PENGEMBANGAN *LEARNING MANAGEMENT SYSTEM* RUMAH/PONDOK TAHFIZH DENGAN *SERVICE ORIENTED ARCHITECTURE*



Disusun Oleh:

Nama : Mathori Hasan Maraghi

NIM : 14523268

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

2021

HALAMAN PENGESAHAN DOSEN PEMBIMBING

PENGEMBANGAN LEARNING MANAGEMENT SYSTEM RUMAH/PONDOK TAHFIZH DENGAN *SERVICE ORIENTED ARCHITECTURE*

TUGAS AKHIR

Disusun Oleh:

N a m a : Mathori Hasan Maraghi

NIM : 14523268

Yogyakarta, 29 November 2021

Pembimbing,



(Andhik Budi Cahyono, S.T., M.T.)

HALAMAN PENGESAHAN DOSEN PENGUJI

PENGEMBANGAN LEARNING MANAGEMENT SYSTEM RUMAH/PONDOK TAHFIZH DENGAN *SERVICE ORIENTED ARCHITECTURE*

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 07 Desember 2021

Tim Penguji

Andhik Budi Cahyono, S.T., M.T.

Anggota 1

Kholid Haryono, S.T., M.Kom.

Anggota 2

Lizda Iswari, S.T., M.Sc.

Mengetahui,
Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia

(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Mathori Hasan Maraghi

NIM : 14523268

Tugas akhir dengan judul:

PENGEMBANGAN LEARNING MANAGEMENT SYSTEM RUMAH/PONDOK TAHFIZH DENGAN SERVICE ORIENTED ARCHITECTURE

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 29 Nopember 2021



(Mathori Hasan Maraghi)

HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahim

Puji syukur saya ucapkan kepada Allah SWT yang telah memberikan rahmat dan karuniaNya sehingga saya dapat menyelesaikan laporan Tugas Akhir. Tugas Akhir ini saya persembahkan kepada orang tua, kedua adik saya, keluarga, sahabat, teman dekat, dan seluruh pihak yang mendukung selama proses pengerjaannya.

HALAMAN MOTO

"*Man jadda wajada – Barang siapa yang bersungguh-sunguh maka ia akan berhasil"*

"Lakukan segala aktivitasmu dengan niat beribadah kepada Allah SWT"

KATA PENGANTAR

Puji syukur penulis ucapkan kepada Allah SWT yang telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul "PENGEMBANGAN *LEARNING MANAGEMENT SYSTEM* RUMAH/PONDOK TAHFIZH DENGAN *SERVICE ORIENTED ARCHITECTURE*" Tugas Akhir ini, disusun penulis untuk memenuhi salah satu syarat kelulusan jenjang sarjana Program Studi Informatika di Universitas Islam Indonesia.

Dalam menyusun Tugas Akhir ini, penulis mendapat banyak motivasi, dukungan, dan bimbingan dari berbagai pihak. Oleh karena itu, dalam kesempatan ini penulis mengucapkan terima kasih Jazakumullaahu Khairan (Semoga Allah membalas kalian dengan kebaikan) kepada:

1. Keluarga penulis, Bapak Nasrudin, Ibu Sukiyani, Adik Sarah Arum Bamega dan Rumaisha Hanun Falihah yang telah memberikan doa, semangat, dan bantuan moril maupun materi yang tak terhingga.
2. Bapak Prof. Fathul Wahid, S.T., M.Sc., Ph.D. selaku Rektor Universitas Islam Indonesia.
3. Bapak Hari Purnomo, Prof., Dr., Ir., M.T., sebagai Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
4. Bapak Hendrik, S.T., M.Eng. selaku Ketua Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku Kaprodi Informatika Universitas Islam Indonesia.
6. Bapak Andhik Budi Cahyono, S.T., M.T. selaku Dosen Pembimbing yang telah membantu memberikan bimbingan, kritik, petunjuk dan dukungan kepada penulis dalam pembuatan Tugas Akhir ini.
7. Bapak Rahadian Kurniawan, S.Kom., M.Kom. selaku Dosen Pembimbing Akademik yang memberikan dukungan dan arahan di masa perkuliahan di Universitas Islam Indonesia.
8. Seluruh dosen Informatika UII yang telah memberikan ilmu bermanfaat, mengajarkan banyak hal selama masa studi, semoga Allah SWT senantiasa melindungi dan membalas jasa, serta kebaikan Bapak/Ibu sekalian.

9. Semua rekan-rekan Informatika UII angkatan 2014, teman seperjuangan yang menemani hari-hari penulis selama masa studi.
10. Seluruh rekan-rekan Pondok Pesantren UII angkatan 2014, yang senantiasa menemani dan mendukung penulis selama masa studi.
11. Semua pihak yang telah membantu dan mendoakan penulis yang tidak dapat disebutkan satu per satu.

Segala arahan, motivasi dan bantuan yang telah diberikan kepada penulis semoga menjadi amal ibadah bagi keluarga, Ibu dan rekan-rekan mendapat balasan yang lebih baik lagi dari Allah SWT. Penulis sangat menyadari bahwa Tugas Akhir ini masih jauh dari kata sempurna.

Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan kepadapembaca. Penulis berharap semoga Tugas Akhir ini bermanfaat bagi para pembaca serta dapat menjadi bahan sumbangan pemikiran dan perkembangan Informatika.

Yogyakarta, 29 November 2021



(Mathori Hasan Maraghi)

SARI

Sistem pendidikan konvensional seringkali menjadi kendala dalam proses pendidikan karena dipengaruhi oleh jarak dan waktu. Hal ini pula yang terjadi di rumah tahfizh, penyampaian materi dan tugas sebagai penunjang kegiatan pembelajaran masih belum menggunakan aplikasi pembelajaran khusus. Maka perlu dikembangkan aplikasi *Learning Management System*. Pengembangan *Learning Management System* ada banyak metode arsitektur aplikasi yang dapat digunakan, salah satunya adalah *Service Oriented Architecture*. pendekatan yang menjadikan aplikasi berfungsi sebagai layanan. SOA juga menawarkan kemudahan integrasi dengan beberapa sistem yang berbeda.

Tujuan dari penelitian ini adalah bagaimana membangun sebuah rancangan untuk membangun sebuah sistem informasi yang dapat digunakan sebagai sarana untuk kegiatan pembelajaran antara siswa dan guru/ustadz di rumah tahfizh pada saat penerapan kebijakan di masa pandemi. Dalam penelitian ini metode yang digunakan dalam membangun sistem adalah dengan metode arsitektur berorientasi layanan (SOA) dan metode *waterfall* sebagai metode pengembangan. Desain menggunakan Unified Modelling Language(UML). Sedangkan bahasa pemrograman yang digunakan adalah PHP dengan Laravel sebagai framework dan MySQL sebagai databasenya. Teknik pengujian sistem dilakukan dengan pendekatan *blackbox testing*. Hasil penelitian berupa aplikasi *Learning Management System* (LMS) berbasis web dengan menggunakan metode *Service Oriented Architecture* (SOA).

Kata kunci: *Learning Management System, SOA, Blackbox Testing, UML, Waterfall*

GLOSARIUM

Application Programming Interface	Sebuah aplikasi yang mengintegrasikan antara aplikasi yang kita buat dengan aplikasi yang lain.
Black Box	Metode pengujian sistem untuk mengetahui kualitas fungsional sistem.
Laravel Framework	Pengembangan website berbasis MVC (*Model View Controller*) dalam versi PHP.
Service Oriented Architecture	Arsitektur pengembangan aplikasi berdasarkan layanan API.
Waterfall	Metode pengembangan perangkat lunak.

DAFTAR ISI

HALAMAN JUDUL ..... **i**

HALAMAN PENGESAHAN DOSEN PEMBIMBING ..... **ii**

HALAMAN PENGESAHAN DOSEN PENGUJI ..... **iii**

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR ..... **iv**

HALAMAN PERSEMBAHAN ..... **v**

HALAMAN MOTO ..... **vi**

KATA PENGANTAR ..... **vii**

SARI ..... **ix**

GLOSARIUM ..... **x**

DAFTAR ISI ..... **xi**

DAFTAR TABEL ..... **xiii**

DAFTAR GAMBAR ..... **xiv**

BAB I PENDAHULUAN **1**

1.1 Latar Belakang 1

1.2 Rumusan Masalah 2

1.3 Batasan Masalah 2

1.4 Tujuan Penelitian 3

1.5 Manfaat Penelitian 3

1.6 Metodologi Penelitian 3

1.7 Sistematika Penulisan 4

BAB II LANDASAN TEORI **5**

2.1 Tinjauan Pustaka 5

2.2 Rumah Tahfiz 5

2.3 Learning Management System (LMS) 7

2.4 Service Oriented Architecture (SOA) 10

2.5 PHP 11

2.6 MySQL 11

2.7 Unified Modeling Language (UML) 11
2.8 *Waterfall* 14
2.9 Black-Box Testing 16

BAB III METODOLOGI PENELITIAN 17
3.1 Metode Pengumpulan Data 17
3.2 Analisis Masalah 19
3.3 Communicating (Komunikasi) 19
3.4 *Planning* (Perencanaan) 20
3.5 *Modeling* (Pemodelan) 20
3.6 Construction (Konstruksi) 41
3.7 *Deployment* (Penerapan) 41

BAB IV HASIL DAN PEMBAHASAN 42
4.1 Implementasi Sistem 42
4.2 Pengujian 55

BAB V PENUTUP 65
5.1 Kesimpulan 65
5.2 Saran 65

DAFTAR PUSTAKA 66

DAFTAR TABEL

Tabel 2. 1 Simbol *Use Case Diagram* (Sumber : (Marbun & Sinaga, 2019))....12
Tabel 2. 2 Simbol *Activity Diagram* (Sumber : (Marbun & Sinaga, 2019))....13
Tabel 2. 3 *Deployment Diagram*....14
Tabel 3. 1 Use Case 22
Tabel 3. 2 Tabel Admin 37
Tabel 3. 3 Tabel Guru/Ustadz....37
Tabel 3. 4 Tabel Kelas 38
Tabel 3. 5 Tabel Kelas Materi....38
Tabel 3. 6 Tabel Kelas Member....38
Tabel 3. 7 Kelas Tugas....39
Tabel 3. 8 Tabel Kelas Tugas Santri....39
Tabel 3. 9 Tabel Santri....39
Tabel 3. 10 Tabel Users 40
Tabel 4.1 Rencana Pengujian Admin....56
Tabel 4.2 Rencana pengujian Guru/Ustadz 56
Tabel 4.3 Rencana Pengujian Santri 57
Tabel 4.4 Pengujian *Blackbox Login* 57
Tabel 4.5 Pengujian *Blackbox* Halaman Guru....58
Tabel 4.6 Pengujian *Blackbox* Halaman Santri....59
Tabel 4.7 Pengujian *Blackbox* Halaman kelas....59
Tabel 4.8 Pengujian *Blackbox Login* Guru/Ustadz....60
Tabel 4.9 Pengujian *Blackbox* Halaman daftar kelas....60
Tabel 4.10 Pengujian *Blackbox* Halaman daftar materi....61
Tabel 4.11 Pengujian *Blackbox* Halaman daftar tugas 62
Tabel 4.12 Pengujian *Blackbox* Halaman daftar penilaian tugas....62
Tabel 4.13 Pengujian *Blackbox Login* Santri....63
Tabel 4.14 Pengujian *Blackbox* Halaman daftar kelas santri....63
Tabel 4.15 Pengujian *Blackbox* Halaman daftar tugas 64

DAFTAR GAMBAR

Gambar 2. 1 *Waterfall Model* 15
Gambar 3. 1 Metodologi Penelitian 17
Gambar 3. 2 *Use Case Diagram* 21
Gambar 3. 3 *Activity Diagram Login* 23
Gambar 3. 4 *Activity Diagram* Mengelola Data Santri 24
Gambar 3. 5 *Activity Diagram* Mengelola Data Guru 25
Gambar 3. 6 *Activity* Diagram Mengelola Data Kelas 26
Gambar 3. 7 *Activity* Diagram Mengelola Data Kelas Santri 27
Gambar 3. 8 *Activity* Diagram Login 28
Gambar 3. 9 *Activity* Diagram Mengelola Data Profil 29
Gambar 3. 10 *Activity* Diagram Mengelola Data Materi 30
Gambar 3. 11 *Activity* Diagram Mengelola Data Tugas 31
Gambar 3. 12 Activity Diagram Mengelola Data Nilai Santri 32
Gambar 3. 13 *Activity* Diagram Login 33
Gambar 3. 14 *Activity* Diagram Melihat Data Kelas Santri 33
Gambar 3. 15 *Activity* Diagram Melihat Data Profil 34
Gambar 3. 16 *Activity* Diagram Melihat Data Materi 34
Gambar 3. 17 *Activity* Diagram Melihat dan Upload Tugas 35
Gambar 3. 18 *Activity* Diagram Melihat Nilai 35
Gambar 3. 19 *Deployment* Diagram 36
Gambar 3. 20 Arsitektur SOA Rumah Tahfizh 40
Gambar 4.1 Halaman Login 42
Gambar 4.2 Halaman Dashboard Admin 43
Gambar 4.3 Halaman Guru 43
Gambar 4.4 Halaman Tambah Guru 43
Gambar 4.5 Halaman Edit Guru 44
Gambar 4.6 Halaman Hapus data Guru 44
Gambar 4.7 Halaman Santri 44
Gambar 4.8 Halaman Tambah Santri 45
Gambar 4.9 Halaman Edit Santri 45
Gambar 4.10 Halaman Hapus Santri 45

Gambar 4.11 Halaman Kelas 46
Gambar 4.12 Halaman Kelola Kelas 46
Gambar 4.13 Halaman Tambah Kelas 46
Gambar 4.14 Halaman Edit Kelas 47
Gambar 4.15 Halaman Hapus Data Kelas 47
Gambar 4.16 Halaman Dashboard Guru 48
Gambar 4.17 Halaman Fitur Daftar Kelas 48
Gambar 4.18 Halaman Daftar Kelas 48
Gambar 4.19 Halaman Tambah Materi 49
Gambar 4.20 Halaman Tambah Tugas 49
Gambar 4.21 Halaman Daftar Materi 49
Gambar 4.22 Halaman Detail Materi 50
Gambar 4.23 Halaman Edit Materi 50
Gambar 4.24 Halaman Hapus Materi 50
Gambar 4.25 Halaman Daftar Tugas 51
Gambar 4.26 Halaman Detail Tugas 51
Gambar 4.27 Halaman Edit Tugas 51
Gambar 4.28 Halaman Hapus Tugas 52
Gambar 4.29 Halaman Daftar Penilaian Tugas 52
Gambar 4.30 Halaman Penilaian 52
Gambar 4.31 Halaman Dashboard Santri 53
Gambar 4.32 Halaman Fitur Daftar Kelas 53
Gambar 4.33 Halaman Deskripsi Kelas 54
Gambar 4.34 Halaman Daftar Materi 54
Gambar 4.35 Halaman Daftar Tugas 54
Gambar 4.36 Halaman Deskripsi Tugas 55
Gambar 4.37 Halaman Upload Tugas 55

BAB I
PENDAHULUAN

1.1 Latar Belakang

Cepatnya kemajuan informasi nyatanya bisa memicu konsekuensi di beragam aspek kesibukan masyarakat. Sehingga mau tidak mau berbagai hal pun dipaksa untuk mengikuti perkembangan zaman yang serba canggih ini, salah satunya ialah dalam dunia pendidikan. Mau tidak mau, dunia Pendidikan harus mengikuti perkembangan teknologi dengan menerapkan sistem informasi ke dalam pusat pendidikan guna menunjang keberhasilan dunia pendidikan. Perkembangan teknologi yang sangat pesat saat ini, membuat terobosan baru dalam pembelajaran terus bermunculan (Yauma, Fitri, & Ningsih, 2021). Dalam bidang pendidikan, teknologi dapat mempermudah kegiatan pembelajaran, terbukti saat ini sebagian pembelajaran sudah beralih dengan metode *online* yang tak terbatas dari sebelumnya yang hanya dilakukan di ruang kelas. Sehingga proses belajar peserta didik dapat dilakukan secara mandiri, baik tanpa bantuan atau bimbingan dari gurunya di kelas ataupun di luar sekolah (Fitriyah & Ghofur, 2021). Namun, masih ada juga beberapa pusat Pendidikan yang belum menerapkan sistem informasi ke dalam proses pembelajaran yang dilakukan dan masih menggunakan metode pendidikan secara langsung atau konvensional sehingga mengakibatkan kegiatan pembelajaran dirasa kurang efektif dan efisien.

