 4.3 Class Diagram

Pada gambar 4.3 *Class Diagram* , berikut penjelasannya:

1. Santri

- a. Atribut: `Id_santri`, `NIS_santri`, `Tmpt_lhr_santri`,
`Tgl_lhr_santri`, `Jenis_klmn_santri`, `Email_santri`,
`Status_aktif_santri`, `Image_santri`, `Kelompok_Sntri`
- b. Metode: `getSantri()`, `createSantri()`, `updateSantri()`,
`deleteSantri()`

2. Kehadiran

- a. Atribut: `Id_santri`, `Id_status`, `Nama_klmpk_sntri`
- b. Metode: `getKehadiran()`, `createKehadiran()`,
`updateKehadiran()`, `deleteKehadiran()`

3. Kelompok

- a. Atribut: `Id_kelas`, `Id_ustadz`, `Nama_klmpk_sntri`
- b. Metode: `getKelompok()`, `createKelompok()`,
`updateKelompok()`, `deleteKelompok()`

4. Kategori

- a. Atribut: `Id_kategori`, `Nama_klgr_sntri`
- b. Metode: `getKategori()`, `createKategori()`, `updateKategori()`,
`deleteKategori()`

5. Ustadz

- a. Atribut: `Id_ustadz`, `NIP_ustadz`, `Nama_ustadz`,
`Tmpt_lhr_ustadz`, `Tgl_lhr_ustadz`, `Jenis_klmn_ustadz`,
`No_tlp_ustadz`, `Email_ustadz`, `Alamat_ustadz`, `Image_ustadz`
- b. Metode: `getUstadz()`, `createUstadz()`, `updateUstadz()`,

deleteUstadz()

6. **Form_Nilai**

- a. Atribut: *Id_ktgr_form_nilai, Id_ujian, Id_sntri*
- b. Metode: *getFormnilai(), createFormnilai(), updateFormnilai(), deleteFormnilai()*

7. **Nilai**

- a. Atribut: *Id_nilai, Santri_id, Nilai_hafalan, Form_nilai*
- b. Metode: *getNilai(), createNilai(), updateNilai(), deleteNilai()*

8. **Ujian**

- a. Atribut: *Id_ujian, Id_thn, Id_periode, Id_wkt_mulai, Id_wkt_slsi, Keterangan*
- b. Metode: *getUjian(), createUjian(), updateUjian(), deleteUjian()*

9. **Hafalan**

- a. Atribut: *Tgl_date, Wkt_time, Ayat_dari_int, Ayat_ke_int, Jmlh_juz, Hal_dari, Hal_ke, Keterangan, Nama_kategori, Id_surat*
- b. Metode: *getHafalan(), createHafalan(), updateHafalan(), deleteHafalan()*

10. **Surat**

- a. Atribut: *Id_surat, No_surat, Nama_surat*
- b. Metode: *getSurat(), createSurat(), updateSurat(), deleteSurat()*

Relasi Antar Entitas

1. *One-to-One (1:1)*

Santri ke **Kehadiran**: Satu Santri memiliki satu Kehadiran.

Kelompok ke **Kategori**: Satu Kelompok memiliki satu Kategori.

Form_Nilai ke **Ujian**: Satu *Form_Nilai* terhubung dengan satu Ujian.

2. *One-to-Many (1)*

Santri ke **Kelompok**: Satu Santri dapat tergabung dalam satu Kelompok, tetapi satu Kelompok bisa memiliki banyak Santri.

Santri ke **Form_Nilai**: Satu Santri bisa memiliki banyak *Form_Nilai*.

Ustadz ke **Kelompok**: Satu Ustadz bisa memimpin banyak Kelompok.

Kelompok ke **Santri**: Satu Kelompok bisa memiliki banyak Santri.

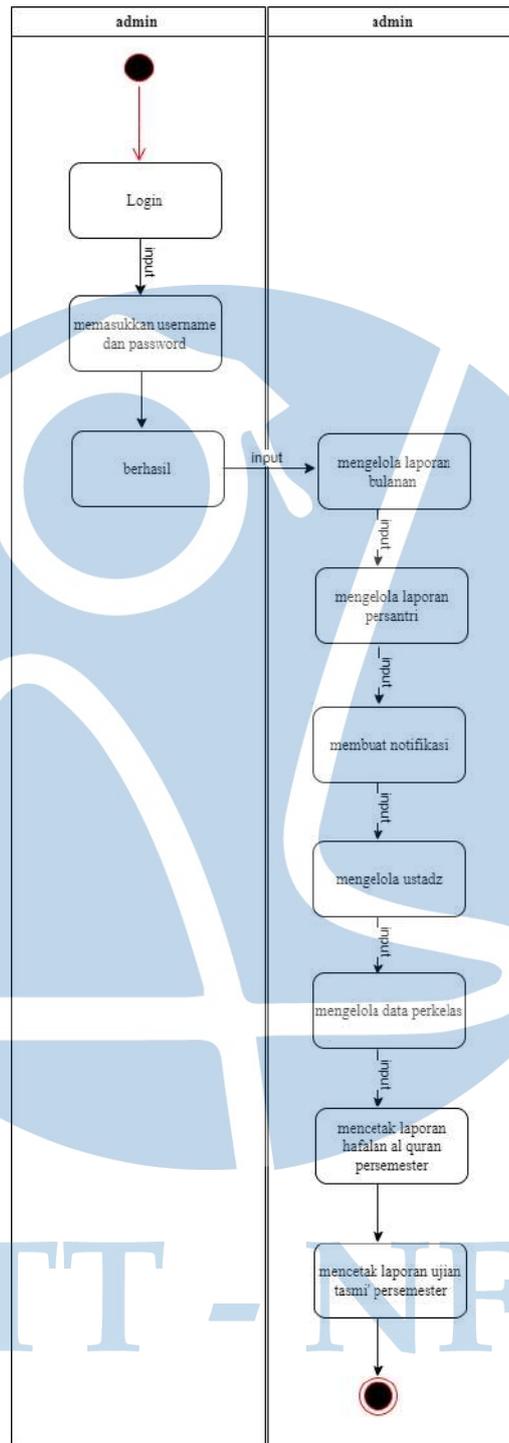
Form_Nilai ke **Nilai**: Satu *Form_Nilai* bisa memiliki banyak Nilai.

Hafalan ke **Surat**: Satu Hafalan terhubung dengan satu Surat, tetapi satu Surat bisa terhubung dengan banyak Hafalan.

Diagram ini menunjukkan hubungan yang kompleks antara entitas yang ada di dalam sebuah sistem pengelolaan santri dan hafalan. Relasi *one-to-one* menggambarkan hubungan langsung antara dua entitas yang saling eksklusif, sementara *one-to-many* menggambarkan hubungan dimana satu entitas bisa berhubungan dengan banyak entitas lainnya.

STT - NF

4.4 Activity Diagram Admin

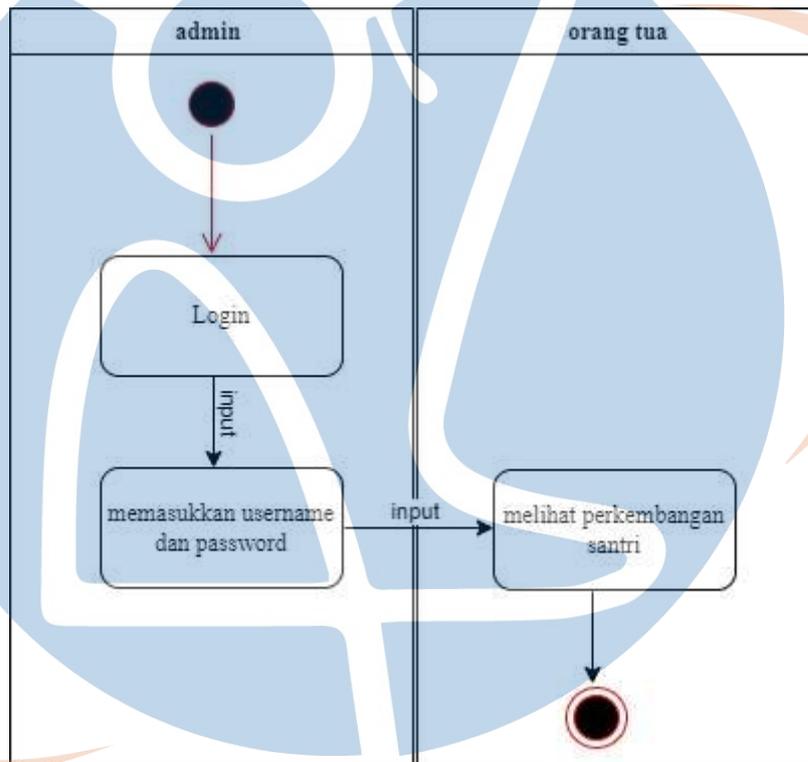


Gambar 4.4 Activity Diagram Admin

Pada gambar 4.4 terdapat actor yaitu *admin*, *admin* dapat melakukan login dengan menggunakan *username* dan *password* yang sudah terdaftar

ketika sudah berhasil maka *admin* akan melakukan pengelolaan laporan bulanan kemudian *admin* mengelola laporan hafalan al-quran persantri kemudian *admin* melakukan membuat notifikasi kepada ustadz/guru kemudian *admin* melakukan pengelolaan terhadap ustadz/guru kemudian *admin* melakukan mengelola data hafalan al-qur'an perkelas kemudian *admin* melakukan pencetakan laporan hafalan al-qur'an persemester kemudian *admin* melakukan pencetakan laporan ujian tasmi' persemester.

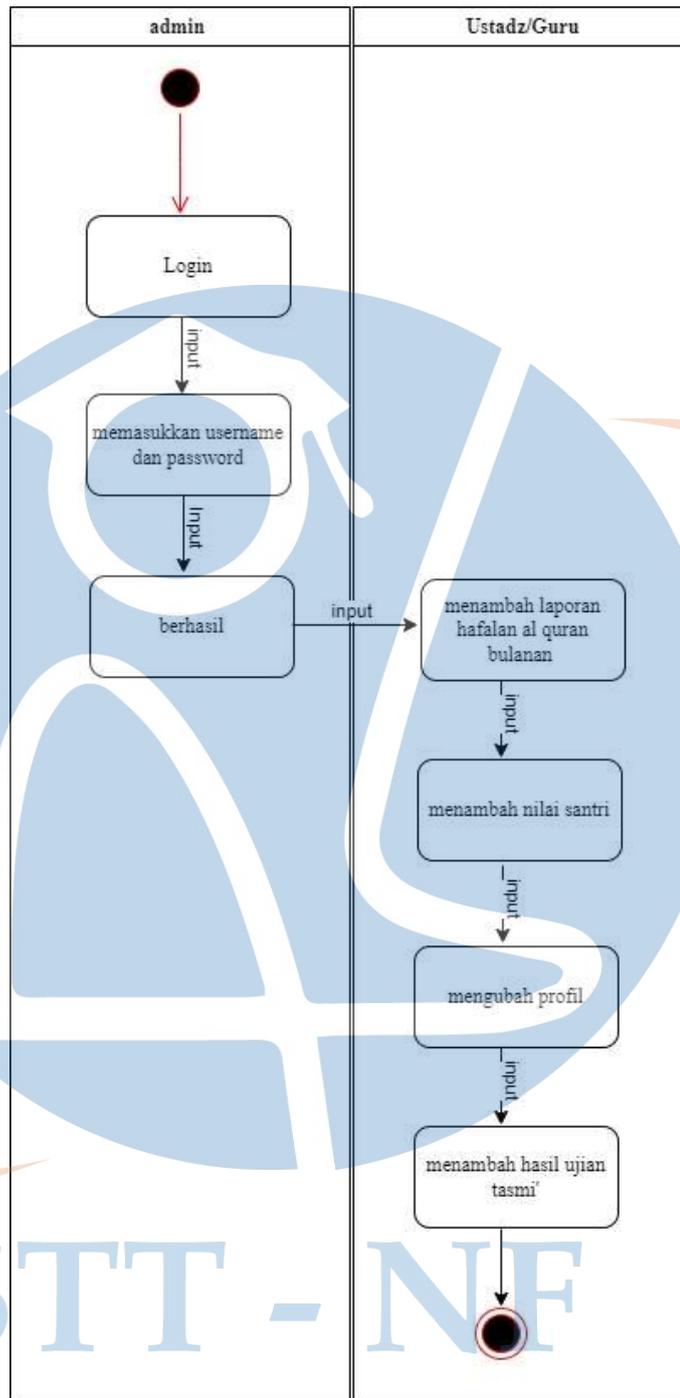
4.5 Activity Diagram Orang Tua



Gambar 4.5 Activity Diagram Orang Tua

Pada gambar 4.5 terdapat actor yaitu orang tua, orang tua dapat melakukan login dengan *username* dan *password* yang sudah terdaftar kemudian orang tua mengklik tombol melihat perkembangan santri.

4.6 Activity Diagram Ustadz/Guru



Gambar 4.6 Activity Diagram Ustadz/Guru

Pada gambar 4.6 terdapat *actor* yaitu ustadz/guru, ustadz/guru dapat melakukan *login* dengan menggunakan *username* dan *password* yang sudah terdaftar kemudian ustadz/guru dapat melakukan menambahkan laporan hafalan

al-qur'an bulanan kemudian ustadz/guru dapat melakukan menambahkan nilai santri kemudian ustadz/guru dapat melakukan mengubah profilnya kemudian ustadz/guru dapat melakukan menambahkan hasil ujian tasmi'.