Learning Management System (LMS) merupakan salah satu perkembangan teknologi yang dapat menjadi alternatif media pembelajaran dizaman yang serba canggih ini. LMS salah satu aplikasi yang dipakai guna kepentingan administrasi, penyuratan, mencari materi, serta informasi aktivitas, pemberian materi, dan pelatihan kegiatan belajar mengajar secara online yang terhubung melalui internet atau dengan kata lain dilakukan secara online. LMS menyediakan berbagai layanan untuk mempermudah *upload* dan *share* material pengajaran, diskusi online, chatting, penyelenggaraan kuis, survei, laporan, dan lain sebagainya (Yauma, Fitri, & Ningsih, 2021).

Rumah Tahfidz salah satu institusi dakwah yang bertujuan memahami serta menghafalkan kitab suci Qur'an. Penanaman nilai agama menjadi prioritas utama dalam pembelajaran. Materi agama disampaikan secara struktur dan sistematis mengikuti dengan kebutuhan santri sesuai dengan tahapan usianya. Namun semenjak ditetapkannya kebijakan oleh pemerintah dimasa pandemi, banyak Rumah Tahfidz masih melakukan sistem penyampaian informasi pendidikan secara konvensional. Hal tersebut dikarenakan Rumah

Tahfidz masih belum menerapkan sistem informasi dalam proses kegiatan pembelajaran yang berlangsung hingga saat ini. Dengan menerapkan protokol kesehatan dan pengurangan jam belajar mengajar santri/santriwati sering kali tidak mendapatkan pemahamaan dari materi yang disampaikan oleh pengajar. Beberapa masalah lainnya yaitu membuat para pengajar terkadang tidak sempat memberikan materi untuk santri/santriwati yang kemudian materi agama tersebut diberikan di pertemuan selanjutnya. Selain itu, dalam kegiatan membaca dan menghafal Al-Quran menjadi terhambat mengingat tidak adanya sarana yang dapat digunakan dalam kegiatan belajar mengajar antara santri/santriwati dengan pengajar. Hal ini tentunya sangat tidak efektif dan efisien mengingat banyak Rumah Tahfidz belum memanfaatkan teknologi dan informasi di zaman yang sudah serba canggih saat ini.

Berdasarkan permasalahan yang telah diuraikan diatas, peneliti bertujuan untuk membuat *Learning Management System* (LMS) berbasis web agar santri/santriwati dapat dengan mudah mengikuti pembelajaran dimanapun dan kapanpun yang tentunya dapat mempermudah proses pembelajaran dengan menaati protokol yang berlaku pada masa pandemi ini. Sistem yang dibuat menggunakan PHP sebagai bahasa pemrograman, MySQL sebagai database manajemen, dibangun dengan menggunakan metode *Service Oriented Architecture* (SOA) dan metode waterfall sebagai metode pengembangan. Daripada itu, program yang sudah dibuat dites manfaatnya memakai teknik uji yaknni *Black-box Testing*.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang telah diuraikan pada latar belakang diatas, maka penulis merumuskan permasalahan sebagai berikut:

a. Bagaimana membangun sebuah sistem informasi yang dapat digunakan sebagai sarana untuk kegiatan pembelajaran antara para santri/santriwati dengan pengajar selama diterapkannya kebijakan dimasa pandemi?
b. Bagaimana membangun sebuah sistem informasi berupa *Learning Management System* (LMS) berbasis *web* menggunakan metode *Service Oriented Architecture* (SOA) dan metode pengembangan Waterfall?

1.3 Batasan Masalah

Agar penelitian dapat berfokus dan berjalan secara terarah serta tidak menyimpang dari pokok bahasan yang dilakukan dalam penelitian ini, maka penulis membuat beberapa batasan masalah. Beberapa batasan masalah tersebut diantaranya ialah:

a. *Learning Management System* (LMS) yang dirancang bangun berbasis *web* dan hanya dapat dijalankan sementara dalam *localhost.*
b. *Learning Management System* (LMS) dirancang bangun dengan menggunakan metode *Service Oriented Architecture* (SOA) metode pengembangan *Waterfall.*
c. Hanya admin yang dapat menambah, mengubah dan menghapus guru, kelas dan santri.
d. Tiap kelas atau materi pelajaran hanya dapat diajar oleh satu ustadz.

1.4 Tujuan Penelitian

Adapun tujuan dengan dilakukannya penelitian ini oleh penulis diantaranya ialah:

a. Merancang bangun sebuah sistem informasi yang dapat digunakan sebagai sarana untuk kegiatan pembelajaran antara para santri/santriwati dengan pengajar selama diterapkannya kebijakan dimasa pandemi.
b. Merancang bangun sebuah sistem informasi berupa *Learning Management System* (LMS) berbasis *web* dengan menggunakan metode *Service Oriented Architecture* (SOA) dan metode pengembangan Waterfall.

1.5 Manfaat Penelitian

Adapun manfaat yang diperoleh melalui penelitian yang dilakukan oleh penulis penulis diantaranya ialah:

a. Bagi peneliti, untuk memenuhi syarat program strata satu (S1) Sistem Informasi.
b. Bagi pengguna, untuk mengatasi masalah yang dihadapi antara para santri/santriwati dengan pengajar pada sistem pembelajaran yang berlangsung selama ditetapkannya kebijakan pemerintah dimasa pandemi, dimana nantinya sistem tersebut dapat dijadikan sebagai sarana kegiatan pembelajaran yang dapat diakses dimanapun dan kapanpun secara efektif dan efisien
c. Bagi akademik, dapat menjadi referensi bagi penulis lainnya pada penelitian dimasa mendatang.

1.6 Metodologi Penelitian

1.6.1 Metode Pengumpulan Data

Ketika mengumpulkan informasi, penulis memakai metode studi pustaka, wawancara dan observasi.

1.6.2 Metode Pengembangan Sistem

Metode yang digunakan penulis dalam perancangan aplikasi ini adalah metode waterfall karena metode ini merupakan pendekatan terstruktur. Berikut tahapan dalam metode waterfall:

a. Analisa kebutuhan, yaitu membuat analisis aliran kerja manajemen yang berjalan dan melakukan perincian mengenai apa saja yang dibutuhkan dalam pengembangan sistem dan membuat perencanaan yang berkaitan dengan proyek sistem.
b. Merancang program, yakni membangun skema alur kegiatan pengelolaan serta skema perancangan yang dibutuhkan sepanjang peningkatan system informasi.
c. Menulis pengkodean informasi, yakni langkah peningkatan sistem informasi melalui menyusun aplikasi yang dibutuhkan.
d. *Testing* aplikasi, yakni menjalankan pengujian program yang sudah dikerjakan.
e. Pelaksanaan dan memelihara sistem, yaitu mempraktikan serta merawat program yang sudah dibuat.

1.7 Sistematika Penulisan

Dalam penyusunan skripsi ini, pembahasan disajikan dalam lima bab pokok bahasan yang secara singkat akan diuraikan sebagai berikut:

BAB I Pendahuluan

Di Bab ini menerangkan mengenai latar belakang, perumusan masalah, batasan permasalahan, tujuan serta manfaat penelitian, juga penataan penyusunan tulisan.

BAB II Landasan Teori

Pada bab ini dibahas teori-teori yang berkaiatan dengan *framework* Laravel dan *Learning Management System* (LMS) untuk mendukung pembahasan bab selanjutnya.

BAB III Metodologi Penelitian

Pada bab ini menjelaskan tentang langkah-langkah dari metode yang digunakan, baik metode pengumpulan data maupun metode pengembangan sistem.

BAB IV Hasil dan Pembahasan

Pada sesi bab ini mengutarakan perihal rencana pembangunan program juga pelaksanaan implementasinya.

BAB V Penutup

Bab ini berisi kesimpulan dari pembahasan masalah serta saran yang berkenaan dengan penelitian ini.

BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian yang dilakukan oleh (Fitriani, 2020), membahas terkait penerapan LMS ditengah pandemi Covid-19 pada pendidikan tinggi. Pandemi COVID-19 mendorong sektor Pendidikan untuk melakukan perubahan pada model proses pembelajaran dimana proses pembelajaran berpindah dari pembelajaran konvensional atau tatap muka di kelas menjadi pembelajaran online atau daring. Mengenai itu tenaga pendidik yakni dosen serta mahasiswa diharuskan untuk menyesuaikan dalam menempuh inovasi serta konfrontasi modern mengenai pergantian rencana ketika pengkajian yang dipraktikan akademi. *Learning Management System* (LMS) adalah salah satu cara yang digunakan oleh perguruan tinggi untuk mendukung dan mengelola pembelajaran secara online agar proses belajar mengajar tetap berjalan dengan baik selama pandemic COVID-19. kesimpulan dari observasi setelah dijalankan bisa diterangkan bahwasanya program *Learning Management System* seperti Google Classroom, Edmodo, Moodle, serta EVIMA Edlink dan LMS yang dikembangkan oleh perguruan tinggi maupun yang disediakan oleh pemerintah sangat bermanfaat sebagai media pembelajaran online atau daring selama pandemic COVID-19. Dari penelitian yang dilakukan, peneliti mengharapkan pandemi COVID-19 dapat berakhir sehingga sistem pembelajaran bisa dilakukan secara tatap muka selain melalui online dan sistem pendidikan di Indonesia tetap maju dan menghasilkan generasi muda yang berkualitas.

Observasi awal yang dijalankan (Istiyan & Nyoto, 2020) menerangkan tentang *Learning Management System* di tingkat MA bermaksud supaya tahu pengembangan menenai pengkajian anak didik memakai teknik pengkajian yang disempurnakan. Hasil dari penelitian ini menunjukkan bahwa metode pembelajaran yang dikembangkan sangat layak berdasarkan hasil pengujian UAT oleh ahli materi dengan skor 84,95% dan siswa dengan skor 84,29%.

2.2 Rumah Tahfiz

Rumah Tahfiz merupakan satuan pendidikan keagamaan Islam nonformal, dimana Rumah Tahfiz pada dasarnya merupakan lembaga keagamaan yang sangat mengakar dikalangan masyarakat. Lembaga tersebut dijadikan sebagai sebagai salah satu media pengembangan dalam membaca Al-Qur’an melalui pemberantasan buta huruf Al-Qur’an.

Kegiatan pengkajian yang berjalan di pondok Tahfiz yakni adalah mencari ilmu, menghafalkan, mematuhi, serta memasyarakatkan kaidah-kaidah Al-Qur'an di aktivitas setiap hari berpedomankan kediaman,wilayah, serta masyarakat. Dalam hal penyampaian materi, Rumah Tahfiz menyampaikan materi dengan menggunakan metode atau teknik kekhasan masing-masing satuan pendidikan. Selain materi inti, terdapat 2 materi penunjang sebagai pengembagan dan kemandirian. Materi-materi tersebut diantaranya ialah materi pembelajaran, dimana materi tersebut dapat bermuatan pelajaran aqidah akhlak, praktek ibadah, sejarah islam, doa harian, muatan lokal dan lainnya menyesuaikan dengan kebutuhan serta kemampuan lembaga. Yang kedua, materi pembelajaran sebagaimana dimaksud disesuaikan dengan satuan pendidikan, kompetensi peserta didik serta kearifan lokal (Mahardhika, 2013).

2.2.1 Model Pendidikan

Sebagai satuan pendidikan al-Quran, Rumah Tahfizh merupakan lembaga yang mengkhususkan untuk menghafal al-Qur'an, mengamalkannya, dan membudayakan nilai-nilainya dalam kehidupan sehari-hari yang berbasis hunian, lingkungan dan komunitas. Berdasarkan SK Direktorat Jenderal Pendidikan Islam nomor 91 tahun 2020, materi pembelajaran rumah tahfizh terdiri dari pembelajaran inti yaitu membaca, menghafal dan memahami al-Qur'an serta pembelajaran penunjang yang menyesuaikan dengn kebutuhan santri dan kemampuan lembaga. Materi pembelajaran penunjang dapat berupa pelajaran agama seperti ulumul qur'an, hadits, tafsir, fiqih, aqidah akhlak, bahasa arab, sejarah islam, pengkajian kitab kuning dan lain sebagainya. Materi pembelajaran tersebut disesuaikan dengan jenjang pendidikan, kompetensi santri serta kearifan lokal.

Masa pendidikan rumah tahfizh sangat beragam, biasanya disesuaikan dengan program pendidikan atau target pencapaiannya. Apabila target pencapaiannya adalah hafal 30 juz dalam 2 tahun, maka santri bisa dinyatakan lulus dalam 2 tahun. Sebelum dinyatakan lulus, santri harus menyelesaikan setoran hafalannya ataupun materi pembelajaran penunjang lainnya dan menempuh ujian kelulusan.

2.2.2 Proses Pembelajaran

Pada dasarnya, proses pembelajaran dalam rumah tahfizh disampaikan dengan menggunakan kekhasan dan metode masing-masing satuan dan jenjang pendidikan. Dalam menghafal al-Qur'an misalnya, terdiri dari beberapa halaqoh yang diampu oleh seorang

ustadz yang siap menerima setoran hafalan. Selain itu, ustadz yang mengajar juga dapat menyampaikan secara *talaqqi* terlebih dahulu ayat-ayat yang akan dihafal agar bacaan santri benar dan tepat tajwidnya. Santri diwajibkan untuk menyetorkan hafalannya setiap hari dan mengulangnya (*muraja'ah)* secara mandiri maupun berpasangan dengan santri lainnya. Selain itu, sebagai penguatan hafalan, santri juga harus mengikuti *tasmi'* membacakan hafalannya secara bergantian dihadapan seluruh peserta halaqoh.

Materi penunjang disampaikan dengan 2 metode, pengajian kitab mingguan setelah shalat wajib dan pembelajaran dalam kelas secara terjadwal setiap harinya di luar waktu menghafal. Santri mendapatkan materi sesuai dengan jenjang ataupun angkatannya. Setiap kelas dan materi diajarkan oleh 1 ustadz dan setiap ustadz dapat mengajar beberapa kelas. Selain mengajarkan materi pembelajaran, ustadz sebagai pengajar dapat memberikan kuis mingguan atau bulanan sebagai bahan evaluasi pemahaman dan kemampuan santri.

Pada akhir periode, selain ujian hafalan, diadakan ujian akhir untuk menentukan keberhasilan santri dalam menguasai materi yang telah dipelajarinya. Santri dinyatakan lulus jika telah menyelesaikan target hafalannya dan lulus dalam ujian evaluasi akhir.

2.3 Learning Management System (LMS)

2.3.1 Pengertian Learning Management System (LMS)

Salah satu program yang dipakai menjalankan pengkajian melalui daring yang termasuk beragam perspektif yakni pelajaran, penghijrahan, manajemen, serta pemeringkatan. berjalannya kemajuan informasi, orang-orang di seluruh penjuru negara mulai terhubung di dunia maya. Keperluan mengenai data yang bisa didapat secara cepat sudah dijumpai seperti mana keperluan absolut serta jangkauan tempat bukanlah sebuah permasalahan. Dari sinilah muncul LMS. berkembangnya LMS semakin cepat membuahkan ide-ide baru bertujuan demi menguasai problem interoperability menyertai LMS satu serta yang lain dengan normal. gambaran normal yang tampil misal normal yang diterbitkan sama AICC (*Airline Industry* CBT *Commettee*), IMS, IEEE LOM, serta ARIADNE (Fitriani, 2020).

2.3.2 Fungsi Learning Management System (LMS)

Menurut (Fitriani, 2020) berikut adalah fungsi dasar LMS:

1. Katalog

   LMS diharuskan bisa memberikan objek training yang dipunyai. Objek-objek bisa berwujud ilmu computer based training, makalah, disertasi, kesimpulan konferens

DLL. catatan yang bagus diharuskan mampu memilah objek didasarkan beragam objek,divisi yang membutuhkan, program studi (contohnya, program studi perhitungan untuk program studi finansial, pengetahuan *coaching* untuk program studi keahlian, serta pengetahuan bahasa Inggris menurut program studi koneksi).

Direktori yang bagus diwajibkan bisa menunjukan data mengenai sebuah pengetahuan secara kompleks, melingkupi titel, objek, jangkauan/konfigurasi, waktu, sasaran anak didik, bulan siap,entitas kata pengantar, ujian yang wajib dijalankan dll.

2. Registrasi dan persetujuan

   Kegunaan ini mungkin saja salah satu calon kandidat training mendata dirinya melalui daring, untuk pengetahuan daring ataupun datang ke kelas. data yang ada di direktori wajib ada disaat kandidat training akan mendata dirinya. kandidat partisipan diwajibkan bisa menentukan, jika saja terdapat opsi jam, lingkungan, anggaran dll.

   Jika kandidat partisipan training sudah mendata serta pelajaran memerlukan penerimaan pimpinan atau manusia yang berbeda, hingga LMS wajib memberitahu pimpinan bahwasanya ada satu registrasi yang wajib langsung diperhatikan. pimpinan bisa membuka aplikasi lalu memperhatikan registrasi juga data mengenai anak didik serta pengetahuan. lalu, ia bisa bertindak, sekiranya setuju atau tidak. LMS lalu memberitahukan hal itu kepada kandidat partisipan training.

   LMS yang bagus bisa juga merapihkan informasi registrasi serta setuju atau tidaknya supaya dapat meringankan divisi training ketika memantau aktivitas e-learning di hari yang akan datang.

3. Menjalankan dan memonitor e-learning

   LMS harus menjalankan materi pelajaran e-learning dengan baik. Jikalau CPU anak didik tak mempunyai susunan yang sinkron,lalu LMS wajib memberi warning terhadap anak didik. Sehabis itu ilmu pembelajaran daring dilakukan, LMS wajib memiliki skill merecord aktivitas supaya bisa dibuatkan penjelasannya. LMS wajib bisa merecord mengenai seberapa waktu kandidat training diakses pengetahuan training pembelajaran, lebih dari satu kali, bulan serta waktunya dll.

4. Evaluasi

   LMS diwajibkan bisa menjalankan beragam penilaian yang bisa diukur skill kandidat training awal dan akhir sehabis menjalankan training. penilaian wajib bisa

diukur betapa jarak akhir kandidat training mengambil pengetahuan. Didasarkan dari kesimpulan penilaian, LMS dengan spontan mengusulkan agar diulang lagi materinya, dibaca lebih dari satu makalah ditambahkan, diikuti materi yang berbeda atau aksi lain.

5. Komunikasi

   LMS berfungsi menjadi alat hubungan untuk divisi training serta member institusi. LMS bisa memberikan dan memberitahu informasi untuk para anak didik yang dipilih. Pengumuman dapat dari pengajar atau administrator pelatihan. hubungan ini bisa berarti guru mengasih pengetahuan bacaan lagi terhadap kandidat training lewat sistem. guru bisa juga meng-*upload* suatu makalah yang dituju untuk beragam anak didik terpilih dan LMS bisa memberitahukannya ke anak didik supaya bisa diakses dan di *download* makalahnya lewat LMS.

6. Laporan

   Lewat LMS, admin trainer bisa mendapat penilaian yang terdapat informasi trainer. pimpinan serta pengelola wajib bisa mengakses sistem serta mengeprint penilaian dengan metode tatap muka tanpa perlu pertolongan admin. contohnya, si eksekutif mau tahu jumlah bawahannya yang sudah mendapat training MS PowerPoint. dia gampang login ke situs LMS serta mengetahui informasi trainer bawahannya.

7. Rencana pelatihan

   Suatu eksekutif bisa membikin jadwal trainer teruntuk sebagian bawahannya tentang penjabaran keperluan pelatihan. bagusnya, apabila jadwal training atau rencana pelatihan serta pengembangan rencana bisa dimasukan pada LMS.

   Jadi, berdasarkan rencana pelatihan, LMS secara otomatis merekomendasikan program pelatihan yang sesuai dan mengatur jadwalnya. Jadi, karyawan dapat melihat pelatihan yang dia butuhkan melalui LMS, waktu untuk melakukan dan selesai tepat waktu.

8. Integrasi

   Dalam suatu organisasi, ada beberapa sistem komputer. Misalnya, bagian SDM memiliki sistem personalia (PeopleSoft, SAP, dan lain-lain) dan bagian keuangan memiliki sistem akuntansi (Accpac, MyOB, dan lain-lain). LMS yang dapat menyampailan komunikasi dan integrasi dengan sistem yang ada adalah LSM yang baik.

Integrasi dengan sistem SDM adalah hal paling vital. Dengan adanya integrasi yang tersusun, LMS dapat mendaftar memperbaharui informasi karyawan dari sistem SDM yang sudah ada. Jadi, bila pada sistem SDM terdapat adanya perubahan data karyawan, penambahan karyawan baru atau karyawan yang keluar, maka informasi dapat digunakan untuk memperbarui data pelatihan terkait.

Selain kedelapan fungsi dasar, dapat ditambahkan fungsi-fungsi penunjang lain, misalnya forum diskusi atau chatroom agar pelajar dan pengajar dapat berdiskusi mengenai topik pelatihan, dan lain-lain.

2.4 Service Oriented Architecture (SOA)

Service Oriented Architecture (SOA) merupakan sebuah model arsitektural yang biasa digunakan untuk membangun suatu aplikasi software dengan memanfaatkan layanan yang tersedia dijaringan seperti *Web*. Pada dasarnya, *Service Oriented Architecture* (SOA) merupakan konsep pokok yang mendasari implementasi *Web Service*. *Service Oriented Architecture* (SOA) mampu meningkatkan *loosely-coupling* antar komponen-komponen perangkat lunak agar dapat digunakan kembali. Dengan kata lain, masing-masing pihak tidak perlu punya ketergantungan yang tinggi satu sama lain. Komponen utama pada *Service Oriented Architecture* (SOA) adalah *service* (layanan), dimana *Service Oriented Architecture* (SOA) memiliki 3 komponen yaitu *service provider*, *service registry*, *service requester*. Selain itu, *Service Oriented Architecture* (SOA) juga memiliki 3 operasi yaitu *publish*, *find*, dan *bind*. *Service Oriented Architecture* (SOA) menggabungkan peran dari semua komponen dan semua operasi tersebut dengan tujuan untuk mempertahankan penemuan dan penggunaan layanan otomatis. Proses yang terjadi ialah *Service Oriented Architecture* (SOA) menggunakan prinsip *client-server*, dimana *Service Provider* menerbitkan layanan ke *registry* sehingga layanan dapat tersedia di Internet untuk melayani permintaan dari konsumen. *Service Requester* (*client*) berfungsi untuk manampilkan semua operasi pada penemuan layanan di *registry* bertujuan untuk menemukan layanan yang dibutuhkan, kemudian mengakses layanan tersebut. *Service Registry* membantu *Service Providers* menemukan *Service Requester*, begitu juga sebaliknya dengan berperan sebagai *registry* dari layanan (Khoirudin, Daru, & Nugroho, 2019).

2.5 PHP

Menurut dokumen resmi PHP, PHP merupakan singkatan dari PHP *Hypertext Preprocessor*. PHP merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Hasil proses tersebut lah yang dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, Anda bisa menampilkan isi *database* ke halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip – skrip seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara *Command Line*. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web* server maupun *browser* (Ruseno, 2019).

2.6 MySQL

MySQL adalah singkatan dari Structured Query Language dan merupakan bahasa yang kami gunakan untuk membuat dan mengelola database dan isinya. MySQL dapat digunakan untuk menambah, mengubah dan menghapus data dalam database. MySQL adalah sistem manajemen database dengan struktur relasional. Artinya data akan disimpan di beberapa bagian database yang terpisah sehingga manipulasi data akan jauh lebih cepat. MySQL dapat mengelola database dengan berbagai ukuran. MySQL juga dapat menjalankan perintah Structured Query Language untuk mengelola database di dalamnya. Hingga kini MySQL sudah berkembang hingga versi 5. MySQL 5 sudah mendukung trigger untuk memudahkan pengelolaan tabel dalam database (Wiliani & Zambi, 2017).

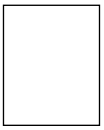
2.7 Unified Modeling Language (UML)

UML merupakan sebuah standar Bahasa yang digunakan untuk menganalisis dan merancang serta menggambarkan arsitektur program dalam pemrograman *object oriented*. UML adalah pengganti metode analisis dan desain berorientasi objek yang diperkenalkan sekitar akhir 1980-an dan awal 1990-an. UML merupakan gabungan dari metode *Booch*, *Rumbaugh* (OMT) dan *Jacobson*. Tetapi UML mencakup lebih luas dari pada OOAD. Pada pertengahan saat pengembangan UML, dilakukan standarisasi proses dengan OMG (*Object Management Group*) dengan harapan UML bakal menjadi bahasa standar pemodelan pada masa yang akan dating (yang sekarang sudah banyak dipakai oleh berbagai kalangan) (Kurniawan, Apriliah, Kurniawan, & Firmansyah, 2020).

2.7.1 *Use Case* Diagram

Use case diagram adalah rangkaian atau gambaran dari suatu kelompok yang saling berhubungan dan membentuk suatu sistem yang teratur yang diimplementasikan atau dimonitor oleh seorang aktor (Tabrani & Aghniya, 2019). Simbol-simbol yang digunakan dalam membuat sebuah *UseCase Diagram* diagram dapat dilihat pada Tabel 2. 1.

Tabel 2. 1 Simbol *Use Case Diagram* (Sumber : (Marbun & Sinaga, 2019))

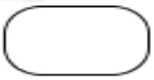
No	Gambar	Nama	Keterangan
1		*Actor*	Mendefinisikan set peran yang dimainkan pengguna saat berinteraksi dengan use case.
2		*Dependency*	Suatu hubungan di mana perubahan dalam satu elemen dapat mempengaruhi elemen lainnya.
3		*Generalization*	Sebuah hubungan di mana anak (keturunan) berbagi perilaku dan struktur data dari objek yang berada di atas.
4		*Include*	Secara eksplisit mendefinisikan sumber *use case*
5		*Extend*	Tentukan perilaku kasus penggunaan target untuk memperluas kasus penggunaan sumber pada titik tertentu.
6		*Association*	Yang menghubungkan satu objek ke objek lain.
7		*System*	Menentukan paket yang menampilkan sistem terbatas.
8		*UseCase*	Aksi yang menghasilkan hasil yang terukur bagi seorang aktor memiliki proses input yang dilakukan oleh seorang aktor.

9		*Collaboration*	Interaksi aturan dan faktor lain yang bekerja sama memberikan perilaku yang lebih besar daripada jumlah faktor individu.
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2.7.2 *Activity Diagram*

Activity diagram memodelkan *workflow* proses bisnis dan urutan aktifitas dalam sebuah proses. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktifitas lainnya atau dari aktifitas ke status (Tabrani & Aghniya, 2019). Simbol-simbol yang digunakan dalam membuat sebuah *Activity Diagram* dapat dilihat pada Tabel 2. 2.

Tabel 2. 2 Simbol *Activity Diagram* (Sumber : (Marbun & Sinaga, 2019))

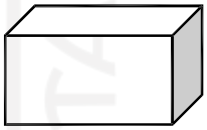
No	Gambar	Nama	Keterangan
1		*Actifity*	Menunjukkan bagaimana setiap kelas antarmuka berinteraksi satu sama lain
2		*Action*	Keadaan sistem mencerminkan pelaksanaan suatu tindakan.
3		*Initial Node*	Bagaimana objek terbentuk atau dimulai
4		*Actifity Final Node*	Bagaimana objek dibentuk dan diakhiri
5		*Fork Node*	Satu aliran berubah pada tahap tertentu menjadi beberapa aliran

2.7.3 Deployment Diagram

Deployment Diagram menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukkan hubungan komputer dengan perangkat (nodes) satu sama lain dan jenis hubungannya dan menampakkan bagian-bagian software yang berjalan

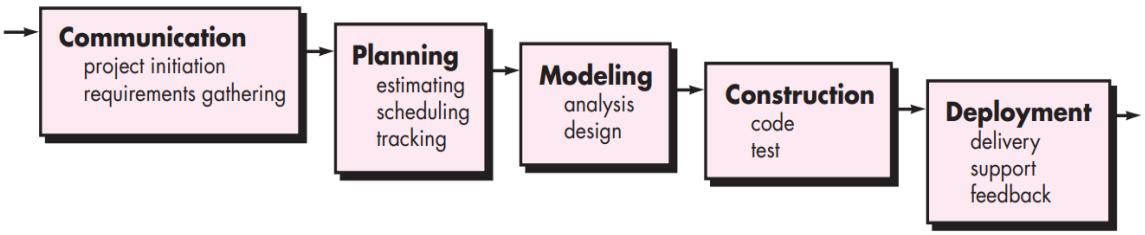
pada bagian hardware (Maiyedra, 2019). Simbol-simbol yang digunakan dalam membuat sebuah *Deployment Diagram* dapat dilihat pada Tabel 2. 3.

Tabel 2. 3 *Deployment Diagram*

No	Simbol	Nama simbol	Deskripsi
.1		Package	Package merupakan simbol bungkusan dari satu atau lebih node
2.		Node	Biasanya mengacu pada perangkat keras (hardware), perangkat lunak yang tidak dibuat sendiri (software), jika didalan mode disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang di ikut sertakan harus sesuai dengan komponen yang telah
3.		Kebergantungan/ depedency	Kebergantungan antar node. Arah panah mengarah pada node yang dipakai.
4.		Link	Relasi antar node

2.8 *Waterfall*

Menurut (Pressman R. S., 2015) nama lain dari *Model Waterfall* adalah Model Air Terjun kadang dinamakan siklus hidup klasik (*classic life cyle*), dimana hal ini menyiratkan pendekatan yang sistematis dan berurutan (sekuensial) pada pengembangan perangkat lunak. Pengembangan perangkat lunak dimulai dari spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (*planning*), pemodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem perangkat lunak ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak yang dihasilkan. Berikut tahapan-tahapan proses *Model Waterfall* pada Gambar 2.1.



Gambar 2. 1 *Waterfall Model*

[Sumber: (Pressman R. S., 2015, hal. 39)]

Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam Model Waterfall menurut (Pressman R. S., 2015) :

1. *Communication*

   Langkah pertama diawali dengan komunikasi kepada konsumen/pengguna. Langkah awal ini merupakan langkah penting karena menyangkut pengumpulan informasi tentang kebutuhan konsumen/pengguna.

2. *Planning*

   Setelah proses *communication* ini,kemudian menetapkan rencana untuk pengerjaan *software* yang meliputi tugas-tugas teknis yang akan dilakukan, risiko yang mungkin terjadi, sumber yang dibutuhkan, hasil yang akan dibuat, dan jadwal pengerjaan.

3. *Modeling*

   Pada proses *modeling* ini menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*, representasi interface, dan detail (algoritma) prosedural.