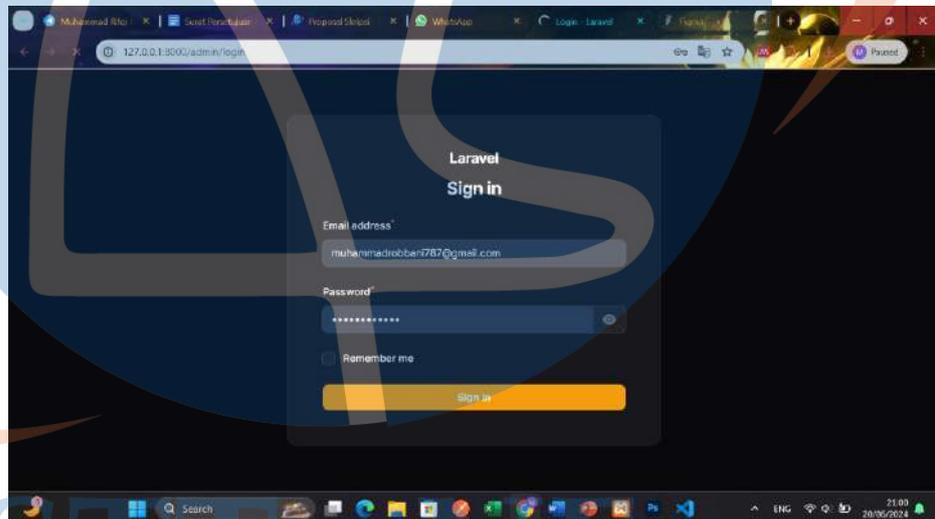
4.7 Implementasi

Tahap implementasi adalah tahap yang menerjemahkan perancangan berdasarkan pada hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangkat lunak pada keadaan yang sesungguhnya.

4.5.1 Implementasi Antarmuka User

Implementasi antarmuka (*interface*) dari perangkat lunak dilakukan berdasarkan rancangan yang telah dilakukan. Implementasi ditampilkan dari *screenshot* dari halaman website yang digunakan sebagai alat dan bahan penelitian.

1. Halaman *Login*



Gambar 4.5.1 Halaman *Login Laravel*

Halaman ini terdiri dari *username* dan *password* untuk melakukan login sebelum masuk ke halaman utama sesuai hak akses masing – masing. Apabila *username* dan *password* tidak sesuai maka tidak bisa *login*

2. Halaman beranda *Laravel*



Gambar 4.5.2 Halaman Beranda *Laravel*

Tampilan pertama ketika website dibuka, berisikan visi dan misi Pondok Pesantren Takhasus Tahfidz Al Ma'shum Mardiyah.

3. Halaman Monitoring Hafalan Al Qur'an



Gambar 4.5.3 Halaman Monitoring Hafalan Al Qur'an

Halaman ini menjelaskan bagian monitoring hafalan Al Qur'an santri berbasis *web* menggunakan *framework laravel*.

4. Halaman Profil Pondok Pesantren Takhassus Tahfidz Al – Ma’shum Mardiyah



Gambar 4.5.4 Halaman Profil Pondok Pesantren Al – Ma’shum Mardiyah

Halaman ini menjelaskan sekilas tentang profil Pondok Pesantren Takhassus Tahfidz Al – Ma’shum Mardiyah termasuk sejarah singkat dan anggota kepala unit Al-Ma’shum Mardiyah.

STT - NF

5. Halaman Informasi Menghubungi Pondok Pesantren Takhassus Tahfidz Al – Ma’shum Mardiyah



Gambar 4.5.5 Halaman Informasi Kontak Pondok Pesantren Al – Ma’shum Mardiyah

Halaman ini berisi kontak komunitas santri untuk menghubungi pihak Pondok Pesantren Takhassus Tahfidz Al – Ma’shum Mardiyah.

4.5.2 Implementasi Sistem Informasi Antar Muka Admin

1. Pengujian halaman Daftar

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------|---|---|---|
| Normal | Pengguna menekan tombol daftar pada halaman <i>login</i> . lalu pengguna mengisi semua formulir data yang ada pada halaman registrasi | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>database</i> |

| | | | |
|--------------|--|---|---|
| <i>Error</i> | Pengguna tidak mengisi formulir yang ada | Sistem akan memberikan <i>error</i> di bawah formulir yang kosong | Berhasil sistem menampilkan pesan di bawah formulir yang kosong |
|--------------|--|---|---|

Tabel 4.5.2.1 Pengujian Halaman Daftar

2. Pengujian halaman *login*

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------------|--|---|---|
| Normal | Pengguna menekan tombol <i>login</i> pada halaman <i>login</i> . lalu pengguna mengisi semua <i>email</i> dan <i>password</i> yang ada pada halaman <i>login</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>database</i> |
| <i>Error</i> | Pengguna tidak mengisi email atau salah <i>email</i> yang sudah di daftarkan | Sistem akan memberikan <i>error</i> di halaman <i>login</i> yang tersedia | Berhasil sistem menampilkan pesan di bawah halaman <i>login</i> yang tidak atau salah memasukkan <i>email</i> |

Tabel 4.5.2.2 Pengujian Halaman *Login*

3. Pengujian Admin Ustadz

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------|---|----------------------------------|---|
| Normal | <i>Admin</i> menekan tombol <i>new guru</i> pada halaman <i>gurus</i> | Sistem dapat menyimpan data yang | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|-------------|---|---|--|
| | <p>lalu <i>admin</i> akan memasukkan nip, nama guru, tmpt lhr guru, tgl lhr guru, jenis kelamin, no tlp, <i>email</i>, alamat, <i>image</i> lalu tekan tombol <i>create</i></p> | <p>telah dimasukan ke <i>database</i></p> | |
| <i>Eror</i> | <p><i>Admin</i> menekan tombol <i>new</i> guru pada halaman gurus lalu <i>admin</i> tidak memasukkan nip, nama guru, tmpt lhr guru, tgl lhr guru, jenis kelamin, no tlp, <i>email</i>, alamat, <i>image</i> lalu tekan tombol <i>create</i></p> | <p>Sistem akan memberikan <i>error</i> di halaman gurus yang tersedia</p> | <p>Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i></p> |

Tabel 4.5.2.3 Pengujian Halaman Gurus

4. Pengujian *Form* Nilais

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------------|--|---|---|
| Normal | <i>Admin</i> menekan tombol <i>form</i> nilais pada halaman <i>form</i> nilais lalu memasukkan nilai lalu tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |
| <i>Error</i> | <i>Admin</i> menekan tombol <i>form</i> nilais pada halaman <i>form</i> nilais lalu tidak memasukkan nilai lalu tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman <i>form</i> nilais yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |

Tabel 4.5.2.4 Pengujian Halaman *Form* Nilais

5. Pengujian Halaman Hafalans

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------|---|----------------------------------|---|
| Normal | <i>Admin</i> menekan tombol hafalans pada | Sistem dapat menyimpan data yang | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|-------------|--|--|--|
| | <p>halaman hafalans lalu memasukkan tgl, waktu, ayat dari, ayat ke, jmlh juz, hal dari, hal ke, keterangan, kategori, surat, santri kemudian tekan tombol <i>create</i></p> | <p>telah dimasukan ke <i>database</i></p> | |
| <p>Eror</p> | <p><i>Admin</i> menekan tombol hafalans pada halaman hafalans lalu tidak memasukkan tgl, waktu, ayat dari, ayat ke, jmlh juz, hal dari, hal ke, keterangan, kategori, surat, santri kemudian</p> | <p>Sistem akan memberikan <i>error</i> di halaman hafalans yang tersedia</p> | <p>Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i></p> |

| | | | |
|--|-------------------------------|--|--|
| | tekan tombol <i>create</i> | | |
|--|-------------------------------|--|--|

Tabel 4.5.2.5 Pengujian Halaman Hafalans

6. Pengujian Halaman Kategoris

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------------|--|--|---|
| Normal | <i>Admin</i> menekan tombol kategoris pada halaman kategoris lalu tekan tombol <i>new</i> kategoris lalu memasukkan nama kategoris kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |
| <i>Error</i> | <i>Admin</i> menekan tombol kategoris pada halaman kategoris lalu tidak memasukkan nama kategoris kemudian | Sistem akan memberikan <i>error</i> di halaman kategoris yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|--|-------------------------------|--|--|
| | tekan tombol <i>create</i> | | |
|--|-------------------------------|--|--|

Tabel 4.5.2.6 Pengujian Halaman Kategoris

7. Pengujian Halaman Kehadirans

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------------|--|---|---|
| Normal | <i>Admin</i> menekan tombol kehadirans pada halaman kehadirans lalu tekan tombol <i>new</i> kehadirans lalu memasukkan tgl, status, santri, kelompok kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |
| <i>Error</i> | <i>Admin</i> menekan tombol kehadirans pada halaman kehadirans lalu tekan tombol <i>new</i> | Sistem akan memberikan <i>error</i> di halaman kehadirans yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|--|--|--|--|
| | kehadirans lalu tidak memasukkan tgl, status, santri, kelompok kemudian tekan tombol <i>create</i> | | |
|--|--|--|--|

Tabel 4.5.2.7 Pengujian Halaman Kehadirans

8. Pengujian Halaman Kelompoks

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------|--|---|---|
| Normal | <i>Admin</i> menekan tombol kelompoks pada halaman kelompoks lalu tekan tombol <i>new</i> kelompoks lalu memasukkan kategori santri, kelas, guru, kategori kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|------|---|---|---|
| Eror | <i>Admin</i> menekan tombol kelompok pada halaman kelompok lalu tekan tombol <i>new</i> kelompok lalu tidak memasukkan kategori santri, kelas, guru, kategori kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman kelompok yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |
|------|---|---|---|

Tabel 4.5.2.8 Pengujian Halaman Kelompok

9. Pengujian Halaman Nilais

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------|---|---|---|
| Normal | <i>Admin</i> menekan tombol nilai pada halaman nilai lalu tekan tombol <i>new</i> nilai lalu memasukkan santri, <i>score</i> , hafalan, <i>form</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|------|---|--|---|
| | nilai kemudian tekan tombol <i>create</i> | | |
| Eror | <i>Admin</i> menekan tombol nilai pada halaman nilai lalu tekan tombol <i>new</i> nilai lalu tidak memasukkan santri, <i>score</i> , hafalan, <i>form</i> nilai kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman <i>form</i> nilai yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |

Tabel 4.5.2.9 Pengujian Halaman Nilais

10. Pengujian Halaman Santris

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------|---|---|---|
| Normal | <i>Admin</i> menekan tombol santris pada halaman santris lalu tekan tombol <i>new</i> santris lalu memasukkan | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|------|--|--|---|
| | santri, <i>score</i> , hafalan, kelompok kemudian tekan tombol <i>create</i> | | |
| Eror | <i>Admin</i> menekan tombol santris pada halaman santris lalu tekan tombol <i>new</i> santris lalu tidak memasukkan santri, <i>score</i> , hafalan, kelompok kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman santris yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |

Tabel 4.5.2.9.1 Pengujian Halaman Santris

11. Pengujian Halaman Surats

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------|---|---|---|
| Normal | <i>Admin</i> menekan tombol surats pada halaman surats lalu | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|------|---|--|--|
| | tekan tombol <i>new</i> surats lalu memasukkan no surat, nama surat kemudian tekan tombol <i>create</i> | | |
| Eror | <i>Admin</i> menekan tombol surats pada halaman surats lalu tekan tombol <i>new</i> surats lalu tidak memasukkan no surat, nama surat kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman surats yang tersedia | Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i> |

Tabel 4.5.2.9.2 Pengujian Halaman Surats

12. Pengujian Halaman Ujians

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------|---|---|--|
| Normal | <i>Admin</i> menekan tombol ujians pada halaman ujians lalu tekan tombol | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>Database</i> |

| | | | |
|------|--|--|--|
| | <p><i>new</i> ujian lalu memasukkan thn, periode, waktu mulai, waktu sksi, keterangan kemudian tekan tombol <i>create</i></p> | | |
| Eror | <p><i>Admin</i> menekan tombol ujians pada halaman ujians lalu tekan tombol <i>new</i> ujian lalu tidak memasukkan thn, periode, waktu mulai, waktu sksi, keterangan kemudian tekan tombol <i>create</i></p> | <p>Sistem akan memberikan <i>error</i> di halaman ujians yang tersedia</p> | <p>Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i></p> |