4. *Construction*

   Construction merupakan proses membuat kode (code generation). *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menerjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

5. *Deployment*

   Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan user. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

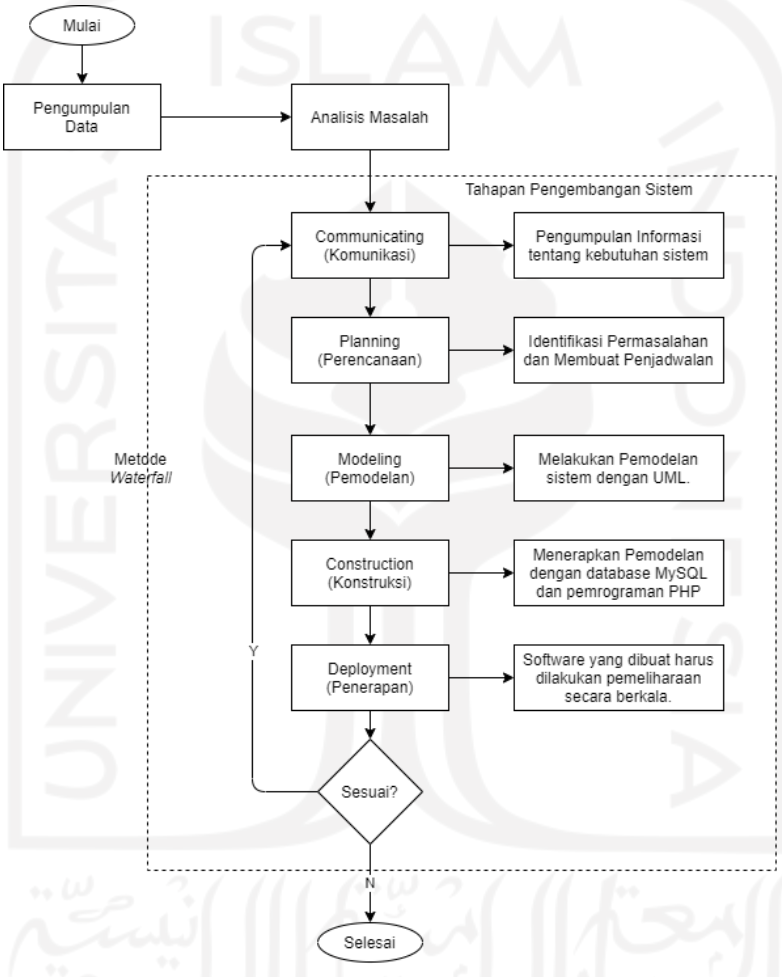
2.9 Black-Box Testing

Metode Black-box Testing adalah metode yang digunakan untuk menguji perangkat lunak tanpa harus memperhatikan detail perangkat lunak. Pengujian ini hanya memeriksa nilai keluaran berdasarkan setiap nilai masukan. Tidak ada upaya untuk mengetahui kode program apa yang digunakan output. Proses Black-Box Testing adalah mencoba program yang telah dibuat dengan memasukkan data pada setiap form. Pengujian ini diperlukan untuk mengetahui apakah program sudah berjalan sesuai kebutuhan perusahaan (Hendri, Manurung, Ferian, Hanaatmoko, & Yulianti, 2020). *Black-Box* Testing digunakan untuk mendeteksi hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi

BAB III
METODOLOGI PENELITIAN

Metodologi penelitian merupakan kerangka pemecahan masalah yang mengambarkan tahap-tahap penyelesaian masalah secara singkat beserta penjelasannya. Berikut metodologi penelitian yang digunakan dalam penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Metodologi Penelitian

3.1 Metode Pengumpulan Data

Dalam melakukan penelitian ini, peneliti melakukan pengumpulan data primer dan pengumpulan data sekunder. Berikut merupakan hasil dari pengumpulan data yang telah dilakukan:

3.1.1 Pengumpulan Data Primer

Dalam melakukan pengumpulan Data Primer ini peneliti melakukan wawancara kepada salah satu pihak Rumah Tahfiz serta melakukan observasi lapangan maupun online guna mengetahui proses berjalan terkait proses pembelajaran di Rumah Tahfiz.

A. Wawancara

Pada tahap ini dilakukannya wawancara secara daring kepada salah satu pihak pengajar Rumah/Pondok Tahfizh. Narasumber wawancara pada penelitian ini adalah Ustadz Muhammad Fajar Suardi yang merupakan pengasuh di Rumah Tahfizh Pusat Dakwah Al-Qur'an (PDA) yang beralamatkan di Jl. Beringin No.7, Leuwinanggung, Kec. Tapos, Kota Depok, Jawa Barat.

B. Observasi

Pada tahap ini penulis melakukan pengamatan langsung secara online pada proses bisnis rumah tahfizh. Penulis melakukan wawancara kepada beberapa ustadz/muhafizh yang mengajar, juga memantau kegiatan pembelajaran online yang berlangsung. Hasil yang didapatkan adalah Rumah Tahfizh Pusat Dakwah Al-Qur'an masih tetap melakukan pembelajaran secara daring meski selama pandemi banyak santri yang pulang kampong. Selain itu, rumah tahfizh ini juga beradaptasi dengan membuka pendaftaran program tahfizh secara daring sehingga santri tidak harus tinggal di asrama. Santri dapat mengikuti kegiatan pembelajaran secara daring dari manapun.

Selama proses pembelajaran, rumah tahfizh ini belum memiliki/menggunakan aplikasi khusus untuk pembelajaran daring. Seluruh proses belajar dilaksanakan via *platform chatting* seperti Whatsapp atau *platform* video *conference* seperti Zoom. Santri bertalaqqi dengan ustadz via panggilan video untuk menyetor hafalannya. Untuk materi pembelajaran lain, ustadz mengirimkan materi dan tugas dalam bentuk file pdf atau word. Kemudian santri mempelajari dan mengumpulkan tugas evaluasi juga via grup Whatsapp. Ustadz akan mengumumkan nilai atau hasil evaluasi tugas yang dikumpulkan santri pada grup Whatsapp.

Di rumah tahfizh ini santri belajar dan menghafalkan Al-Qur'an selama setahun. Ketika selesai masa pendidikan, maka seluruh santri akan diuji hafalan yang telah mereka dapatkan. Materi keislaman penunjang lainnya juga diujikan selama ujian akhir. Setelah lulus, sebagian santri ada yang mengabdi menjadi ustadz/muhafizh selama setahun dan juga ada yang langsung melanjutkan ke jenjang perguruan tinggi.

3.1.2 Pengumpulan Data Sekunder

Dalam melakukan pengumpulan Data sekunder ini peneliti mengumpulkan dengan cara membaca, me*review* dan mempelajari buku atau jurnal terkait, seperti mereview terkait *Learning Management System* (LMS), *Service Oriented Architecture* (SOA) dan lain sebagainya.

3.2 Analisis Masalah

Setelah tahapan pengumpulan data peneliti menemukan berapa masalah yang terdapat di rumah tahfizh yaitu, masih melakukan sistem penyampaian informasi pendidikan secara konvensional, sering kali tidak mendapatkan pemahaman dari materi yang disampaikan oleh pengajar, membaca dan menghafal Al-Quran menjadi terhambat mengingat tidak adanya sarana yang dapat digunakan dalam kegiatan belajar mengajar antara santri/santriwati dengan pengajar.

3.3 Communicating (Komunikasi)

Tahap komunikasi merupakan tahapan komunikasi dengan konsumen/pengguna karena menyangkut pengumpulan informasi tentang kebutuhan konsumen/pengguna. Setelah melakukan komunikasi, peneliti melakukan analisis kebutuhan fungsional dan analisis kebutuhan non fungsional. Adapun uraiannya sebagai berikut:

3.3.1 Analisis Kebutuhan Fungsional

Dalam melakukan penelitian ini, peneliti menganalisis kebutuhan fungsional yaitu:

1. Admin

   Admin LMS memiliki hak untuk mengelola dan mengatur pembelajaran berbasis *Learning Management System* (LMS)

2. Guru/Usta`dz

   Guru memiliki hak untuk mengelola data diri, kegiatan mengajar, dan nilai anak didik termasuk didalamnya menambahkan, menghapus, melihat, dan mengubah. Guru juga dapat mencetak hasil e-*raport* anak didik

3. Santri

   Santri memiliki hak akses untuk melihat tugas, melihat kelas dan melihat nilai.

3.3.2 Analisis Kebutuhan Non Fungsional

Agar dapat mendukung hasil analisis penelitian ini, diperlukan adanya suatu implementasi terhadap sistem tersebut. Hal ini dilakukan agar sistem dapat memberikan dukungan hasil keputusan. Sistem yang digunakan harus mendukung untuk memberikan hasil yang terbaik. Oleh karena itu, perlu dibuat suatu perancangan implementasi kebutuhan dalam pembuatan laporan agar sesuai kebutuhan dari *Learning Management System* (LMS) pada Rumah/Pondok Tahfiz. Sumber daya teknologi tersebut meliputi kebutuhan *software* dan *hardware* agar sistem aplikasi penjualan berbasis website ini dapat berjalan dengan baik.

1. Kebutuhan Spesifikasi Perangkat Lunak (*Software*)

   Berikut merupakan kebutuhan spesifikasi *software* dari penelitian ini.

   Sistem Operasi : Windows 10 Pro 64bit
   Database : MySQL
   Pemrograman : Laravel
   Kode Editor : Sublime Text 3

2. Kebutuhan Spesifikasi Perangkat Keras (*Hardware*)

   Adapun spesifikasi *hardware* yang dibutuhkan dapat dilihat pada tabel berikut:

   Laptop : Processor : Intel
   RAM : 4 GB
   HDD : 1 Terabyte
   Grapichs : NVIDIA Geforce 720

3.4 *Planning* (Perencanaan)

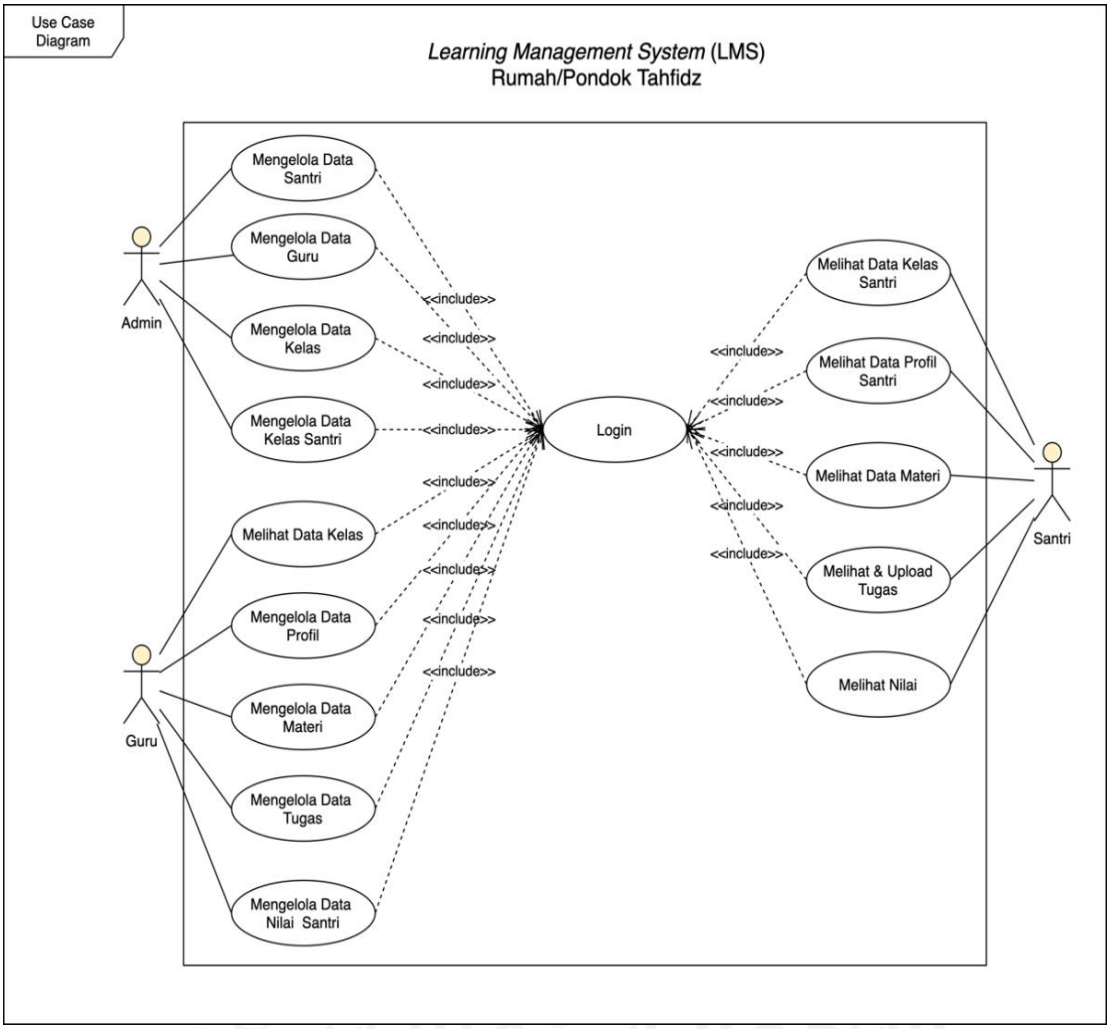
Setelah proses *communication* ini,kemudian menetapkan rencana untuk pengerjaan *software* yang meliputi tugas-tugas teknis yang akan dilakukan, risiko yang mungkin terjadi, sumber yang dibutuhkan, hasil yang akan dibuat, dan jadwal pengerjaan.

3.5 *Modeling* (Pemodelan)

Pada pembuatan *Learning Management System* (LMS) pada Rumah Tahfizh peneliti membuat perancangan sistem menggunakan UML seperti *use case* diagram, *activity* diagram dan *deployment* diagram serta perancangan database.

3.5.1 *Use Case* Diagram

Use Case Diagram menjelaskan interaksi yang terjadi pada sistem dengan aktor pengguna sistem. Berdasarkan analisis dari data yang telah dikumpulkan maka dihasilkan diagram *use case* seperti pada Gambar 3.2.



Gambar 3. 2 *Use Case Diagram*

Berdasarkan *use case* diagram di atas terdapat 15 *use case* dan 3 aktor yang mana setiap *use case* memiliki *prerequisite* atau persyaratan awal yang harus dipenuhi yang ditandai dengan panah *include*. Oleh karena itu, supaya dapat mengakses setiap *use case*, maka aktor harus login terlebih dahulu. Berikut adalah penjelasan dari *use case* diagram yang dapat dilihat pada Tabel 3.1.

Tabel 3. 1 Use Case

Use Case	**Deskripsi**
UC01 – Mengelola Data Santri	Admin dapat mengelola data santri sepeti tambah, ubah dan hapus data santri.
UC02 – Mengelola Data Guru	Admin dapat mengelola data guru sepeti tambah, ubah dan hapus data guru.
UC03 – Mengelola Data Kelas	Admin dapat mengelola data kelas sepeti tambah, ubah dan hapus data kelas.
UC04 – Mengelola Data Kelas Santri	Admin dapat mengelola data kelas santri sepeti tambah, ubah dan hapus data kelas santri.
UC05 – Melihat Data Kelas	Guru dapat melihat data kelas.
UC06 – Mengelola Data Profil	Guru dapat mengelola data kelas profil seperti mengubah data profil
UC07 – Mengelola Data Materi	Guru dapat mengelola data materi sepeti tambah, ubah dan hapus data materi.
UC08 – Mengelola Data Tugas	Guru dapat mengelola data tugas sepeti tambah, ubah dan hapus data tugas.
UC09 – Mengelola Data Nilai Santri	Guru dapat mengelola data nilai santri sepeti tambah, ubah dan hapus data nilai santri.
UC10 – Melihat Data Kelas Santri	Santri dapat melihat data kelas santri.
UC11 – Melihat Data Profil	Santri dapat melihat data profil.
UC12 – Melihat Data Materi	Santri dapat melihat data materi.
UC13 – Melihat dan Upload Tugas	Santri dapat melihat dan mengupload data tugas.
UC14 – Melihat Nilai	Santri dapat melihat nilai.
UC15 – Login	Pengguna dapat melakukan login dengan memasukkan *username* dan *password* yang benar.

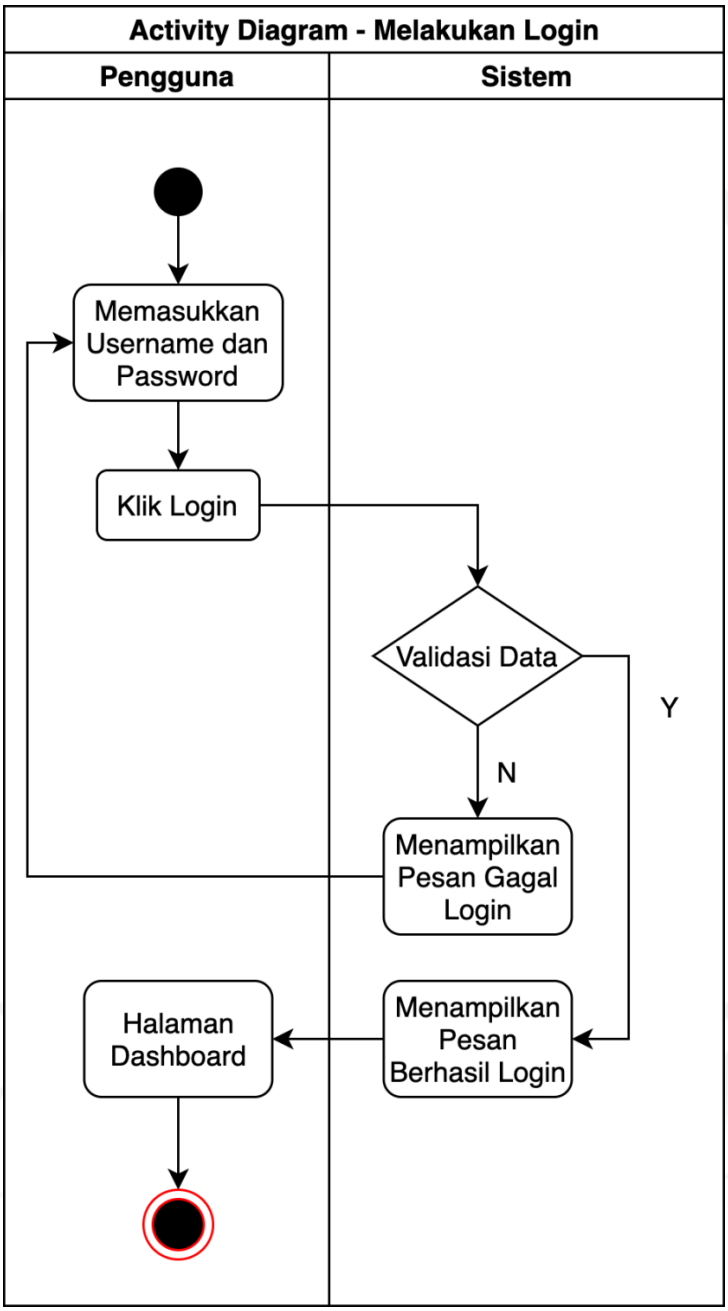
3.5.2 *Activity* Diagram

Activity diagram merupakan suatu diagram yang menjelaskan mengenai alur kerja pada sistem. Berikut merupakan *activity* diagram dari LMS Rumah Tahfidz.

1. Admin

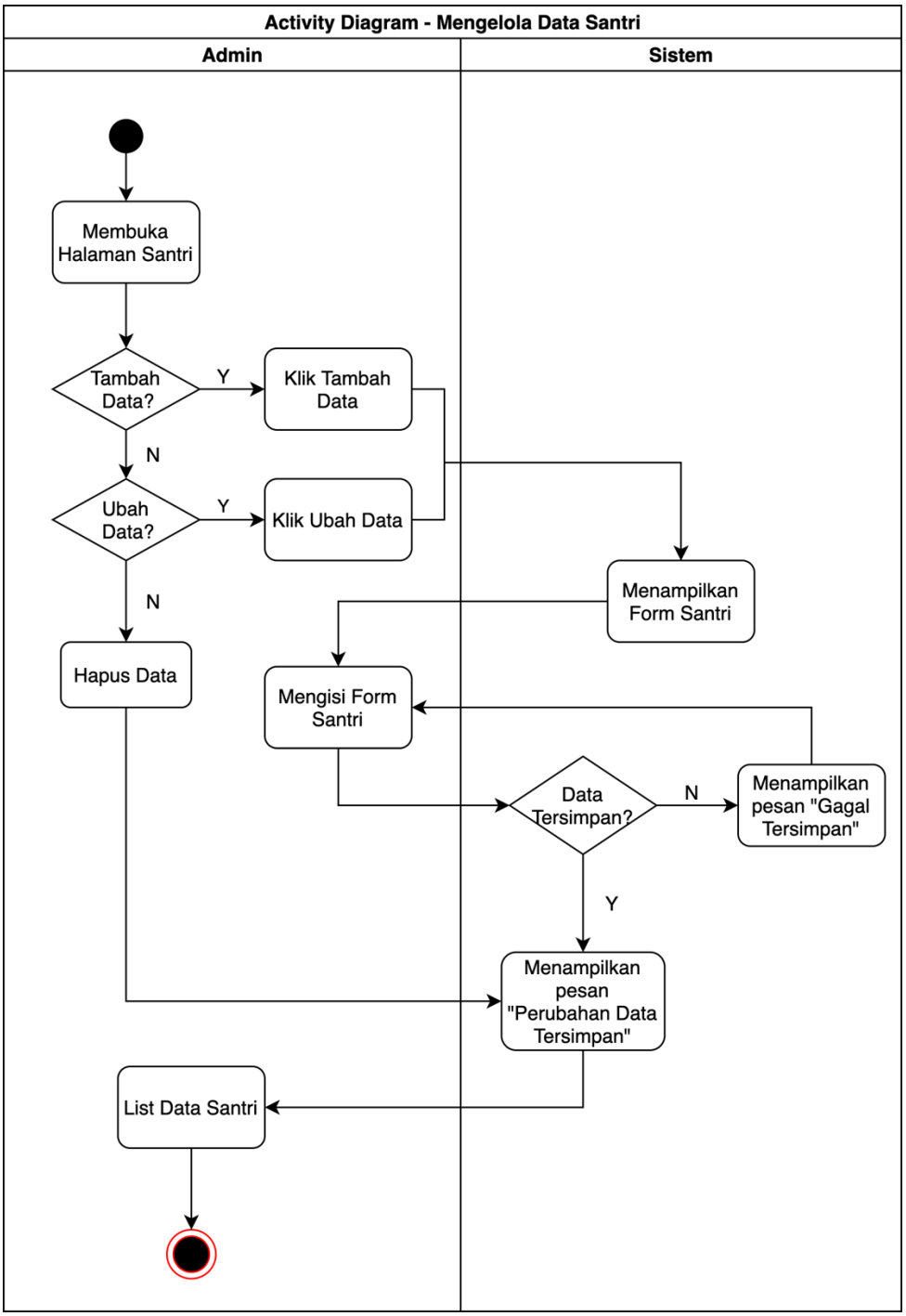
Pada *activity diagram* admin terdapat beberapa alur atau proses sebagai berikut:

a. *Login* atau masuk. Admin melakukan *login* dengan memasukkan *username* dan *password*. Jika berhasil akan diarahkan menuju halaman *dashboard* admin. *Activity* diagram *login* admin dapat dilihat pada Gambar 3.3.



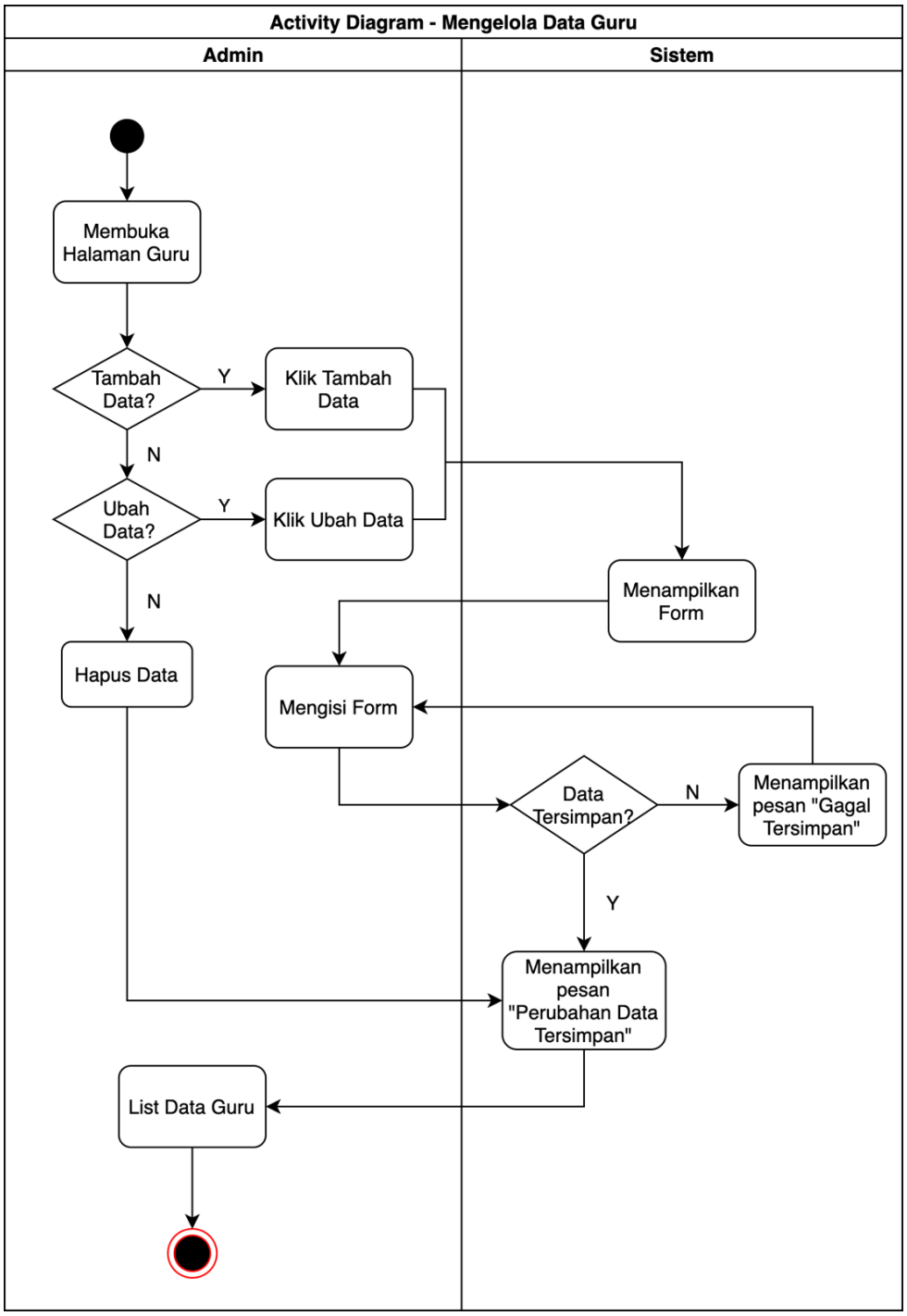
Gambar 3. 3 *Activity Diagram Login*

b. Mengelola Data Santri. Admin dapat menambahkan data santri, mengubah dan menghapus data santri. *Activity* diagram mengelola data santri dapat dilihat pada Gambar 3.4.



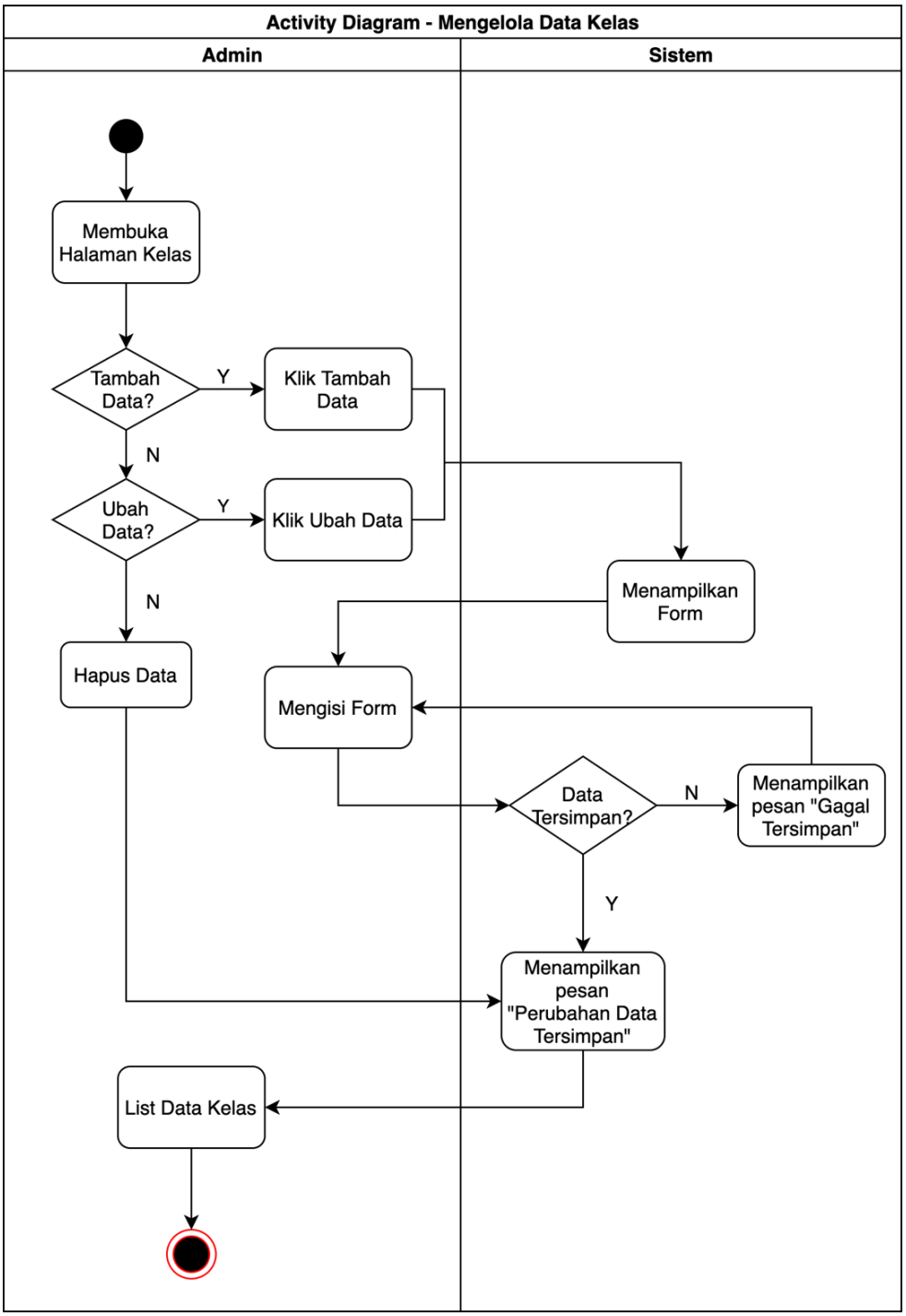
Gambar 3. 4 *Activity Diagram* Mengelola Data Santri

c. Mengelola Data Guru. Admin dapat menambahkan data guru, mengubah dan menghapus data santri. *Activity* diagram mengelola data guru dapat dilihat pada Gambar 3.5.