Tabel 4.5.2.9.3 Pengujian Halaman Ujians

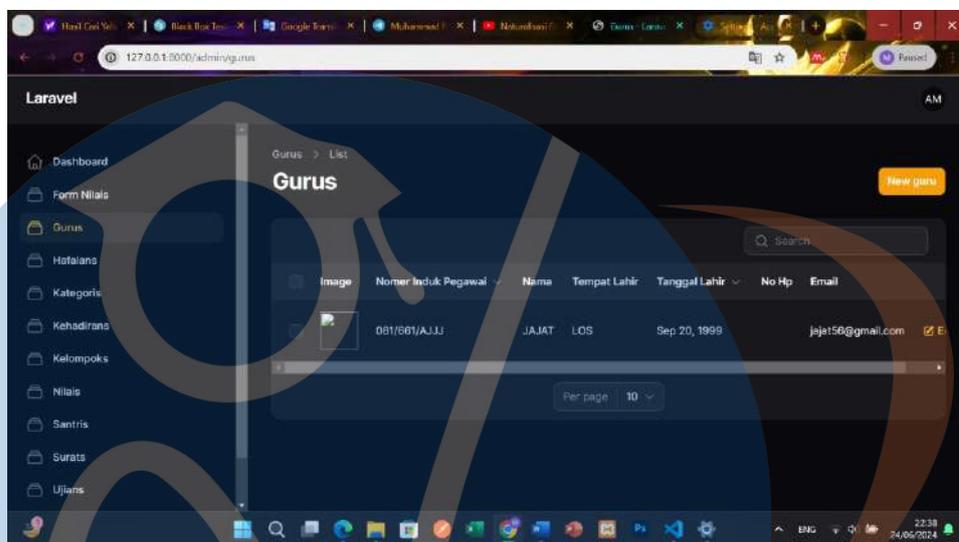
13. Pengujian Halaman *Users*

| Tipe | Skenario Pengujian | <i>Output</i> yang diharapkan | <i>Output</i> |
|--------|---|--|--|
| Normal | <p><i>Admin</i> menekan tombol <i>users</i> pada halaman <i>users</i> lalu tekan tombol <i>new users</i> lalu memasukkan nama, <i>email</i>, <i>password</i>, profil <i>photo path</i>, <i>role</i> kemudian tekan tombol <i>create</i></p> | <p>Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i></p> | <p>Berhasil data pengguna tersimpan ke dalam <i>Database</i></p> |
| Eror | <p><i>Admin</i> menekan tombol <i>users</i> pada halaman <i>users</i> lalu tekan tombol <i>new users</i> lalu memasukkan nama, <i>email</i>, <i>password</i>, profil <i>photo path</i>, <i>role</i></p> | <p>Sistem akan memberikan <i>error</i> di halaman <i>users</i> yang tersedia</p> | <p>Tidak berhasil data pengguna tersimpan ke dalam <i>Database</i></p> |

| | | | |
|--|-------------------------------------|--|--|
| | kemudian tekan tombol <i>create</i> | | |
|--|-------------------------------------|--|--|

Tabel 4.5.2.9.4 Pengujian Halaman Users

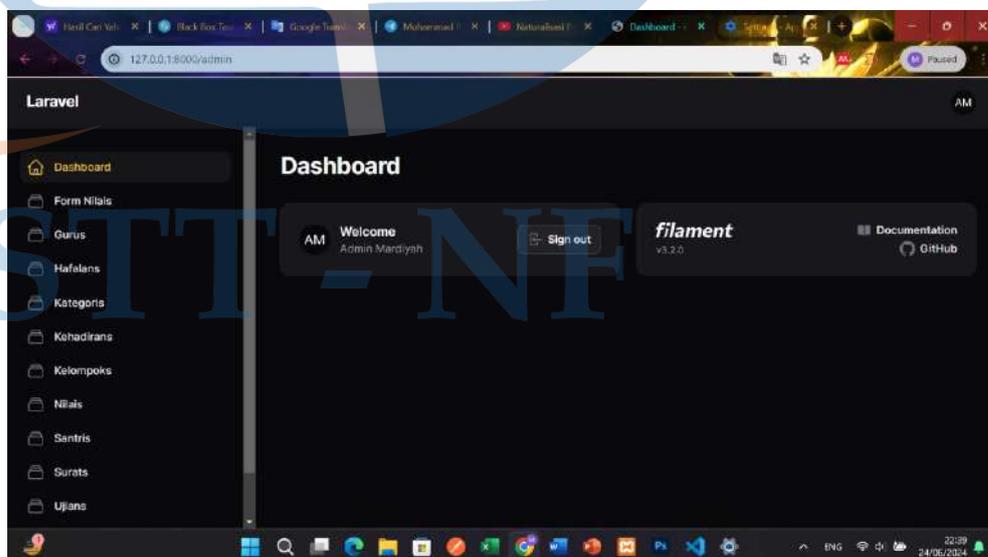
14. Halaman *Admin Ustadz*



Gambar 4.5.6 Halaman *Admin Ustadz*

Pada Halaman ini menjelaskan tentang menambahkan ustadz atau guru yang dilakukan ketika ada ustadz atau guru yang baru masuk di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

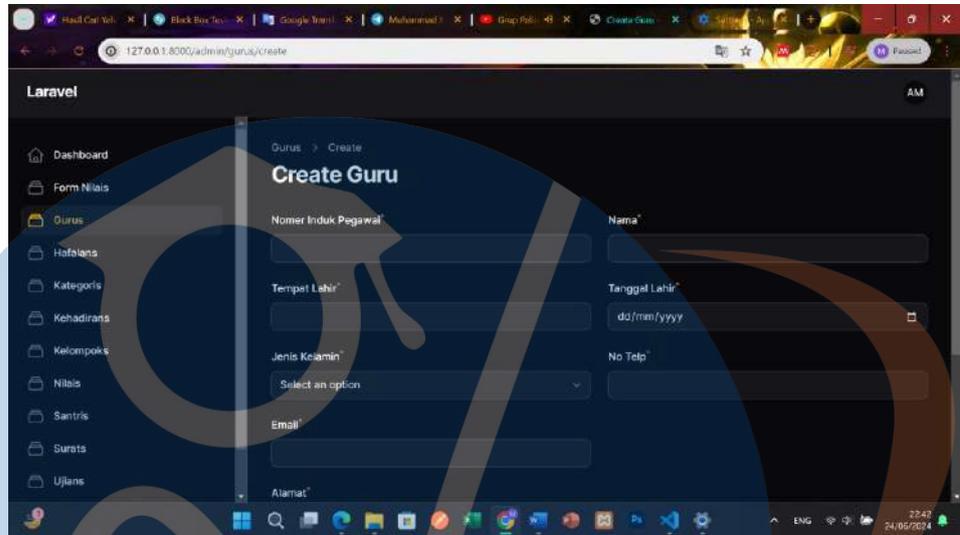
15. Halaman *Dashboard Admin*



Gambar 4.5.7 Halaman *Dashboard Admin*

Pada Halaman ini menjelaskan tentang Halaman *Dashboard Admin* yang akan melakukan penambahan guru, nilai, kategori, juz, *form* nilai, kehadiran, hafalan, kehadiran. Santri, *user*.

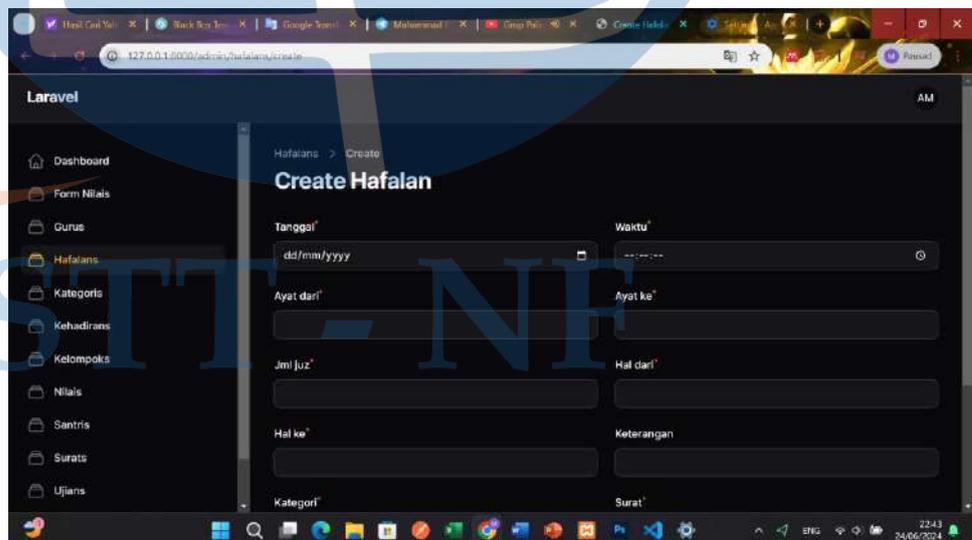
16. Halaman Menambahkan Ustadz atau Guru



Gambar 4.5.8 Halaman Menambahkan Ustadz atau Guru

Pada Halaman ini akan menambahkan Ustadz atau Guru yang dilakukan oleh *admin* ketika ada penambahan ustadz atau guru di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

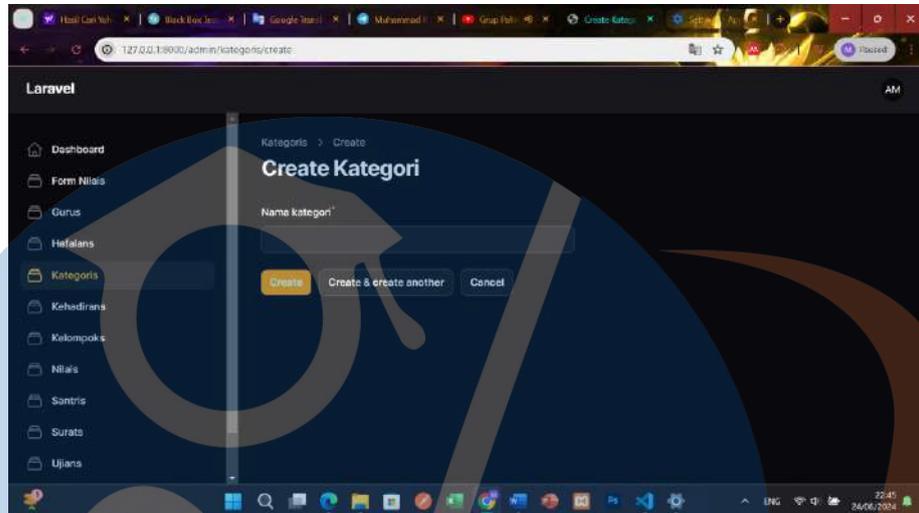
17. Halaman Menambahkan Hafalan Al-Quran



Gambar 4.5.9 Halaman Menambahkan Hafalan Al-Quran

Pada Halaman ini akan menambahkan Hafalan Al-Quran yang dilakukan oleh *admin* ketika ada penambahan hafalan Al-Quran di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

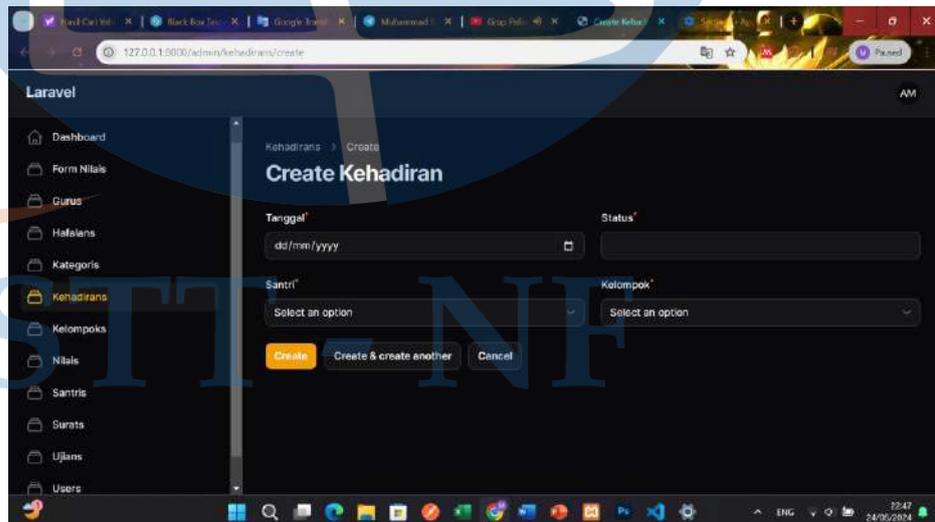
18. Halaman Menambahkan Kategori



Gambar 4.6.0 Halaman Menambahkan Kategori

Pada Halaman ini akan menambahkan kategori penilaian yang dilakukan oleh *admin* ketika ada penambahan nilai di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

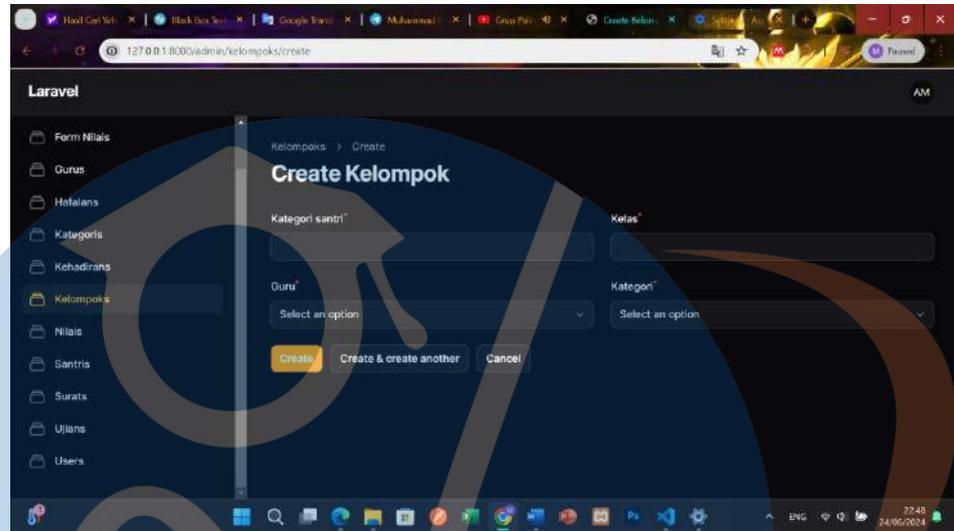
19. Halaman Menambahkan Kehadiran Santri



Gambar 4.6.1 Halaman Menambahkan Kehadiran Santri

Pada Halaman ini akan menambahkan kehadiran santri yang dilakukan oleh *admin* ketika ada penambahan kehadiran santri di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah.

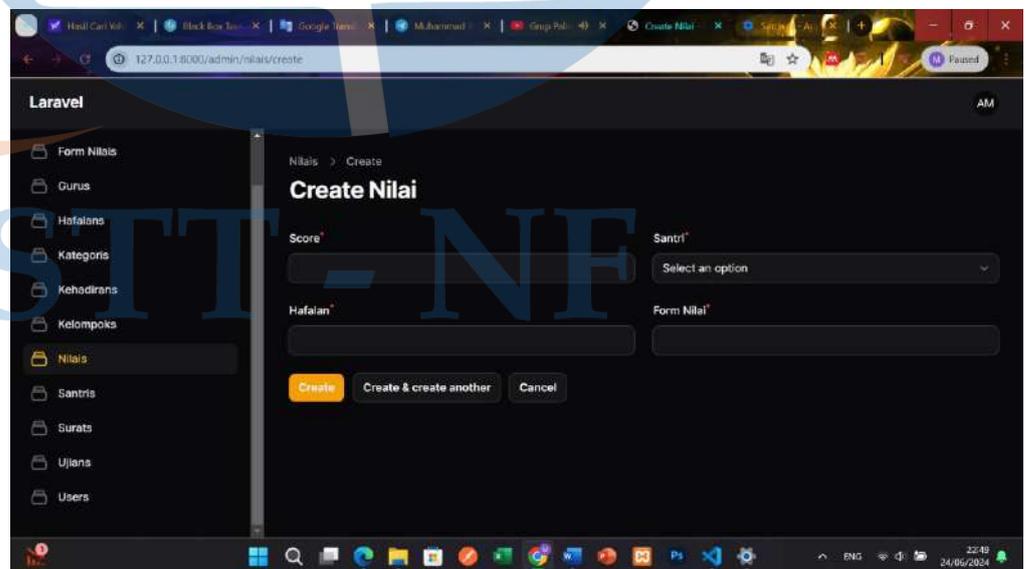
20. Halaman Menambahkan Kelompok Setoran Hafalan Al-Quran



Gambar 4.6.2 Halaman Menambahkan Kelompok Setoran Hafalan Al-Quran

Pada Halaman ini akan menambahkan kelompok setoran hafalan Al-Quran yang dilakukan oleh *admin* ketika ada penambahan kelompok baru di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

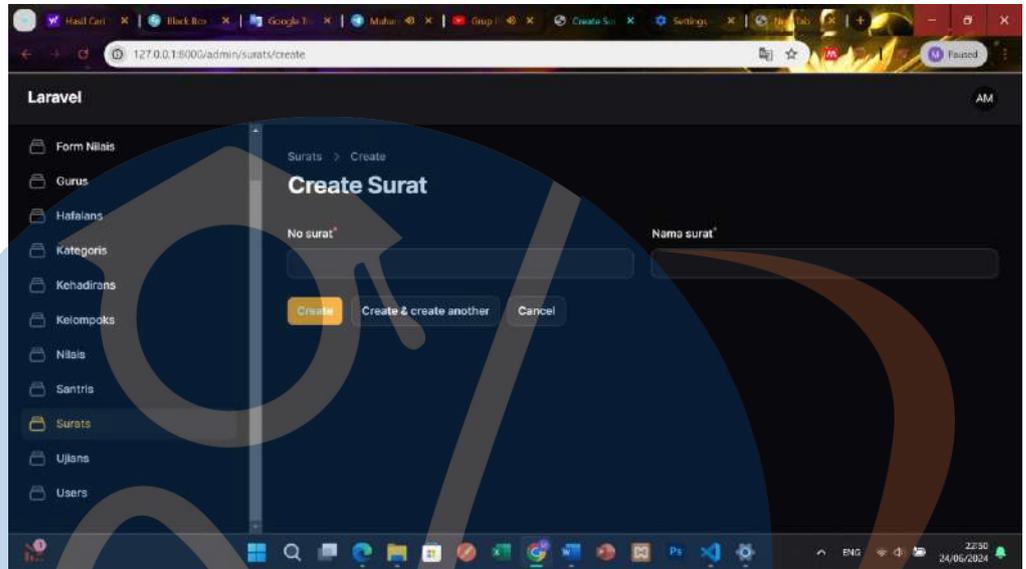
21. Halaman Menambahkan Nilai



Gambar 4.6.3 Halaman Menambahkan Nilai

Pada Halaman ini akan menambahkan nilai yang dilakukan oleh *admin* ketika ada penambahan nilai di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah.

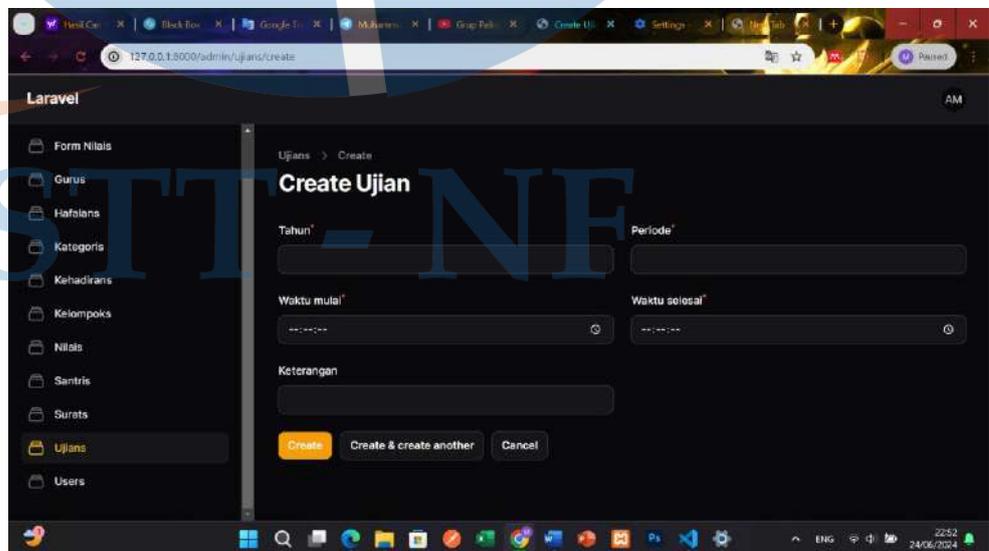
22. Halaman Menambahkan Surat Hafalan Al-Qur'an



Gambar 4.6.4 Halaman Menambahkan Surat Hafalan Al-Quran

Pada Halaman ini akan menambahkan surat dalam Al-Qur'an yang dilakukan oleh *admin* ketika ada penambahan surat Al-Qur'an yang baru di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

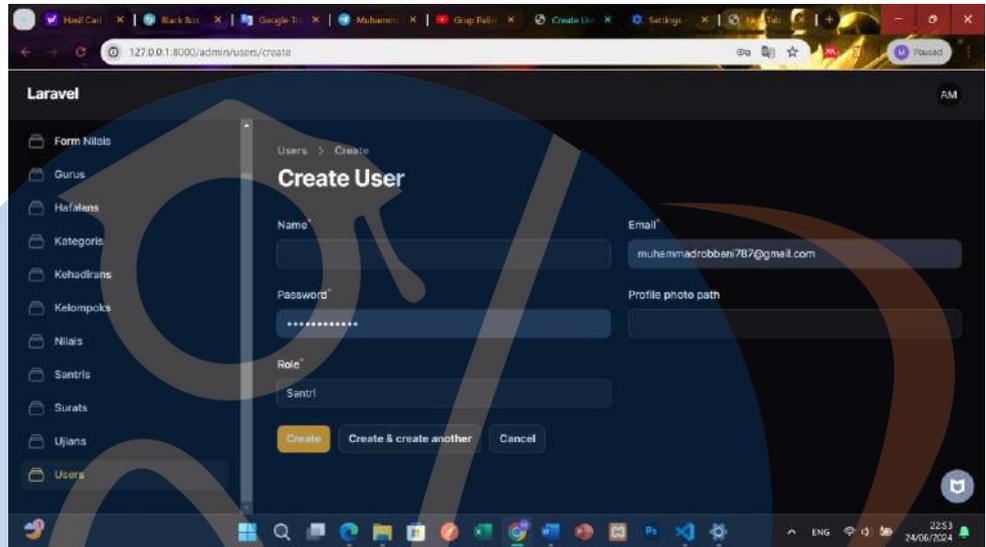
23. Halaman Menambahkan Ujian



Gambar 4.6.5 Halaman Menambahkan Ujian

Pada Halaman ini akan menambahkan ujian yang dilakukan oleh *admin* ketika ada penambahan ujian yang baru di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah

24. Halaman Menambahkan *User*



Gambar 4.6.6 Halaman Menambahkan *User*

Pada Halaman ini akan menambahkan *user* yang dilakukan oleh *admin* ketika ada penambahan *user* yang baru di Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah.

25. Perkembangan Santri

| Tipe | Skenario Pengujian | Output yang diharapkan | Output |
|--------|---|---|---|
| Normal | Pengguna menekan tombol <i>login</i> pada halaman <i>login</i> . lalu pengguna mengisi semua <i>email</i> dan <i>password</i> yang ada pada halaman | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Berhasil data pengguna tersimpan ke dalam <i>database</i> |

| | | | |
|--------------|---|--|---|
| | <i>login</i> kemudian tekan tombol lihat perkembangan santri | | |
| <i>Error</i> | Pengguna tidak mengisi <i>email</i> atau salah <i>email</i> yang sudah di daftarkan | Sistem akan memberikan error di halaman <i>login</i> yang tersedia | Berhasil sistem menampilkan pesan di bawah halaman <i>login</i> yang tidak atau salah memasukkan <i>email</i> |

Tabel 4.5.2.9.5 perkembangan santri

4.5.3 Pengujian Blackbox Testing

Berikut merupakan skenario implementasi pengujian *black box testing* untuk pengembangan aplikasi be-hafizh. Berikut tahapan-tahapan pengujian *black box testing* dalam bentuk tabel:

Table 4.5.3.1 Pengujian *Black Box Testing* – *Login Admin*

| Skenario Uji | | <i>Login admin</i> | |
|--------------|---|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/login | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1 | Admin melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang sesuai | Akan menghasilkan <i>response</i> ke <i>dashboard admin</i> | Sesuai (100%) |
| 2 | Admin melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang tidak sesuai | Akan menghasilkan <i>response error</i> ketika mau <i>login</i> | Sesuai (100%) |

Pada tabel 4.5.3.1 Pengujian *Black Box Testing* – *Login Admin*, ini digunakan sebagai pengujian mulai dari admin melakukan *login* dengan menggunakan *username* dan *password* yang sesuai sampai *admin login* dengan menggunakan *username* dan *password* yang tidak sesuai.

Table 4.5.3.2 Pengujian *Black Box Testing* – ustadz atau guru

| Skenario Uji | | Ustadz atau Guru | |
|--------------|--|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/gurus | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1 | <p><i>Admin</i> menekan tombol <i>new</i> guru pada halaman gurus lalu <i>admin</i> akan memasukkan nip, nama guru, tmpt lhr guru, tgl lhr guru, jenis kelamin, no tlp, <i>email</i>, alamat, <i>image</i> lalu tekan tombol <i>create</i></p> | <p>Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i></p> | Sesuai (100%) |
| 2 | <p><i>Admin</i> menekan tombol <i>new</i> guru pada halaman gurus lalu <i>admin</i> tidak memasukkan nip, nama</p> | <p>Sistem akan memberikan <i>error</i> di halaman gurus yang tersedia</p> | Sesuai (100%) |

| | | | |
|--|---|--|--|
| | guru, tmpt lhr guru, tgl lhr guru, jenis kelamin, no tlp, <i>email</i> , alamat, <i>image</i> lalu tekan tombol <i>create</i> | | |
|--|---|--|--|

Pada tabel 4.5.3.2 Pengujian *Black Box Testing* – ustadz atau guru, ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol *new* guru pada halaman *gurus* lalu *admin* akan memasukkan nip, nama guru, tmpt lhr guru, tgl lhr guru, jenis kelamin, no tlp, *email*, alamat, *image* lalu tekan tombol *create* setelah itu *Admin* menekan tombol *new* guru pada halaman *gurus* lalu *admin* tidak memasukkan nip, nama guru, tmpt lhr guru, tgl lhr guru, jenis kelamin, no tlp, *email*, alamat, *image* lalu tekan tombol *create*.

Table 4.5.3.3 Pengujian *Black Box Testing* – *Form* Nilais

| Skenario Uji | | Form Nilais | |
|--------------|--|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/form-nilais | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol <i>form</i> nilais pada halaman <i>form</i> nilais lalu memasukkan nilai lalu tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |

| | | | |
|----|--|---|---------------|
| 2. | <i>Admin</i> menekan tombol <i>form</i> nilais pada halaman <i>form</i> nilais lalu tidak memasukkan nilai lalu tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman <i>form</i> nilais yang tersedia | Sesuai (100%) |
|----|--|---|---------------|

Pada tabel 4.5.3.3 Pengujian *Black Box Testing* – *form* nilais, ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol *form* nilais pada halaman *form* nilais lalu memasukkan nilai lalu tekan tombol *create* setelah itu *Admin* menekan tombol *form* nilais pada halaman *form* nilais lalu tidak memasukkan nilai lalu tekan tombol *create*.