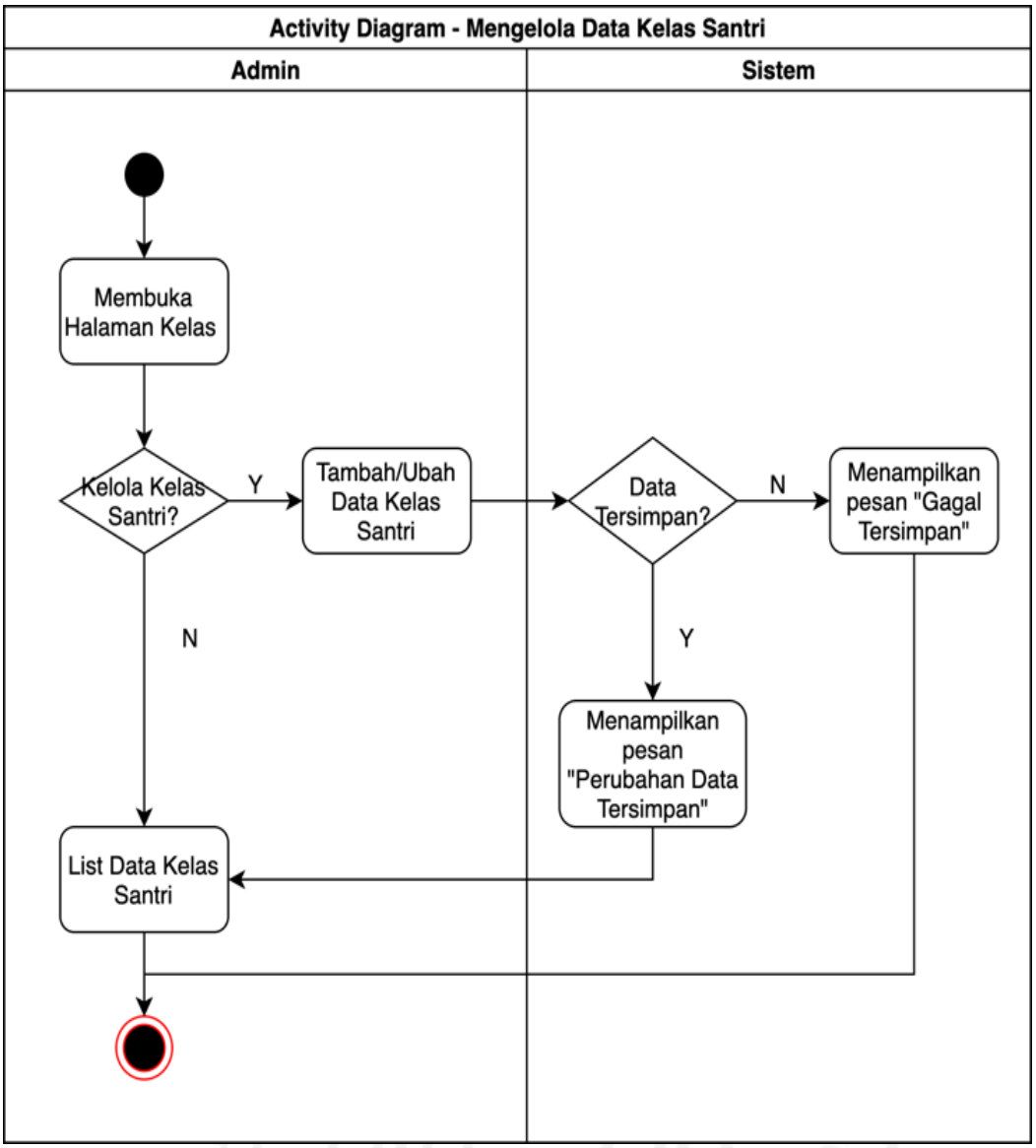
Gambar 3. 5 *Activity Diagram* Mengelola Data Guru

d. Mengelola Data Kelas. Admin dapat menambahkan data kelas yang tersedia, mengubah dan menghapus data kelas. *Activity* diagram mengelola data kelas dapat dilihat pada Gambar 3.6.



Gambar 3. 6 *Activity* Diagram Mengelola Data Kelas

e. Mengelola Data Kelas Santri. Admin dapat menambahkan santri ke dalam kelas yang tersedia, mengubah dan menghapus data santri dari kelas. *Activity* diagram mengelola data kelas santri dapat dilihat pada Gambar 3.7.

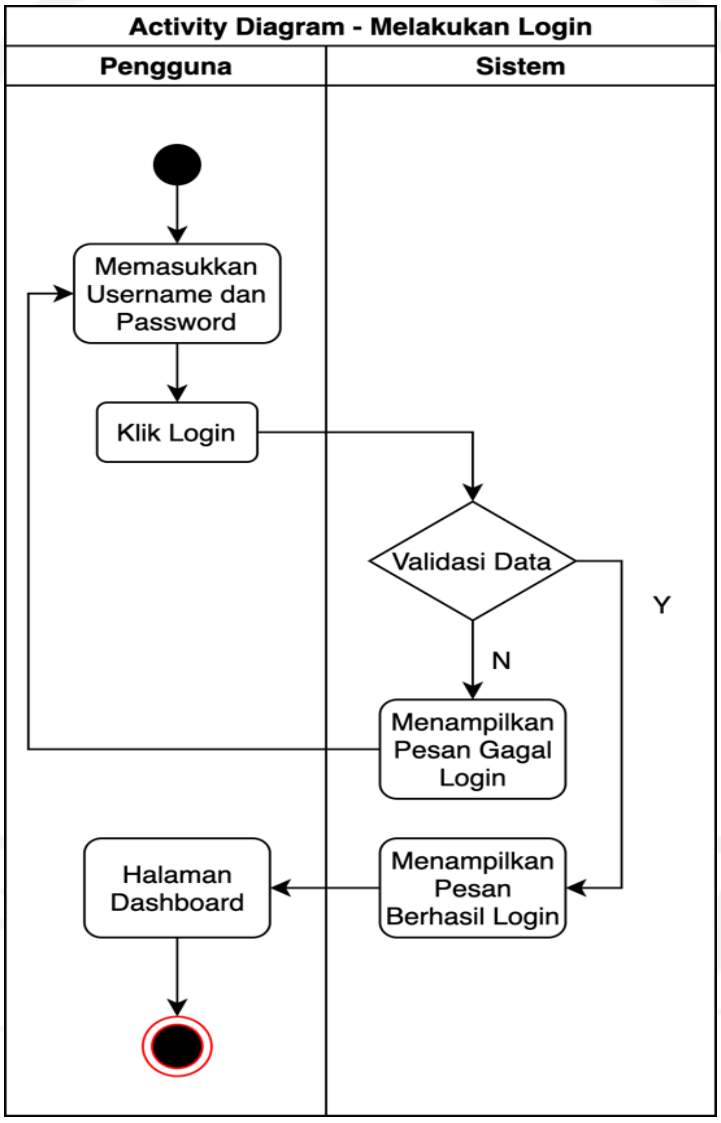


Gambar 3. 7 *Activity* Diagram Mengelola Data Kelas Santri

2. Guru/Ustadz

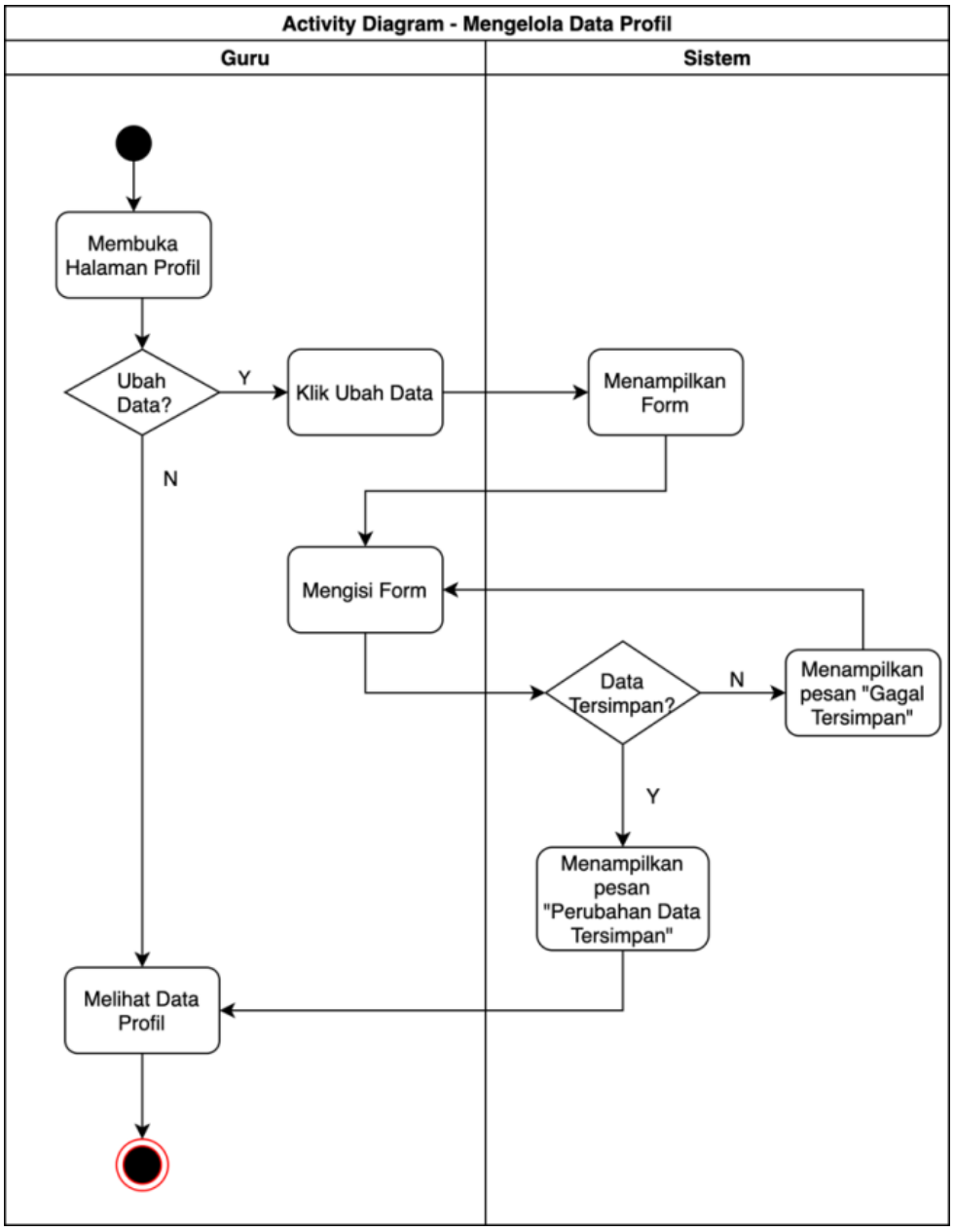
Pada *activity diagram* guru/ustadz terdapat beberapa alur atau proses sebagai berikut:

a. *Login* atau masuk. Guru/Ustadz melakukan *login* dengan memasukkan *username* dan *password.* Jika berhasil akan diarahkan menuju halaman *dashboard* Guru/Ustadz. *Activity* diagram *login* Guru/Ustadz dapat dilihat pada Gambar 3.8.



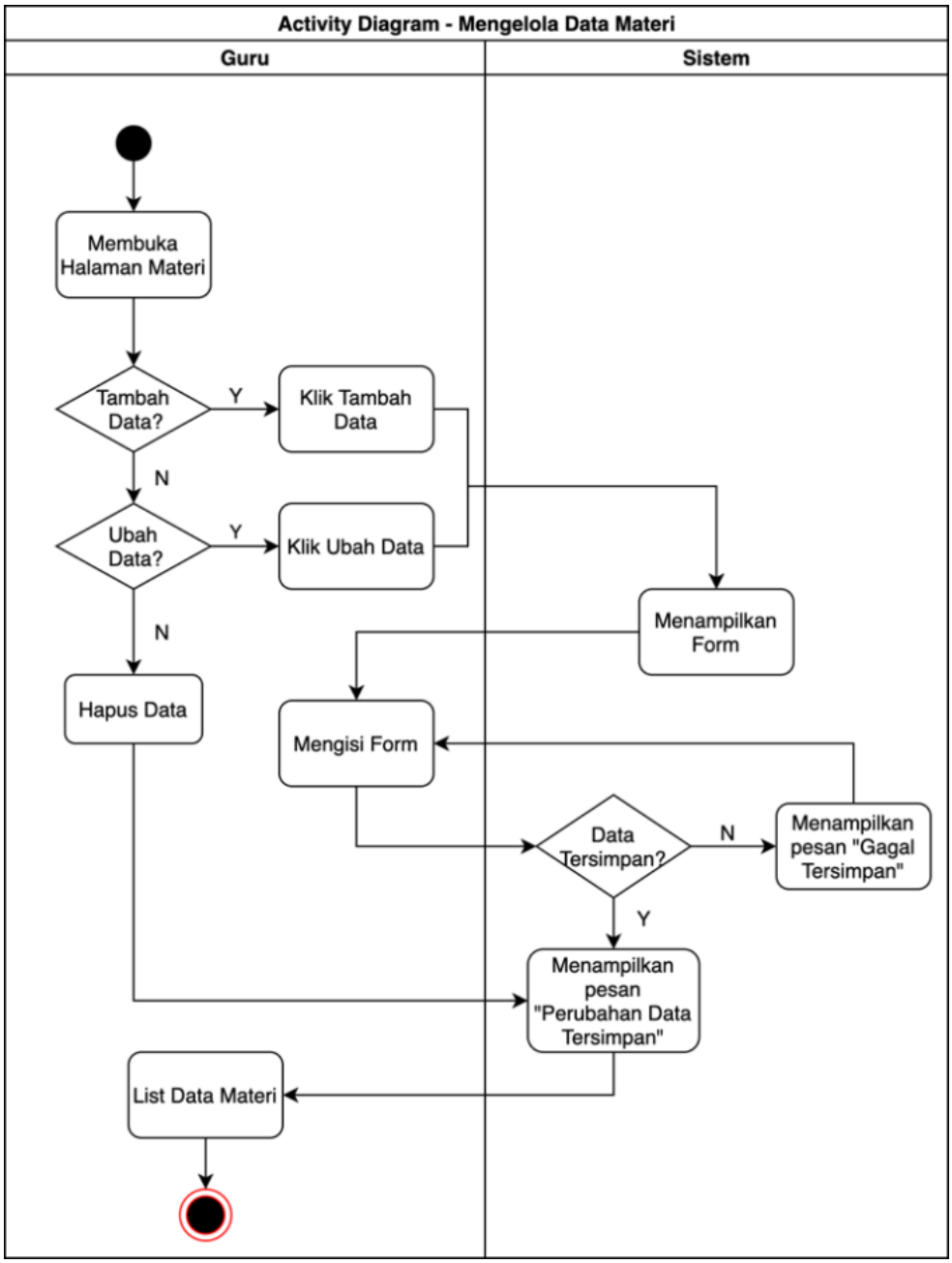
Gambar 3. 8 *Activity* Diagram Login

b. Mengelola Data Profil. Guru/Ustadz dapat melihat profil dirinya dan mengubah data profilnya. *Activity* diagram mengelola data profil dapat dilihat pada Gambar 3.9.