Pada tabel 4.5.3.4 Pengujian *Black Box Testing* – Hafalans

| Skenario Uji | | Hafalans | |
|--------------|--|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/hafalans | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol hafalans pada halaman hafalans lalu memasukkan tgl, waktu, ayat dari, ayat ke, jmlh juz, hal dari, hal ke, | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |

| | | | |
|----|---|--|---------------|
| | keterangan, kategori, surat, santri kemudian tekan tombol <i>create</i> | | |
| 2. | <i>Admin</i> menekan tombol hafalans pada halaman hafalans lalu tidak memasukkan tgl, waktu, ayat dari, ayat ke, jmlh juz, hal dari, hal ke, keterangan, kategori, surat, santri kemudian tekan tombol <i>create</i> | Sistem akan memberikan error di halaman hafalans yang tersedia | Sesuai (100%) |

Pada tabel 4.5.3.4 Pengujian *Black Box Testing* – Hafalans, ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol hafalans pada halaman hafalans lalu memasukkan tgl, waktu, ayat dari, ayat ke, jmlh juz, hal dari, hal ke, keterangan, kategori, surat, santri kemudian tekan tombol *create* setelah itu *Admin* menekan tombol hafalans pada

halaman hafalans lalu tidak memasukkan tgl, waktu, ayat dari, ayat ke, jmlh juz, hal dari, hal ke, keterangan, kategori, surat, santri kemudian tekan tombol *create*

Pada tabel 4.5.3.5 Pengujian *Black Box Testing* – Kategoris

| Skenario Uji | | Kategoris | |
|--------------|--|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/kategoris | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol kategoris pada halaman kategoris lalu tekan tombol <i>new</i> kategoris lalu memasukkan nama kategoris kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |
| 2. | <i>Admin</i> menekan tombol kategoris pada halaman kategoris lalu tidak memasukkan nama kategoris kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman kategoris yang tersedia | Sesuai (100%) |

Pada tabel 4.5.3.5 Pengujian *Black Box Testing* – Kategoris ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol kategoris pada halaman kategoris lalu memasukkan nama kategoris kemudian tekan

tombol *create* setelah itu *Admin* menekan tombol kategoris pada halaman kategoris lalu tidak memasukkan nama kategoris kemudian tekan tombol *create*

Pada tabel 4.5.3.6 Pengujian *Black Box Testing* – Kehadirans

| Skenario Uji | | Kehadirans | |
|--------------|---|--|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/kehadirans | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol kehadiran pada halaman kehadiran lalu tekan tombol <i>new</i> kehadiran lalu memasukkan tgl, status, santri, kelompok kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |
| 2. | <i>Admin</i> menekan tombol kehadiran pada halaman kehadiran lalu tekan tombol <i>new</i> kehadiran lalu tidak memasukkan tgl, status, santri, kelompok kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman kehadiran yang tersedia | Sesuai (100%) |

Pada tabel 4.5.3.6 Pengujian *Black Box Testing* – Kehadirans ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol kehadiran pada halaman kehadiran lalu tekan tombol *new* kehadiran lalu memasukkan tgl, status, santri, kelompok kemudian tekan tombol *create* setelah itu *Admin* menekan tombol kehadiran pada halaman kehadiran lalu tekan tombol *new* kehadiran lalu tidak memasukkan tgl, status, santri, kelompok kemudian tekan tombol *create*.

Pada tabel 4.5.3.7 Pengujian *Black Box Testing* – Kelompok

| Skenario Uji | | Kelompok | |
|--------------|---|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/kelompok | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol kelompok pada halaman kelompok lalu tekan tombol <i>new</i> kelompok lalu memasukkan kategori santri, kelas, guru, kategori kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |
| 2. | <i>Admin</i> menekan tombol | Sistem akan memberikan <i>error</i> di halaman kelompok yang tersedia | Sesuai (100%) |

| | | |
|---|--|--|
| kelompok pada halaman kelompok lalu tekan tombol <i>new</i> kelompok lalu tidak memasukkan kategori santri, kelas, guru, kategori kemudian tekan tombol <i>create</i> | | |
|---|--|--|

Pada tabel 4.5.3.7 Pengujian *Black Box Testing* – Kelompok ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol kelompok pada halaman kelompok lalu tekan tombol *new* kelompok lalu memasukkan kategori santri, kelas, guru, kategori kemudian tekan tombol *create* setelah itu *Admin* menekan tombol kelompok pada halaman kelompok lalu tekan tombol *new* kelompok lalu tidak memasukkan kategori santri, kelas, guru, kategori kemudian tekan tombol *create*

Pada tabel 4.5.3.8 Pengujian *Black Box Testing* – Nilais

| Skenario Uji | | Nilais | |
|--------------|---|---|---------------|
| HTTP Method | | http://127.0.0.1:8000/admin/nilais | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol <i>nilais</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |

| | | | |
|----|--|--|----------------------|
| | <p>pada halaman nilais lalu tekan tombol <i>new</i> nilais lalu memasukkan santri, <i>score</i>, hafalan, <i>form</i> nilai kemudian tekan tombol <i>create</i></p> | | |
| 2. | <p><i>Admin</i> menekan tombol nilais pada halaman nilais lalu tekan tombol <i>new</i> nilais lalu tidak memasukkan santri, <i>score</i>, hafalan, <i>form</i> nilai kemudian tekan tombol <i>create</i></p> | <p>Sistem akan memberikan <i>error</i> di halaman <i>form</i> nilais yang tersedia</p> | <p>Sesuai (100%)</p> |

Pada tabel 4.5.3.8 Pengujian *Black Box Testing* – Nilais ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol nilais pada halaman

nilais lalu tekan tombol *new* nilais lalu memasukkan santri, *score*, hafalan, *form* nilai kemudian tekan tombol *create* setelah itu *Admin* menekan tombol nilais pada halaman nilais lalu tekan tombol *new* nilais lalu tidak memasukkan santri, *score*, hafalan, *form* nilai kemudian tekan tombol *create*

Pada tabel 4.5.3.9 Pengujian *Black Box Testing* – Santris

| Skenario Uji | | Santris | |
|--------------|--|---|----------------|
| HTTP Method | | http://127.0.0.1:8000/admin/santris | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol santris pada halaman santris lalu tekan tombol <i>new</i> santris lalu memasukkan santri, <i>score</i> , hafalan, kelompok kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai (100%) |
| 2. | <i>Admin</i> menekan tombol santris pada halaman santris lalu tekan tombol <i>new</i> santris lalu | Sistem akan memberikan <i>error</i> di halaman santris yang tersedia | Sesuai (100%) |

| | | |
|---|--|--|
| tidak memasukkan santri, <i>score</i> , hafalan, kelompok kemudian tekan tombol <i>create</i> | | |
|---|--|--|

Pada tabel 4.5.3.9 Pengujian *Black Box Testing* – Santris ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol santris pada halaman santris lalu tekan tombol *new* santris lalu memasukkan santri, *score*, hafalan, kelompok kemudian tekan tombol *create* setelah itu *Admin* menekan tombol santris pada halaman santris lalu tekan tombol *new* santris lalu tidak memasukkan santri, *score*, hafalan, kelompok kemudian tekan tombol *create*

Pada tabel 4.5.3.9.1 Pengujian *Black Box Testing* – Surats

| Skenario Uji | | Surats | |
|--------------|--|---|--------------|
| HTTP Method | | http://127.0.0.1:8000/admin/surats | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol surats pada halaman surats lalu tekan tombol <i>new</i> surats lalu memasukkan no surat, nama surat kemudian | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai(100%) |

| | | | |
|----|---|---|--------------|
| | tekan tombol <i>create</i> | | |
| 2. | <i>Admin</i> menekan tombol surats pada halaman surats lalu tekan tombol <i>new</i> surats lalu tidak memasukkan no surat, nama surat kemudian tekan tombol <i>create</i> | Sistem akan memberikan <i>error</i> di halaman surats yang tersedia | Sesuai(100%) |

Pada tabel 4.5.3.9.1 Pengujian *Black Box Testing* – Santris ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol surats pada halaman surats lalu tekan tombol *new* surats lalu memasukkan no surat, nama surat kemudian tekan tombol *create* setelah itu *Admin* menekan tombol surats pada halaman surats lalu tekan tombol *new* surats lalu tidak memasukkan no surat, nama surat kemudian tekan tombol *create*.

Pada tabel 4.5.3.9.2 Pengujian *Black Box Testing* – Ujians

| Skenario Uji | | Ujians | |
|--------------|---|---|--------------|
| HTTP Method | | http://127.0.0.1:8000/admin/ujians | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol ujians pada halaman ujians lalu | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai(100%) |

| | | | |
|----|--|--|---------------------|
| | <p>tekan tombol <i>new</i> ujian lalu memasukkan thn, periode, waktu mulai, waktu slsi, keterangan kemudian tekan tombol <i>create</i></p> | | |
| 2. | <p><i>Admin</i> menekan tombol ujians pada halaman ujians lalu tekan tombol <i>new</i> ujian lalu tidak memasukkan thn, periode, waktu mulai, waktu slsi, keterangan kemudian tekan tombol <i>create</i></p> | <p>Sistem akan memberikan <i>error</i> di halaman ujians yang tersedia</p> | <p>Sesuai(100%)</p> |

Pada tabel 4.5.3.9.2 Pengujian *Black Box Testing* – Santris ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol ujians pada halaman ujians lalu tekan tombol *new* ujian lalu memasukkan thn,

periode, waktu mulai, waktu slsi, keterangan kemudian tekan tombol *create* setelah itu *Admin* menekan tombol ujians pada halaman ujians lalu tekan tombol *new* ujian lalu tidak memasukkan thn, periode, waktu mulai, waktu slsi, keterangan kemudian tekan tombol *create*.

Pada tabel 4.5.3.9.3 Pengujian *Black Box Testing* – *Users*

| Skenario Uji | | <i>Users</i> | |
|--------------|--|---|--------------|
| HTTP Method | | http://127.0.0.1:8000/admin/users | |
| No | Butiran Uji | Hasil yang diharapkan | Hasil Uji |
| 1. | <i>Admin</i> menekan tombol <i>users</i> pada halaman <i>users</i> lalu tekan tombol <i>new users</i> lalu memasukkan nama, <i>email</i> , <i>password</i> , profil <i>photo path</i> , role kemudian tekan tombol <i>create</i> | Sistem dapat menyimpan data yang telah dimasukan ke <i>database</i> | Sesuai(100%) |
| 2. | <i>Admin</i> menekan tombol <i>users</i> pada halaman <i>users</i> lalu tekan tombol | Sistem akan memberikan <i>error</i> di halaman <i>users</i> yang tersedia | Sesuai(100%) |

| | | |
|---|--|--|
| <p><i>new users</i> lalu tidak memasukkan nama, <i>email</i>, <i>password</i>, profil <i>photo</i> <i>path</i>, role kemudian tekan tombol <i>create</i></p> | | |
|---|--|--|

Pada tabel 4.5.3.9.3 Pengujian *Black Box Testing – Users* ini digunakan sebagai pengujian mulai dari *Admin* menekan tombol *users* pada halaman *users* lalu tekan tombol *new users* lalu memasukkan nama, *email*, *password*, profil *photo path*, role kemudian tekan tombol *create* setelah itu *Admin* menekan tombol *users* pada halaman *users* lalu tekan tombol *new users* lalu tidak memasukkan nama, *email*, *password*, profil *photo path*, role kemudian tekan tombol *create*.

4.5.4 Pengujian *User Acceptance Test (UAT)*

Tabel 4.5.4.1 Pengujian UAT *Admin*

| No | Fitur | Proses | Hasil yang diharapkan | Hasil Tes |
|----|------------------|---|--|-----------|
| 1. | <i>Login</i> | <i>Admin</i> memasukan <i>username</i> dan <i>password</i> | <i>Admin</i> berhasil <i>login</i> | Berhasil |
| 2. | <i>Dashboard</i> | <i>Admin</i> berhasil <i>login</i> , setelah itu sistem otomatis mengarahkan ke | <i>Admin</i> berhasil melihat <i>dashboard</i> | Berhasil |

| | | | | |
|-----|-------------------|--|--|----------|
| | | halaman <i>dashboard</i> | | |
| 3. | Tahun Akademik | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu tahun akademik | <i>Admin</i> berhasil mengelola tahun akademik | Berhasil |
| 4. | Ustadz/Guru | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu ustadz/guru | <i>Admin</i> berhasil mengelola ustadz/guru | Berhasil |
| 5. | Kelas | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu kelas | <i>Admin</i> berhasil mengelola kelas | Berhasil |
| 6. | Kelompok | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu kelompok | <i>Admin</i> berhasil mengelola kelompok | Berhasil |
| 7. | Santri | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu santri | <i>Admin</i> berhasil mengelola santri | Berhasil |
| 8. | <i>User</i> | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu <i>user</i> | <i>Admin</i> berhasil mengelola <i>user</i> | Berhasil |
| 9. | Jadwal Ujian | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu jadwal ujian | <i>Admin</i> berhasil mengelola jadwal ujian | Berhasil |
| 10. | Kategori | <i>Admin</i> berhasil <i>login</i> , setelah itu | <i>Admin</i> berhasil mengelola kategori | Berhasil |

| | | | | |
|-----|-------|--|---------------------------------------|----------|
| | | klik menu kategori | | |
| 11. | Surat | <i>Admin</i> berhasil <i>login</i> , setelah itu klik menu surat | <i>Admin</i> berhasil mengelola surat | Berhasil |

Pada Tabel 4.5.4.1 Pengujian UAT *Admin* berikut penjelasannya:

1. *Login*

- a. **Proses:** *Admin* memasukkan *username* dan *password* pada halaman *login*.
- b. **Hasil yang Diharapkan:** *Admin* berhasil *login* ke dalam sistem.
- c. **Hasil Tes:** Berhasil. Artinya, *admin* dapat masuk ke dalam sistem dengan memasukkan *username* dan *password* yang benar.