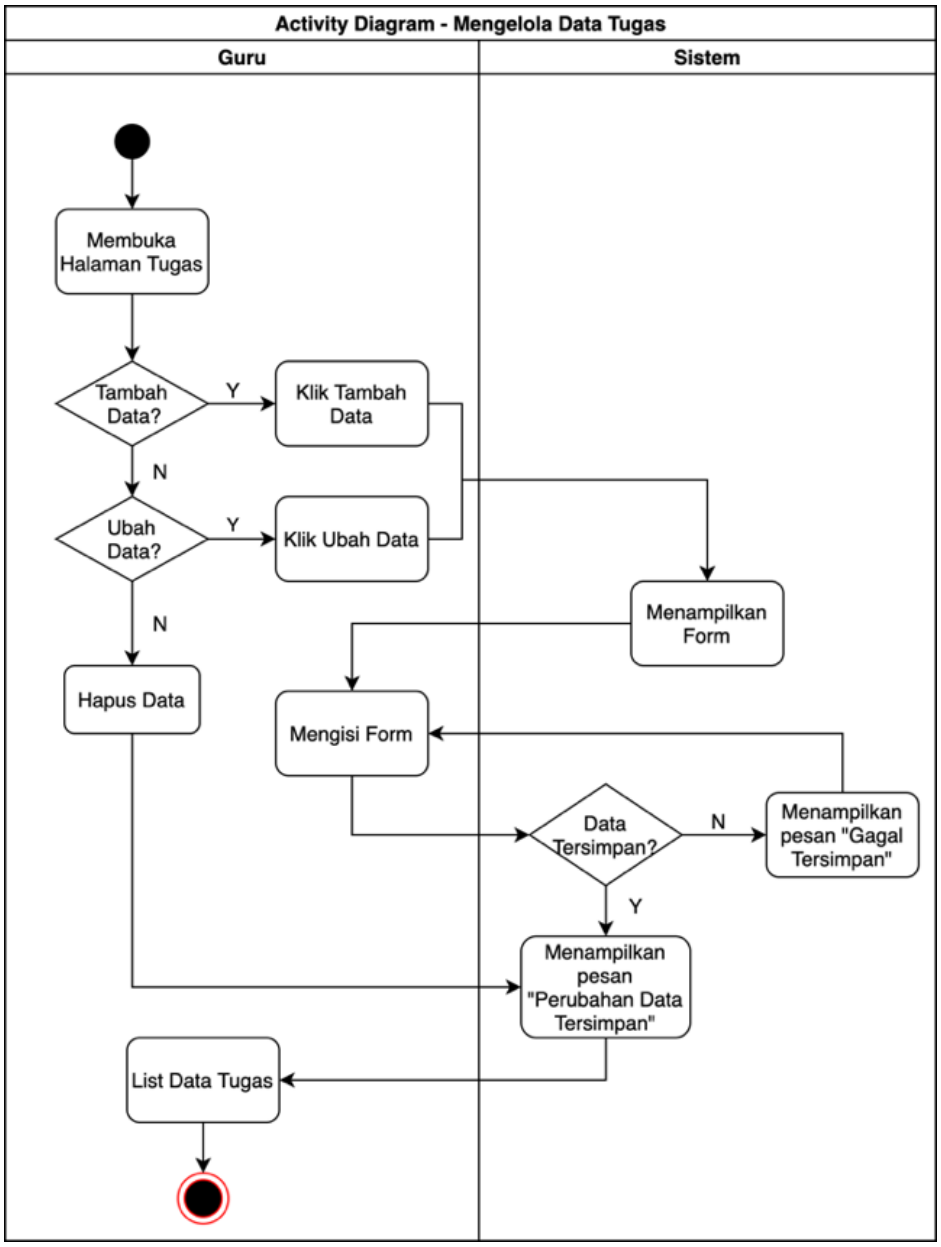
Gambar 3. 9 *Activity* Diagram Mengelola Data Profil

c. Mengelola Data Materi. Guru/Ustadz dapat menambahkan materi pembelajaran, mengubah dan menghapus data materi. *Activity* diagram mengelola data materi dapat dilihat pada Gambar 3.10.



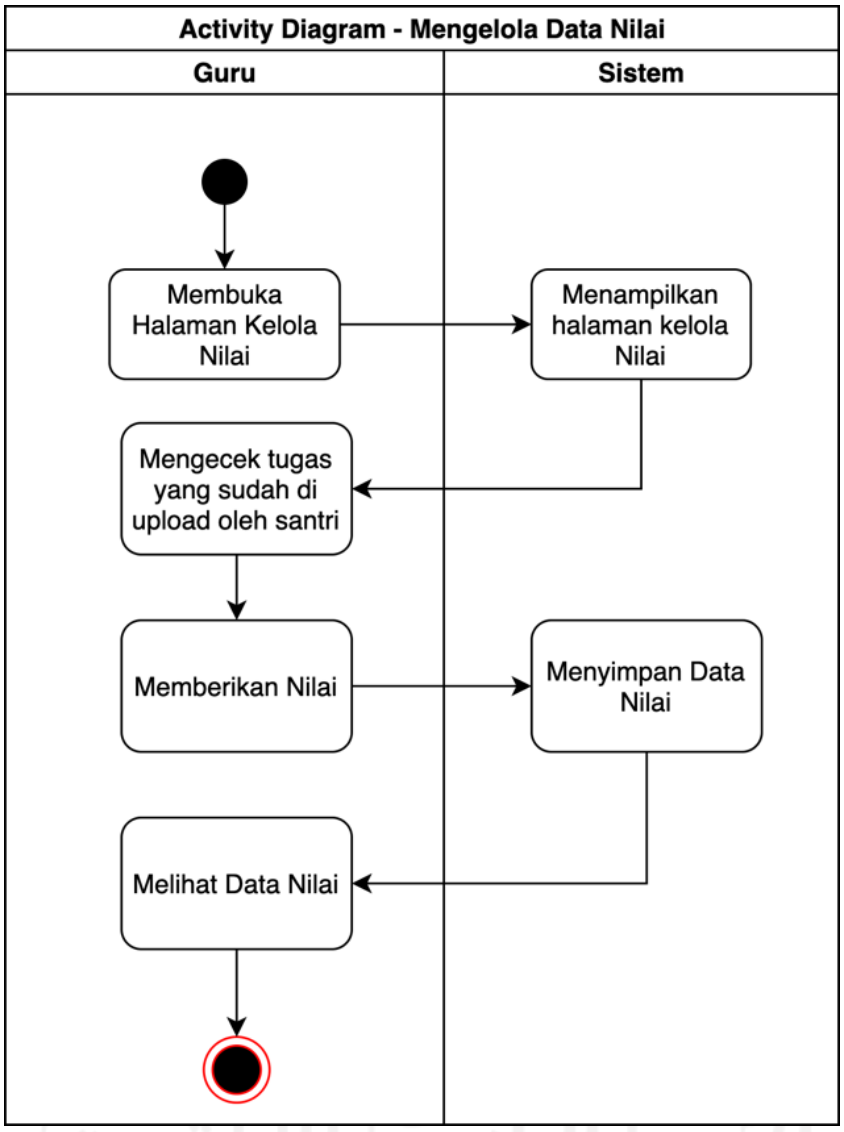
Gambar 3. 10 *Activity* Diagram Mengelola Data Materi

d. Mengelola Data Tugas. Guru/Ustadz dapat menambahkan tugas evaluasi, mengubah dan menghapus data tugas. *Activity* diagram mengelola data tugas dapat dilihat pada Gambar 3.11.



Gambar 3. 11 *Activity* Diagram Mengelola Data Tugas

e. Mengelola Data Nilai. Guru/Ustadz dapat memberikan nilai pada tugas yang dikumpulkan oleh santri. *Activity* diagram mengelola data nilai dapat dilihat pada Gambar 3.12.

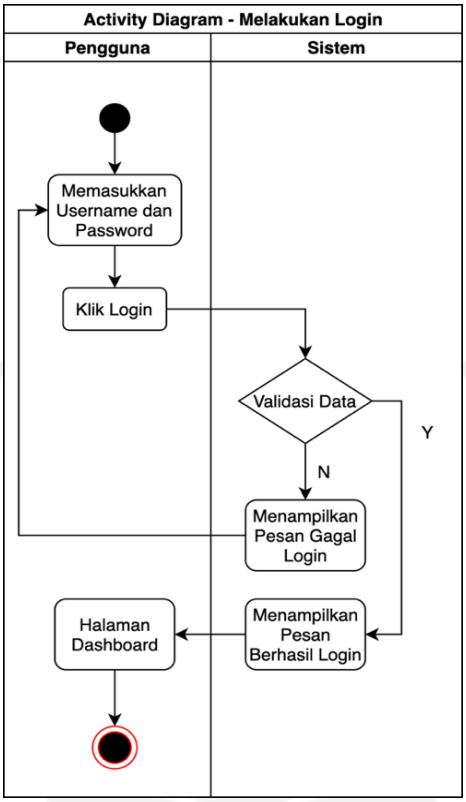


Gambar 3. 12 Activity Diagram Mengelola Data Nilai Santri

3. Santri

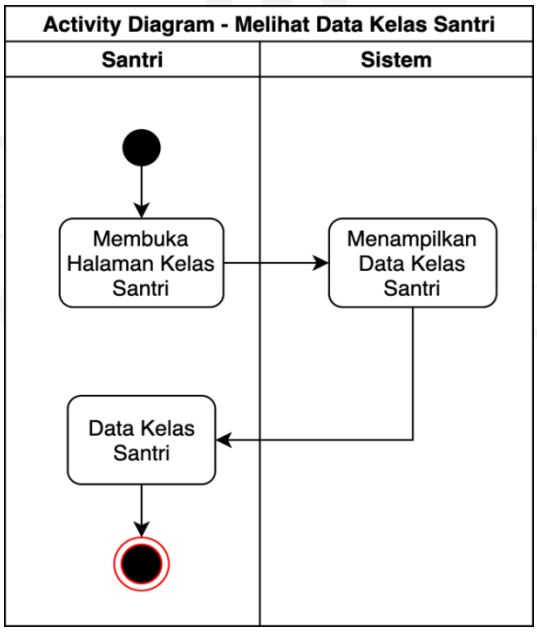
Pada *activity diagram* santri terdapat beberapa alur atau proses sebagai berikut:

a. *Login* atau masuk. Santri melakukan *login* dengan memasukkan *username* dan *password.* Jika berhasil akan diarahkan menuju halaman *dashboard* Santri. *Activity* diagram *login* Santri dapat dilihat pada Gambar 3.13.



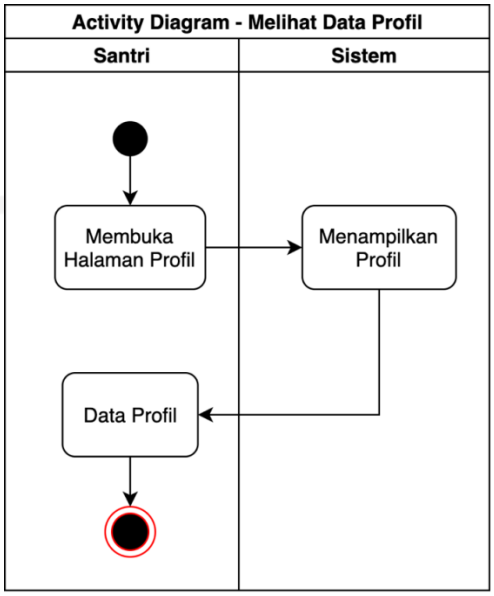
Gambar 3. 13 *Activity* Diagram Login

b. Melihat Data Kelas Santri. Santri dapat melihat data kelas apa saja yang diikutinya. *Activity* diagram melihat data kelas santri dapat dilihat pada Gambar 3.14.



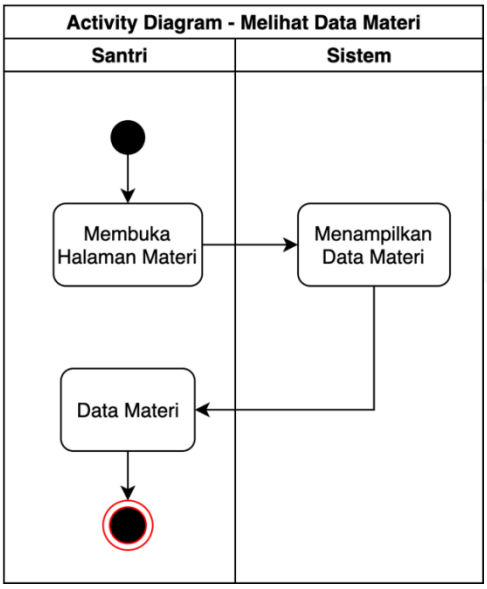
Gambar 3. 14 *Activity* Diagram Melihat Data Kelas Santri

c. Melihat Data Profil. Santri dapat melihat profil data dirinya. *Activity* diagram melihat data profil dapat dilihat pada Gambar 3.15.



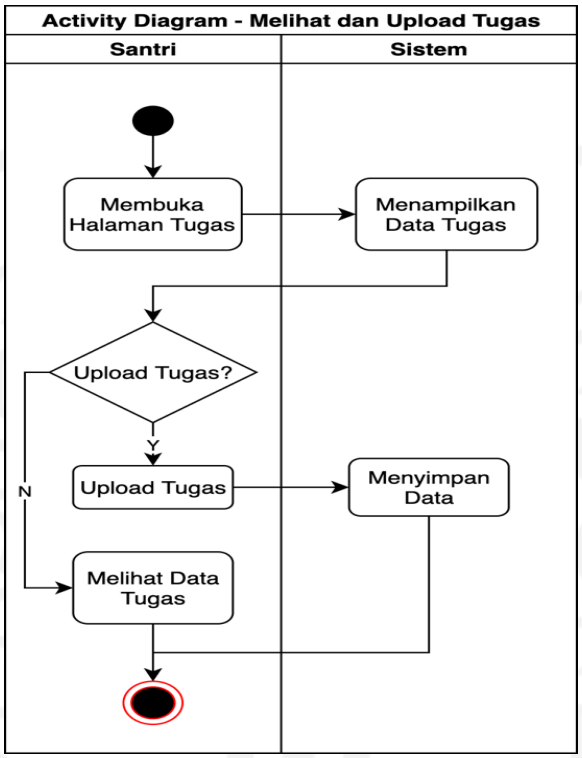
Gambar 3. 15 *Activity* Diagram Melihat Data Profil

d. Melihat Data Materi. Santri dapat melihat materi pembelajaran yang diberikan oleh gurunya pada tiap kelas yang diikuti. *Activity diagram* melihat data materi dapat dilihat pada Gambar 3.16.



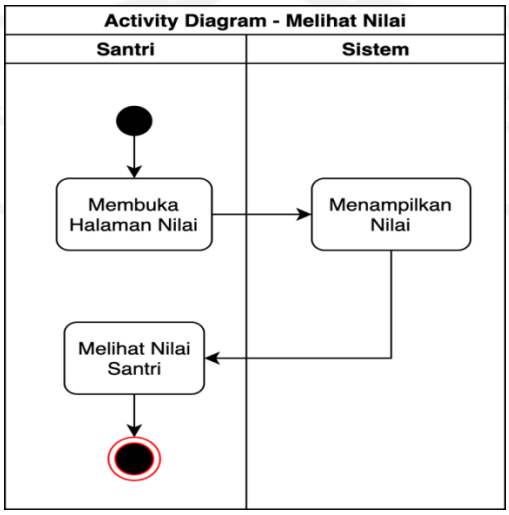
Gambar 3. 16 *Activity* Diagram Melihat Data Materi

e. Melihat dan *Upload* Tugas. Santri dapat melihat tugas yang diberikan oleh guru di kelasnya dan mengumpulkan tugas yang telah dikerjakan. *Activity diagram* melihat dan *upload* tugas dapat dilihat pada Gambar 3.17.



Gambar 3. 17 *Activity* Diagram Melihat dan Upload Tugas

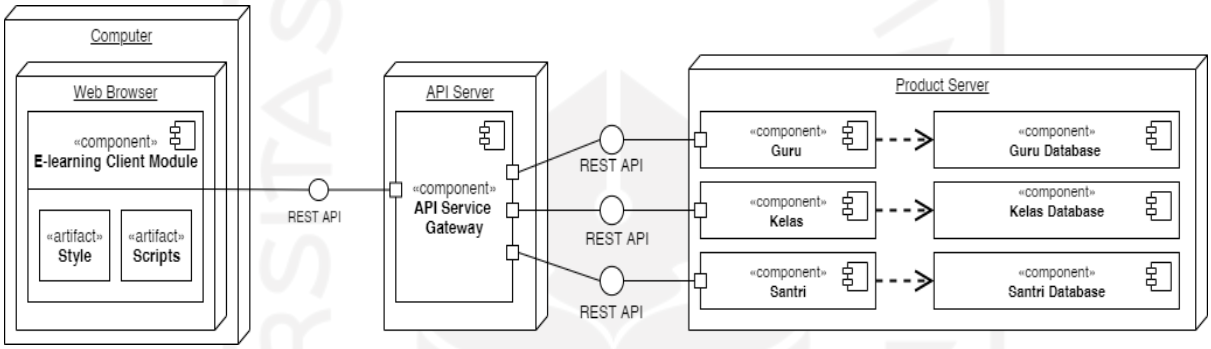
f. Melihat Nilai. Santri dapat melihat nilai dari tugas yang telah dikumpulkannya. *Activity diagram* melihat nilai dapat dilihat pada Gambar 3.18.