2. *Dashboard*

- a. **Proses:** Setelah *login*, sistem otomatis mengarahkan *admin* ke halaman *dashboard*.
- b. **Hasil yang Diharapkan:** *Admin* dapat melihat tampilan *dashboard* yang berisi informasi umum dan akses ke fitur-fitur lain di dalam aplikasi.
- c. **Hasil Tes:** Berhasil. Ini berarti sistem mengarahkan *admin* ke halaman *dashboard* dan *admin* dapat melihat tampilan *dashboard* dengan benar.

3. **Tahun Akademik**

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Tahun Akademik."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola data tahun akademik, seperti menambah, mengedit, atau menghapus tahun akademik.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan tahun

akademik berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

4. Ustadz/Guru

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Ustadz/Guru."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola data ustadz/guru, seperti menambah, mengedit, atau menghapus data ustadz/guru.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan data ustadz/guru berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

5. Kelas

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Kelas."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola data kelas, seperti menambah, mengedit, atau menghapus data kelas.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan data kelas berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

6. Kelompok

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Kelompok."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola data kelompok, seperti menambah, mengedit, atau menghapus data kelompok.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan data kelompok berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

7. Santri

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Santri."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola data santri, seperti

menambah, mengedit, atau menghapus data santri.

- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan data santri berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

8. *User*

- a. **Proses:** Setelah *login*, *admin* mengklik menu "*User*."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola data *user*, seperti menambah, mengedit, atau menghapus data *user*.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan data *user* berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

9. Jadwal Ujian

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Jadwal Ujian."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola jadwal ujian, seperti menambah, mengedit, atau menghapus jadwal ujian.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan jadwal ujian berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

10. Kategori

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Kategori."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola kategori, seperti menambah, mengedit, atau menghapus kategori.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan kategori berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

11. Surat

- a. **Proses:** Setelah *login*, *admin* mengklik menu "Surat."
- b. **Hasil yang Diharapkan:** *Admin* dapat mengelola surat, seperti menambah, mengedit, atau menghapus surat.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur pengelolaan surat berfungsi dengan baik dan *admin* dapat melakukan manajemen data tersebut.

Berdasarkan hasil UAT yang telah dilakukan, semua fitur utama yang diuji (*Login*, *Dashboard*, Tahun Akademik, Ustadz/Guru, Kelas, Kelompok, Santri, *User*, Jadwal Ujian, Kategori, dan Surat) berfungsi dengan baik dan sesuai dengan hasil yang diharapkan. Pengguna (*Admin*) dapat mengakses dan mengelola informasi yang relevan setelah melakukan *login* dan mengklik menu yang sesuai.

Tabel 4.5.4.2 Pengujian UAT Ustadz/Guru

| No | Fitur | Proses | Hasil yang diharapkan | Hasil Tes |
|----|------------------|---|---|-----------|
| 1. | <i>Dashboard</i> | Ustadz/Guru berhasil <i>login</i> , setelah itu sistem otomatis mengarahkan ke halaman <i>dashboard</i> | Ustadz/Guru berhasil melihat <i>dashboard</i> | Berhasil |
| 2. | Hafalan | Ustadz/Guru berhasil <i>login</i> , setelah itu klik menu hafalan | Ustadz/Guru berhasil mengelola hafalan | Berhasil |
| 3. | Jadwal Ujian | Ustadz/Guru berhasil <i>login</i> , setelah itu klik menu jadwal ujian | Ustadz/Guru berhasil mengelola jadwal ujian | Berhasil |
| 4. | Nilai | Ustadz/Guru berhasil <i>login</i> , setelah itu klik menu nilai | Ustadz/Guru berhasil mengelola nilai | Berhasil |

Pada Tabel 4.5.4.2 Pengujian UAT Ustadz/Guru berikut penjelasannya:

1. **Dashboard**

- a. **Proses:** Setelah Ustadz/Guru berhasil *login* ke dalam sistem, sistem akan secara otomatis mengarahkan pengguna ke halaman *dashboard*.
- b. **Hasil yang Diharapkan:** Ustadz/Guru dapat melihat tampilan *dashboard* yang berisi informasi umum dan akses ke fitur-fitur lain di dalam aplikasi.
- c. **Hasil Tes:** Berhasil. Ini berarti setelah *login*, sistem benar-benar mengarahkan pengguna ke halaman *dashboard* dan pengguna dapat melihat tampilan *dashboard* dengan benar.

2. **Hafalan**

- a. **Proses:** Setelah *login*, Ustadz/Guru perlu mengklik menu "Hafalan" yang tersedia di *dashboard* atau menu utama.
- b. **Hasil yang Diharapkan:** Setelah mengklik menu "Hafalan," Ustadz/Guru dapat mengelola hafalan santri, seperti menambah, mengedit, atau menghapus data hafalan.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur hafalan berfungsi dengan baik dan Ustadz/Guru dapat mengelola hafalan santri sesuai dengan kebutuhan.

3. **Jadwal Ujian**

- a. **Proses:** Setelah *login*, Ustadz/Guru mengklik menu "Jadwal Ujian."
- b. **Hasil yang Diharapkan:** Setelah mengklik menu "Jadwal Ujian," Ustadz/Guru dapat mengelola jadwal ujian, termasuk menambah, mengedit, atau menghapus jadwal ujian.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa informasi jadwal ujian dapat dikelola dengan benar oleh Ustadz/Guru melalui menu yang sesuai.

4. Nilai

- a. **Proses:** Setelah *login*, Ustadz/Guru mengklik menu "Nilai."
- b. **Hasil yang Diharapkan:** Setelah mengklik menu "Nilai," Ustadz/Guru dapat mengelola nilai santri, seperti memasukkan, mengedit, atau menghapus data nilai.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa sistem memungkinkan Ustadz/Guru untuk mengelola nilai santri dengan benar ketika diakses melalui menu yang sesuai.

Berdasarkan hasil UAT yang telah dilakukan, semua fitur utama yang diuji (*Dashboard*, Hafalan, Jadwal Ujian, dan Nilai) berfungsi dengan baik dan sesuai dengan hasil yang diharapkan. Pengguna (Ustadz/Guru) dapat mengakses dan mengelola informasi yang relevan setelah melakukan *login* dan mengklik menu yang sesuai.

Tabel 4.5.4.3 Pengujian UAT Wali Santri

| No | Fitur | Proses | Hasil yang diharapkan | Hasil Tes |
|----|------------------|---|---|-----------|
| 1. | <i>Dashboard</i> | Wali Santri berhasil <i>login</i> , setelah itu sistem otomatis mengarahkan ke halaman <i>dashboard</i> | Wali Santri berhasil melihat <i>dashboard</i> | Berhasil |
| 2. | Hafalan | Wali Santri berhasil <i>login</i> , setelah itu klik menu hafalan | Wali Santri berhasil melihat hafalan | Berhasil |
| 3. | Jadwal Ujian | Wali Santri berhasil <i>login</i> , setelah itu klik menu jadwal ujian | Wali Santri berhasil melihat jadwal ujian | Berhasil |
| 4. | Nilai | Wali Santri berhasil <i>login</i> , setelah itu klik menu nilai | Wali Santri berhasil melihat nilai | Berhasil |

Pada Tabel 4.5.4.3 Pengujian UAT Wali Santri berikut penjelasannya:

1. *Dashboard*

- a. **Proses:** Setelah Wali Santri berhasil *login* ke dalam sistem, sistem akan secara otomatis mengarahkan pengguna ke halaman *dashboard*.
- b. **Hasil yang Diharapkan:** Wali Santri dapat melihat tampilan *dashboard* yang berisi informasi umum dan akses ke fitur-fitur lain di dalam aplikasi.
- c. **Hasil Tes:** Berhasil. Artinya, setelah *login*, sistem benar-benar mengarahkan pengguna ke halaman *dashboard* dan pengguna dapat melihat tampilan *dashboard* dengan benar.

2. *Hafalan*

- a. **Proses:** Setelah *login*, Wali Santri perlu mengklik menu "Hafalan" yang tersedia di *dashboard* atau menu utama.
- b. **Hasil yang Diharapkan:** Setelah mengklik menu "Hafalan," Wali Santri dapat melihat daftar hafalan yang telah diselesaikan atau sedang dikerjakan.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa fitur hafalan berfungsi dengan baik dan menampilkan informasi yang sesuai ketika diakses oleh pengguna.

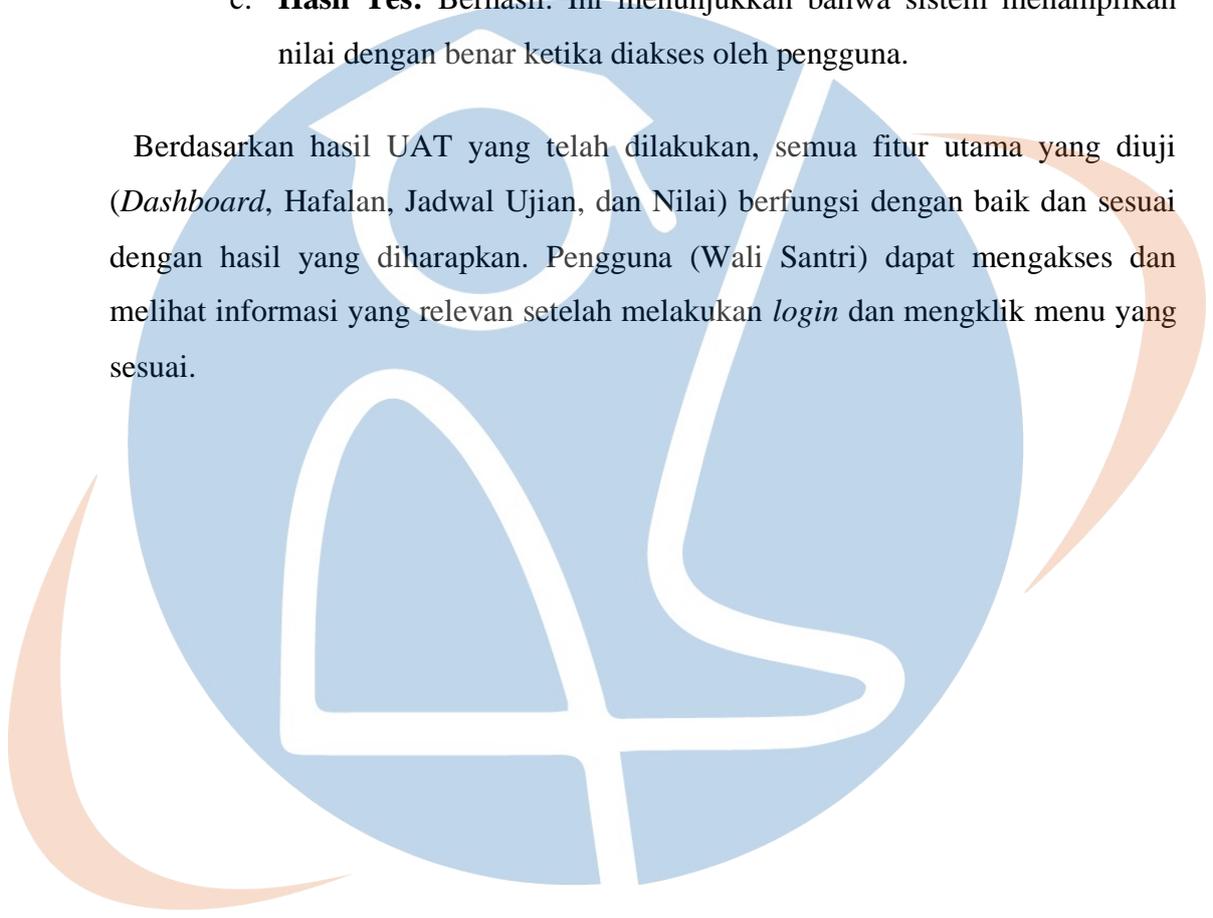
3. *Jadwal Ujian*

- a. **Proses:** Setelah *login*, Wali Santri mengklik menu "Jadwal Ujian."
- b. **Hasil yang Diharapkan:** Setelah mengklik menu "Jadwal Ujian," Wali Santri dapat melihat jadwal ujian yang telah ditentukan, termasuk tanggal, waktu, dan mata pelajaran.
- c. **Hasil Tes:** Berhasil. Artinya, informasi jadwal ujian ditampilkan dengan benar setelah diakses melalui menu yang sesuai.

4. Nilai

- a. **Proses:** Setelah *login*, Wali Santri mengklik menu "Nilai."
- b. **Hasil yang Diharapkan:** Setelah mengklik menu "Nilai," Wali Santri dapat melihat daftar nilai untuk setiap mata pelajaran atau ujian yang telah diikuti.
- c. **Hasil Tes:** Berhasil. Ini menunjukkan bahwa sistem menampilkan nilai dengan benar ketika diakses oleh pengguna.

Berdasarkan hasil UAT yang telah dilakukan, semua fitur utama yang diuji (*Dashboard*, Hafalan, Jadwal Ujian, dan Nilai) berfungsi dengan baik dan sesuai dengan hasil yang diharapkan. Pengguna (Wali Santri) dapat mengakses dan melihat informasi yang relevan setelah melakukan *login* dan mengklik menu yang sesuai.



STT - NF

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian tugas akhir yang berjudul “Rancang Bangun Sistem Informasi Monitoring Hafalan Al Qur’an Santri (Be-Hafizh) Berbasis Web Menggunakan *Framework Laravel*” menghasilkan beberapa kesimpulan, yaitu:

- 1 Aplikasi be-hafizh dalam penelitian ini melibatkan berbagai analisis, seperti analisis kebutuhan pengguna, identifikasi *actor*, wawancara serta pembuatan *activity* diagram, *class* diagram, dan *use case* diagram. Implementasi dari perancangan ini menggunakan *framework laravel*. Pada tahap akhir, pengujian sistem dilakukan berdasarkan rancangan pengujian yang telah dibuat sebelumnya, menggunakan metode *black box testing*.
- 2 Berdasarkan hasil *black box testing* yang menunjukkan persentase keberhasilan 100%, hal ini mengindikasikan bahwa aplikasi be-hafizh telah memenuhi ekspektasi yang diharapkan.

5.2 Saran

Berdasarkan dari pengkajian hasil penelitian dilapangan maka penulis memberikan saran yang mudah-mudahan dapat bermanfaat bagi Lembaga maupun bagi peneliti yang selanjutnya, yaitu sebagai berikut: Bagi Pihak Lembaga seperti yang penulis sudah jelaskan, penginputan data setoran hafalan Al-Qur’an merupakan salah satu sumber yang sangat dibutuhkan di Pondok Pesantren Takhasuss Tahfidz Al-Ma’shum Mardiyah untuk itu perlu dilakukan digitalisasi dalam penginputan data setoran hafalan Al-Qur’an guna mempermudah efektivitas pembelajaran di Pondok Pesantren Takhasuss Tahfidz Al-Ma’shum Mardiyah. Bagi Peneliti Selanjutnya adapun beberapa saran yang perlu diperhatikan bagi peneliti selanjutnya yang tertarik meneliti tentang penginputan data setoran hafalan Al-Qu’ran Pondok Pesantren Takhasuss Tahfidz Al-Ma’shum Mardiyah adalah: Peneliti selanjutnya diharapkan untuk mengkaji lebih banyak sumber maupun referensi yang terkait dengan penginputan data setoran hafalan Al-Qu’ran di Pondok Pesantren Takhasuss Tahfidz Al-Ma’shum Mardiyah maupun efektivitas proses

pembelajaran agar hasil penelitiannya dapat lebih baik dan lebih lengkap lagi. Peneliti selanjutnya diharapkan lebih mempersiapkan diri dalam proses pengambilan dan pengumpulan dan segala sesuatunya sehingga penelitian dapat dilaksanakannya dengan lebih baik. Peneliti selanjutnya diharapkan ditunjang pula dengan wawancara dengan sumber kompeten dalam kajian penginputan data setoran hafalan Al-Qu'ran Pondok Pesantren Takhasuss Tahfidz Al-Ma'shum Mardiyah.



DAFTAR PUSTAKA

- [1] M. YMM, “Pesantren Takhassus Tahfizd Al Ma’shum Mardiah,” *Pesantren Tak. Tahfizd Al Ma’shum Mardiah*, 2023, [Online]. Available: <https://www.almashummardiyah.or.id/Profil-Lengkap/>
- [2] B. Hermanto, M. Yusman, J. Ilmu Komputer FMIPA Universitas Lampung Jalan Sumantri Brojonegoro No, and B. Lampung, “SISTEM INFORMASI MANAJEMEN KEUANGAN PADA PT. HULU BALANG MANDIRI MENGGUNAKAN FRAMEWORK LARAVEL,” 2019.
- [3] M. jeni lumban gaol Yohana, “Sistem Informasi Akuntansi,” *Konsep-konsep Dasar Sist. Inf. Akunt.*, p. 3, 2021, [Online]. Available: <https://pustaka.ut.ac.id/lib/wp-content/uploads/pdfmk/EKSI431203-M1.pdf>
- [4] Y. S. Novitasari, Q. J. Adrian, and W. Kurnia, “RANCANG BANGUN SISTEM INFORMASI MEDIA PEMBELAJARAN BERBASIS WEBSITE (STUDI KASUS: BIMBINGAN BELAJAR DE POTLOOD),” *J. Teknol. dan Sist. Inf.*, vol. 2, no. 3, pp. 136–147, 2021, [Online]. Available: <http://jim.teknokrat.ac.id/index.php/JTSI>
- [5] K. Wijaya, R. Supariyanto, and E. Istiawan, “IMPLEMENTASI FRAMEWORK BOOTSTRAP DALAM PERANCANGAN SISTEM PENERIMAAN MAHASISWA BARU PADA SEKOLAH TINGGI ILMU TARBIYAH AL-QURAN AL-ITTIFAQIAH (STITQI) INDRAALAYABERBASIS WEB,” *J. Sist. Inf. dan Komputerisasi Akunt.*, vol. 04, no. 02.
- [6] M. Yusup, D. Aryani, and S. Suhendi, “Desain Aplikasi Tracer Study Berbasis Web Menggunakan Laravel Framework,” *J. CERITA*, vol. 5, no. 2, pp. 215–222, 2019, doi: 10.33050/cerita.v5i2.506.
- [7] A. Setiawan, M. Fauzi Makarim, R. Fathoni Andi Rafif, and D. Haryo Sulaksono, “Sistem Informasi Total Penjualan SPBU Berbasis Web,” *Pros. Semin. Implementasi Teknol. Inf. dan Komun.*, vol. 1, no. 1, pp. 25–32, 2022, doi: 10.31284/p.semtik.2022-1.2491.
- [8] Edy Soesanto, Alifah Jiddal Masyruroh, Ganis Aliefiani Mulya Putri, and

- Srirahayu Putri Maharani, “Peranan Manajemen Sekuriti Dalam Mengamankan Dan Memecahkan Masalah PT SK Keris Indonesia,” *J. Manaj. Ris. Inov.*, vol. 1, no. 3, pp. 46–57, 2023, doi: 10.55606/mri.v1i3.1259.
- [9] A. T. Prakoso, S. Davin, N. Y. Mahendra, M. A. A. Saputra, and H. Basri, “Pemanfaatan Limbah Plastik Dalam Pembuatan Filamen 3D Printer Menggunakan Mesin Ekstrusi Pada Lab Konversi Energi Universitas Sriwijaya,” *J. Pelita Sriwij.*, vol. 1, no. 2, pp. 043–052, 2022, doi: 10.51630/jps.v1i2.69.
- [10] N. Penulis and M. Rahman, “Pelatihan Membangun Web Dinamis Dengan Model View Controller (MVC) Pada Siswa SMK Tritech Informatika Medan Corresponding Author,” vol. 1, no. 10, 2023, [Online]. Available: <https://jurnalpengabdianmasyarakatbangsa.com/index.php/jpmba/index>
- [11] T. Wijaya, “Implementasi Web API pada Aplikasi Desktop untuk Meningkatkan Performa Aplikasi,” vol. XII, no. 1, pp. 497–506, 2023.
- [12] E. Yandani and D. Winarti, “Perancangan Sistem Informasi Akademik Perguruan Tinggi Berbasis Web,” *Simtika*, vol. 2, no. 3, pp. 21–27, 2019, [Online]. Available: <http://lppm.undhari.ac.id/ejournal/index.php/simtika/article/view/51>
- [13] Q. E. Fazrin, T. Lisnawati, S. Nurhayati, J. B. Satya, and D. Alamsyah, “Penerapan Metode Pengembangan Sistem Extreme Programing (XP) Pada Aplikasi Presensi Karyawan dengan QR Code,” *Build. Informatics, Technol. Sci.*, vol. 3, no. 3, pp. 164–170, 2021, doi: 10.47065/bits.v3i3.1018.
- [14] A. Trisnadoli, “Implementasi Extreme Programming (XP) Agile Software Development pada Pengembangan Sistem Informasi KELUARGAKU,” vol. 6, no. 2, pp. 305–311, 2021, doi: 10.32493/informatika.v6i2.10088.
- [15] “PERANCANGAN SISTEM BERORIENTASI OBJEK DENGAN PEMODELAN UML (UNIFIED MODELING LANGUAGE) TOOLS.”
- [16] A. Fitriadi and H. A. Tawakal, “Jurnal Informatika Terpadu,” *J. Inform. Terpadu*, vol. 7, no. 2, pp. 62–69, 2021, [Online]. Available: <https://journal.nurulfikri.ac.id/index.php/JIT>
- [17] J. Wahyudi, “Pembangunan Sistem Aplikasi PenyewaanMobil Berbasis

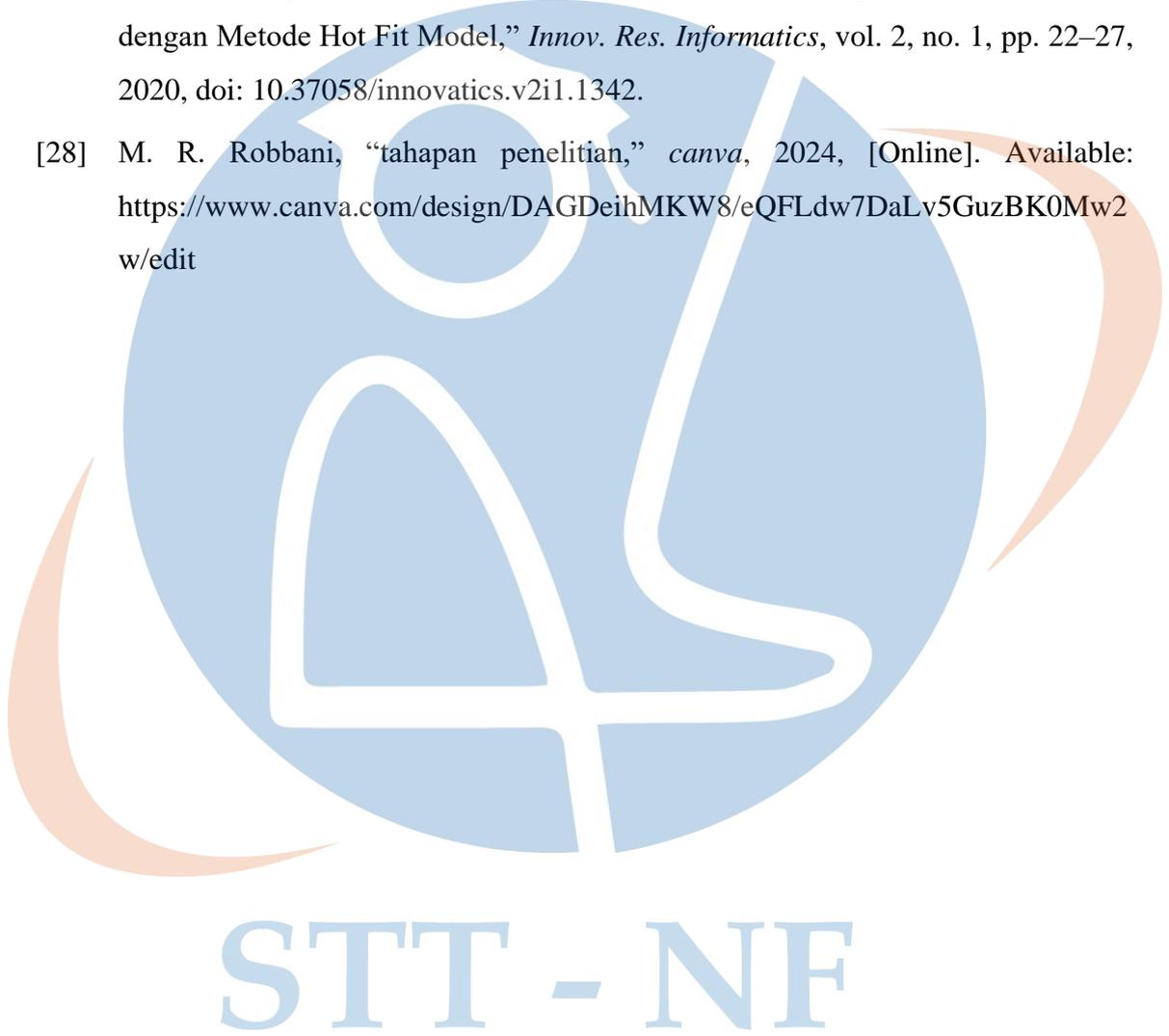
- AndroidStudi Kasus : Cv. Amanah Kalimantan Rent,” *J. Jieom*, vol. 2, no. 1, pp. 10–16, 2019.
- [18] D. E. Cahyono and A. Jayanti, “Implementasi Aplikasi Kasir Berbasis Web pada Toko Ghafya Fruits Shop,” *J. Ekon. dan Tek. Inform.*, vol. 10, no. 1, pp. 32–40, 2022.
- [19] M. Tabrani and I. R. Aghniya, “Implementasi Metode Waterfall Pada Program Simpan Pinjam Koperasi Subur Jaya Mandiri Subang,” *J. Interkom J. Publ. Ilm. Bid. Teknol. Inf. dan Komun.*, vol. 14, no. 1, pp. 44–53, Apr. 2019, doi: 10.35969/interkom.v14i1.46.
- [20] J. D. Rahardjo, R. Tullah, and H. Setiana, “Sistem Informasi Pemesanan Dan Pembelian Tiket Bus Online Berbasis Web Pada P.O. Budiman,” *J. Sisfotek Glob.*, vol. 9, no. 2, pp. 120–125, 2019, doi: 10.38101/sisfotek.v9i2.259.
- [21] H. Muttu, K. K. Yuliawan, and P. Rantelinggi, “Aplikasi Pengenalan Organ Tubuh Manusia Berbasis Android Menggunakan App Inventor Di Jurusan Biologi Universitas Papua,” *INFORMAL Informatics J.*, vol. 6, no. 1, p. 28, 2021, doi: 10.19184/isj.v6i1.23108.
- [22] N. Suryana and S. D. Yulianti, “Aplikasi Penjadwalaan Manajemen Artis Daily Schedule (Studi Kasus : Pt . Tetap Seratus Selamanya),” *J. Apl. Penjadwalaan Manaj. Artis Dly. Sched.*, vol. 7, no. 2, pp. 149–158, 2021, [Online]. Available: [https://maklumatika.i-tech.ac.id/index.php/maklumatika/article/download/109/114#:~:text=\(Sari%2C2017%3A83\),komputerisasi yang dilakukan oleh pengguna.](https://maklumatika.i-tech.ac.id/index.php/maklumatika/article/download/109/114#:~:text=(Sari%2C2017%3A83),komputerisasi yang dilakukan oleh pengguna.)
- [23] A. Wantoro, “Sistem Informasi Berbasis Web Untuk Pengelolaan Penerima Dana Zakat, Infaq Dan Sedekah,” *J. Tekno Kompak*, vol. 13, no. 2, p. 31, 2019, doi: 10.33365/jtk.v13i2.338.
- [24] Renaldy and A. Rustam, “Perancangan Sistem Informasi Inventory Berbasis Web Pada Gudang Di Pt. Spin Warriors,” *J. Homepage*, vol. 4, no. 1, pp. 27–32, 2020, [Online]. Available: <http://jti.aisyahuniversity.ac.id/index.php/AJIEE>
- [25] B. Priyatna, A. Lia Hananto, M. Nova, P. Studi Sistem Informasi, and U. Buana

Perjuangan Karawang, “Application of UAT (User Acceptance Test) Evaluation Model in Minggon E-Meeting Software Development,” *Systematics*, vol. 2, no. 3, pp. 110–117, 2020.

[26] A. Subianto, *Kebijakan Publik Tinjauan Perencanaan < Implementasi dan Evaluasi*. 2020.

[27] T. Lestariningsih, B. Artono, and Y. Afandi, “Evaluasi Implementasi E-learning dengan Metode Hot Fit Model,” *Innov. Res. Informatics*, vol. 2, no. 1, pp. 22–27, 2020, doi: 10.37058/innovatics.v2i1.1342.

[28] M. R. Robbani, “tahapan penelitian,” *canva*, 2024, [Online]. Available: <https://www.canva.com/design/DAGDeihMKW8/eQFLdw7DaLv5GuzBK0Mw2w/edit>



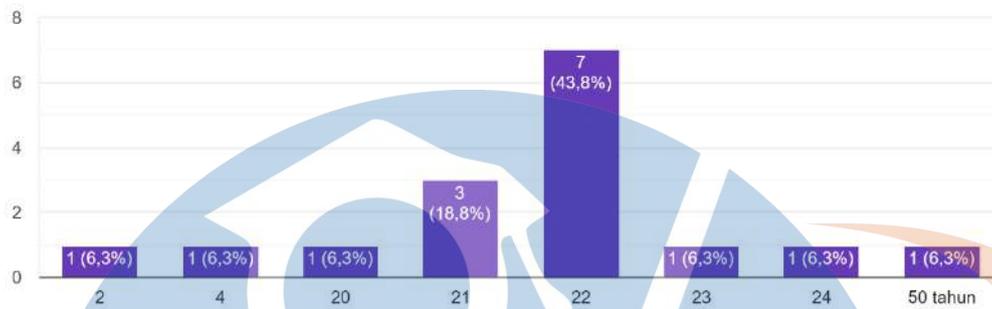
STT - NF

LAMPIRAN

Lampiran 1 Hasil Kuesioner

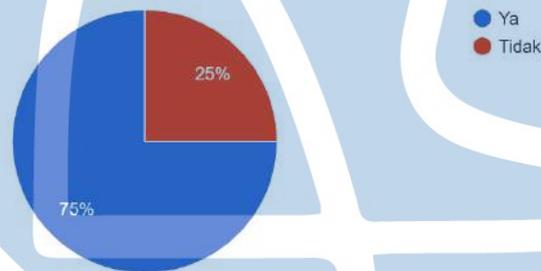
Usia

16 jawaban



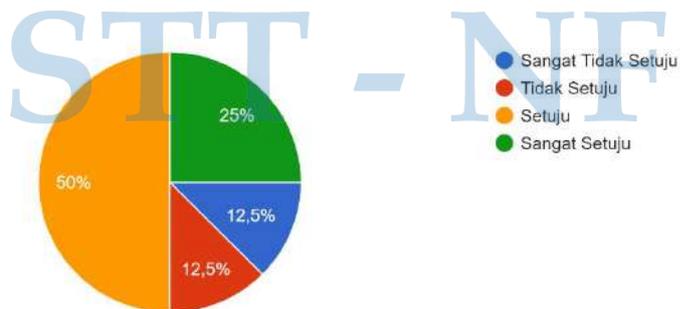
Apakah saat melakukan registrasi data dapat disimpan dan dapat masuk ke halaman utama sistem?

16 jawaban

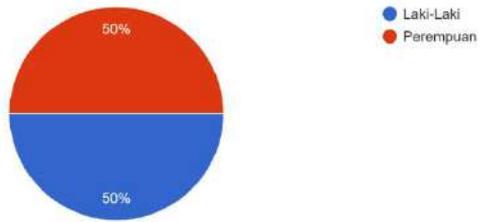


Apakah website mudah digunakan

16 jawaban



Jenis Kelamin
16 jawaban



Lampiran 2 Dokumentasi dan Hasil Wawancara



STT - NF

| No | Pertanyaan | Jawaban |
|----|---|---|
| 1 | Assalamualaikum Warahmatullahi Wabarakatuh kak | Waalaikumussalam Warahmatullahi Wabarakatuh rifqi |
| 2 | maaf sebelumnya mengganggu waktu nya rifqi mau mewawancarai kakak terkait dengan pondok pesantren Takhassus tahfidz al ma'shum mardiyah apakah diperbolehkan kak? | silahkan aja rifqi in syaa allah kakak jawab apabila sepengetahuan kakak aja y rifqi |
| 3 | baik kakak, bagaimana dengan penginputan data setoran hafalan al-quran di pondok pesantren ini kak? | baik ukhti jawab y rifqi, sebagaimana yang rifqi ketahui selama rifqi mengajar dan ukhti mengajar di pondok pesantren ini dalam penginputan data setoran hafalan al-quran setiap bulan nya masih menggunakan <i>Microsoft word, Microsoft Excel</i> rifqi |
| 4 | apakah penginputan secara menggunakan Microsoft word dan excel itu perlu menggunakan banyak banget waktu kesita kak? | benar banget rifqi banyak sekali waktu yang ukhti gunakan untuk menginput secara manual seperti itu. |
| 5 | siapa sebenarnya yang bagian menginput data setoran hafalan al-qur'an santri tersebut? | biasanya yang menginput data setoran hafalan al-qur'an santri adalah ustadz/guru nya kak akan tetapi dibantu juga dengan ukhti sebagai bagian kurikulum pondok kak dikarenakan SDM pondok kurang memadai maka di bantu dengan ukhti kak. |

| | | |
|---|--|---|
| 6 | bagian siapa kak yang mengelola data hafalan al-quran santri setiap bulanan? | biasanya yang bagian mengelola data hafalan al-quran santri setiap bulanan itu bagian administrasi kak kalau biasanya diperusahaan itu adalah bagian TU akan tetapi dibantu juga dengan ukhti sebagai bagian kurikulum pondok kak dikarenakan SDM pondok kurang memadai maka di bantu dengan ukhti kak. |
| 7 | dimana tempat untuk menginput dan mengelola data hafalan al-quran santri kak? | biasanya tempat yang digunakan untuk menginput dan mengelola data hafalan al-quran santri kak itu di kantor pondok pesantren nya kak. |
| 8 | baik terimakasih banyak kak atas waktunya sudah mau rifqi wawancarai , rifqi izin pamit wassalamualaikum warahmatullahi wabarakatuh kak. | Walaikumussalam Warahmatullahi Wabarakatuh rifqi. |

| Keterangan | |
|------------|--|
| Q | ust Taufiq Hidayatullah S.Pd.i (1), Ibu Aqilatul Muallamah S.Pd.i (2), kak Mayang Fauni S.si (3) |
| A | Muhammad Rifqi Robbani (pewawancara) |

Lampiran 3 Lampiran Kode

Lampiran kode : *controller.php*, *santripolici.php*, *gurupolici.php*

1. *controller.php*

```
<?php
namespace App\Http\Controllers;

use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
use Illuminate\Foundation\Validation\ValidatesRequests;
use Illuminate\Routing\Controller as BaseController;
```

```
class Controller extends BaseController
{
    use AuthorizesRequests, ValidatesRequests;
}
```

2. gurupolici.php

```
<?php

namespace App\Policies;

use App\Models\Guru;
use App\Models\User;
use Illuminate\Auth\Access\Response;

class GuruPolicy
{
    /**
     * Determine whether the user can view any models.
     */
    public function viewAny(User $user): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can view the model.
     */
    public function view(User $user, Guru $guru): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can create models.
     */
    public function create(User $user): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can update the model.
     */
}
```

```

    */
    public function update(User $user, Guru $guru): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can delete the model.
     */
    public function delete(User $user, Guru $guru): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can restore the model.
     */
    public function restore(User $user, Guru $guru): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can permanently delete the model.
     */
    public function forceDelete(User $user, Guru $guru): bool
    {
        return $user->isAdmin();
    }
}

```

3. santripolici.php

```

<?php

namespace App\Policies;

use App\Models\Santri;
use App\Models\User;
use Illuminate\Auth\Access\Response;

class SantriPolicy

```

```

{
    /**
     * Determine whether the user can view any models.
     */
    public function viewAny(User $user): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can view the model.
     */
    public function view(User $user, Santri $santri): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can create models.
     */
    public function create(User $user): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can update the model.
     */
    public function update(User $user, Santri $santri): bool
    {
        return $user->isAdmin();
    }

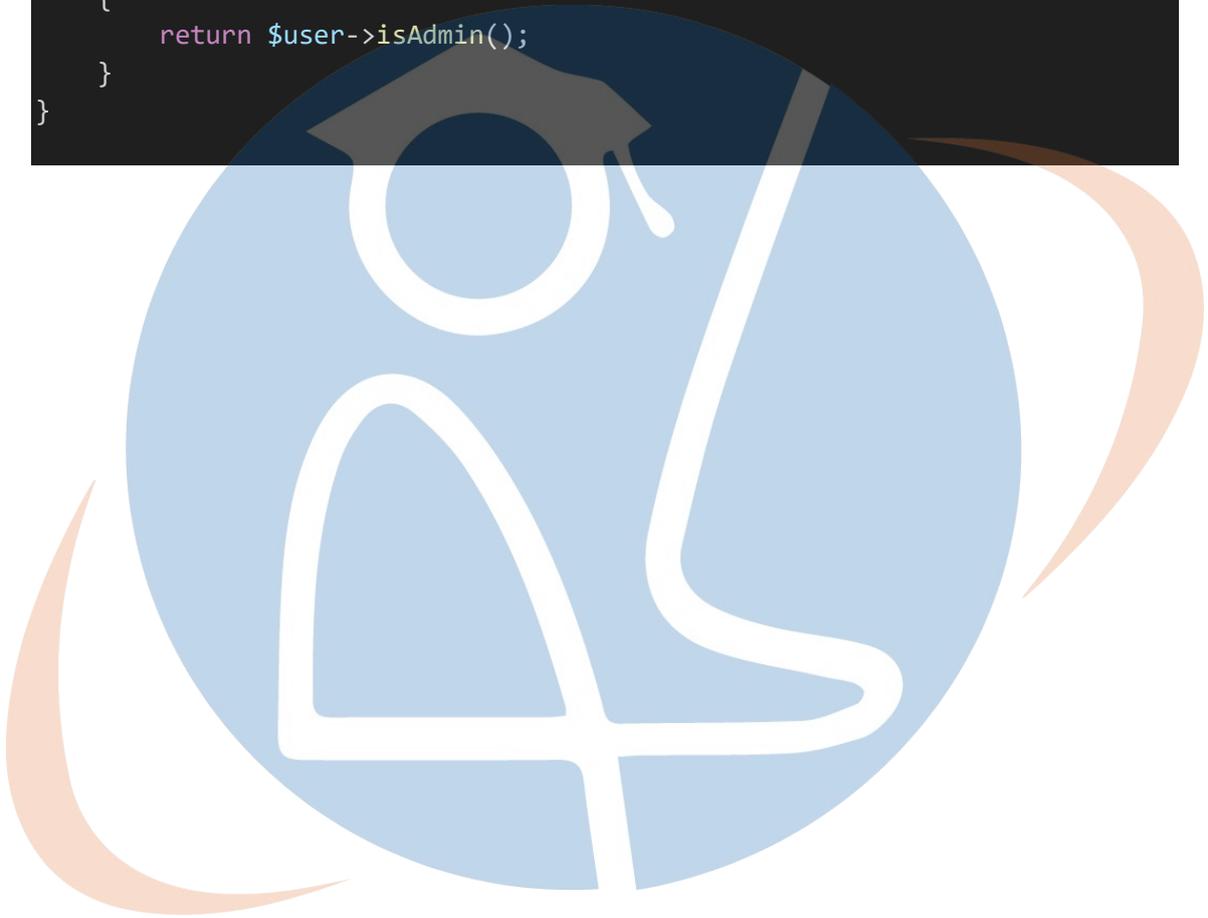
    /**
     * Determine whether the user can delete the model.
     */
    public function delete(User $user, Santri $santri): bool
    {
        return $user->isAdmin();
    }

    /**
     * Determine whether the user can restore the model.
     */
    public function restore(User $user, Santri $santri): bool

```

```
{
    return $user->isAdmin();
}

/**
 * Determine whether the user can permanently delete the model.
 */
public function forceDelete(User $user, Santri $santri): bool
{
    return $user->isAdmin();
}
}
```



STT - NF