Gambar 3. 18 *Activity* Diagram Melihat Nilai

3.5.3 Deployment Diagram

Deployment diagram digunakan untuk menggambarkan/memvisualisasikan secara umum proses yang terjadi pada suatu sistem. Dalam penelitian ini client menggunakan aplikasi berbasis web yang berkomunikasi melalui REST API. REST API merupakan penerapan API (*Application Programming Interface)* dan REST adalah sebuah arsitektur metode interaksi/komunikasi yang menggunakan protokol HTTP dalam pertukaran data yang mana sering digunakan dalam pengmbangan aplikasi. Adapun *Deployment Diagram* ditunjukkan pada Gambar 3.19.



Gambar 3. 19 *Deployment* Diagram

Pada gambar di atas, *product server* pada *backend* berisi beberapa tabel basisdata. Ketika REST *client* pada *frontend* mengakses atau mengirimkan sebuah data melalui HTTP *Request* pada REST API, maka REST *server* akan menjawab melalui HTTP *Response.* Pada setiap *request* yang dilakukan oleh REST *client,* terdapat metode berikut yang dipakai agar *server* memahami apa yang sedang diminta oleh *client*:

- GET, merupakan metode HTTP *Request* yang digunakan untuk membaca atau mengambil data dari *resource.*
- POST, merupakan metode HTTP *Request* yang digunakan untuk menambah dan membuat data baru dengan menyisipkan data dalam *body* saat *request* dilakukan.
- PUT, merupakan metode HTTP *Request* yang digunakan untuk melakukan *update/edit* data pada *resource*.
- DELETE, merupakan metode HTTP *Request* yang digunakan untuk menghapus suatu data pada *resource*.

Setiap *request* yang dilakukan REST *client* akan menerima data *response* dari *server. Response* tersebut berupa data dalam bentuk JSON atau XML. Setelah mendapatkan data

response tersebut, *client* dapat menggunakan datanya dengan cara *parsing* atau mengurai data tersebut dan diolah sesuai kebutuhan.

3.5.4 Perancangan Basis Data

Pada sebuah sistem membutuhkan wadah untuk menyimpan data. Perancangan basis data bertujuan untuk mengetahui relasi antar data yang dibutuhkan sistem. Berikut merupakan perancangan basis data pada LMS di Rumah Tahfidz.

1. Tabel Admin

   Tabel basis data untuk admin dapat dilihat pada Tabel 3.2.

Tabel 3. 2 Tabel Admin

Kolom	***Type* Data**	**Keterangan**
admin_id	Integer (11)	*Primary Key*
user_id	Integer (11)	*Foreign Key*
name	Varchar (255)	
jenis_kelamin	Varchar (255)	
phone	Varchar (255)	

2. Tabel Guru/Ustadz

   Tabel basis data untuk guru/ustadz dapat dilihat pada Tabel 3.3.

Tabel 3. 3 Tabel Guru/Ustadz

Kolom	***Type* Data**	**Keterangan**
admin_id	Integer (11)	*Primary Key*
user_id	Integer (11)	*Foreign Key*
name	Varchar (255)	
jenis_kelamin	Varchar (255)	
phone	Varchar (255)	

3. Tabel Kelas

Tabel basis data untuk kelas dapat dilihat pada Tabel 3.4.

Tabel 3. 4 Tabel Kelas

Kolom	***Type* Data**	**Keterangan**
kelas_id	Integer (11)	*Primary Key*
nama_kelas	Varchar (255)	
deskripsi	Text	
guru_id	Integer (11)	*Foreign Key*

4. Tabel Kelas Materi

Tabel basis data untuk kelas materi dapat dilihat pada Tabel 3.5.

Tabel 3. 5 Tabel Kelas Materi

Kolom	***Type* Data**	**Keterangan**
kelas_materi_id	Integer (11)	*Primary Key*
kelas_id	Integer (11)	*Foreign Key*
judul_materi	Varchar (255)	
materi_konten	Text	
file_materi	Text	

5. Tabel Kelas Member

Tabel basis data untuk kelas member dapat dilihat pada Tabel 3.6.

Tabel 3. 6 Tabel Kelas Member

Kolom	***Type* Data**	**Keterangan**
santri_id	Integer (11)	*Primary Key*
kelas_id	Integer (11)	*Foreign Key*

6. Tabel Kelas Tugas

Tabel basis data untuk kelas tugas dapat dilihat pada Tabel 3.7.

Tabel 3. 7 Kelas Tugas

Kolom	***Type* Data**	**Keterangan**
kelas_tugas_id	Integer (11)	*Primary Key*
kelas_id	Integer (11)	*Foreign Key*
judul_tugas	Varchar (255)	
deskripsi	Text	
nilai_kelulusan	Decimal (10)	
start_date	Date	
end_date	Date	
status	Varchar (20)	

7. Tabel Kelas Tugas Santri

Tabel basis data untuk kelas tugas santri dapat dilihat pada Tabel 3.8.

Tabel 3. 8 Tabel Kelas Tugas Santri

Kolom	***Type* Data**	**Keterangan**
kelas_tugas_santri_id	Integer (11)	*Primary Key*
tugas_id	Integer (11)	*Foreign Key*
santri_id	Integer (11)	*Foreign Key*
nilai	Varchar (255)	

8. Tabel Santri

Tabel basis data untuk santri dapat dilihat pada Tabel 3.9.

Tabel 3. 9 Tabel Santri

Kolom	***Type* Data**	**Keterangan**
santri_id	Integer (11)	*Primary Key*
user_id	Integer (11)	*Foreign Key*
name	Varchar (255)	
jenis_kelamin	Varchar (255)	
phone	Varchar (255)	

9. Tabel *Users*

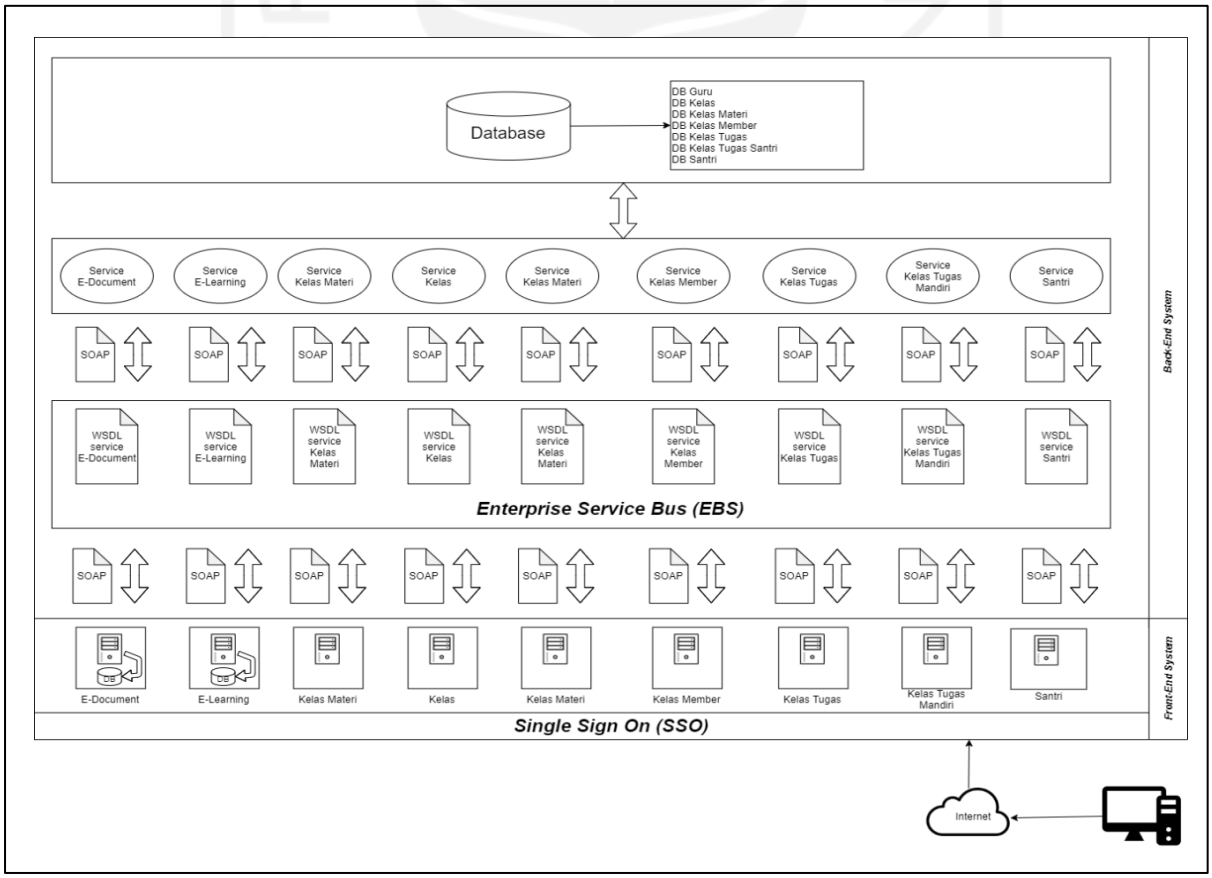
Tabel basis data untuk *users*/pengguna dapat dilihat pada Tabel 3.10.

Tabel 3. 10 Tabel Users

Kolom	***Type* Data**	**Keterangan**
user_id	Integer (11)	*Primary Key*
name	Varchar (255)	
email	Varchar (255)	
password	Varchar (255)	
roles	Varchar (255)	

3.5.5 Arsitektur Service Oriented Architecture

Dalam sistem rumah tahfizh, sistem dirancang agar dapat berintegrasi dengan sistem pada departemen lain. Berikut merupakan Arsitektur *Service Oriented Architecture* pada LMS Rumah Tahfidz yang dapat dilihat pada Gambar 3.20.



Gambar 3. 20 Arsitektur SOA Rumah Tahfizh

Berdasarkan gambar 3.20 dapat dijelaskan bahwa proses pada sistem rumah tahfizh berawal ketika *user* melakukan *login*. Setelah *login* ke dalam sistem maka masuk ke aplikasi. Untuk memudahkan hal manajemen modul, pada sistem *learning management system* menggunakan basisdata MySQL. Sehingga bisa tetap saling berhubungan dengan sistem lain dalam sistem rumah tahfizh. Dalam arsitektur ini, metode akses basisdata sistem terletak terpisah dengan tampilan antarmuka. Basisdata diakses dengan API melalui *request* dari sisi *frontend* kepada sistem pada *backend. Frontend* bertindak sebagai *client* dan *backend* berperan sebagai *service provider.*

Ketika *user* mengakses tampilan antarmuka dan melakukan *login*, maka sistem akan melakukan *request* ke *service provider* dengan HTTP *request POST*. Jika autentikasi berhasil maka *service provider* akan menyampaikan data yang diakses *client* pada tampilan antarmuka. Begitu pula jika *user* hendak mengubah *password* maka sistem akan melakukan *request* dengan HTTP *request PUT*. Jika berhasil, maka *service provider* akan memperbarui perubahan yang telah dibuat oleh *user.* Proses ini berulang tiap kali *user* mengakses halaman pada tampilan antarmuka dan melakukan tindakan yang membutuhkan akses terhadap basisdata pada sisi *backend.*

3.6 Construction (Konstruksi)

Tahap *Construction* merupakan proses membuat kode (code generation). Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya pemodelan yang sebelumnya telah dibuat pada penelitian ini kemudian dibangun dengan bahasa pemrograman PHP (*Hypertext Preprocessor)* .Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat menggunakan metode *Black-box* testing dengan tujuan untuk menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

3.7 *Deployment* (Penerapan)

Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan diterapkan di rumah tahfizh. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

BAB IV
HASIL DAN PEMBAHASAN

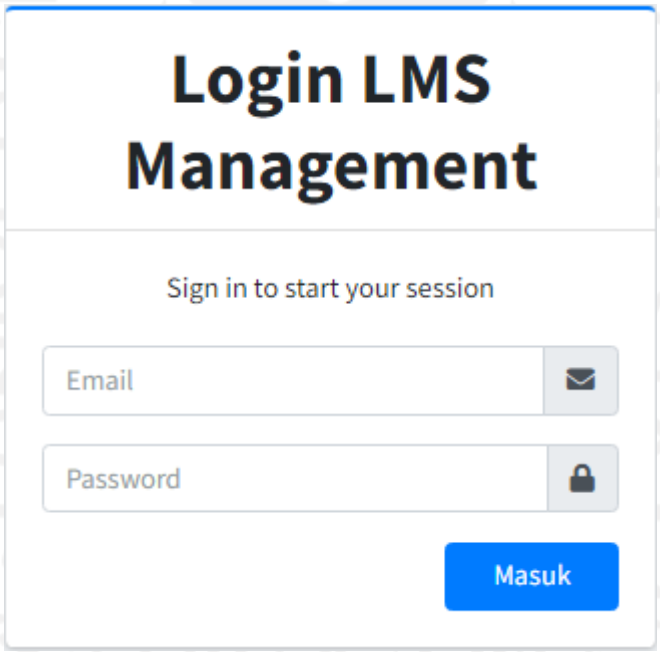
4.1 Implementasi Sistem

Implementasi sistem merupakan aktivitas atau kegiatan pasca pembuatan dan penetapan suatu kebijakan. Dalam proses ini terdapat berbagai kegiatan, seperti memimpin pekerjaan, mengatur, dan mengendalikan pekerjaan yang sedang berlangsung.

4.1.1 Admin

1) Halaman Login

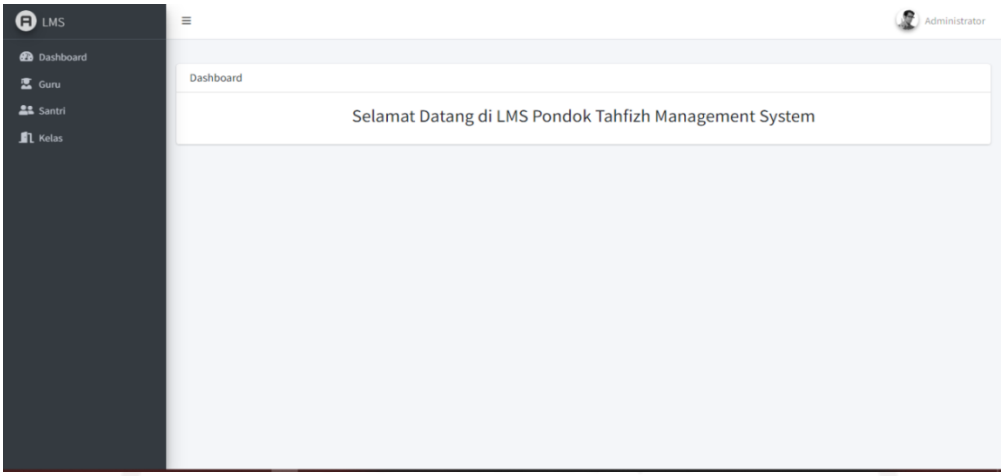
   Pada tampilan halaman login, users atau pengguna memasukkan email dan password agar dapat mengakses sistem yang dapat dilihat pada gambar 4.1.



Gambar 4.1 Halaman Login

2) Halaman *Dashboard* Admin

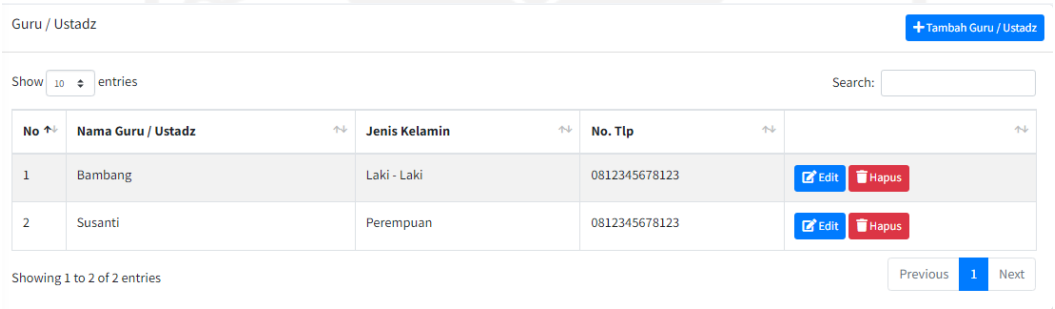
   Pada tampilan dashboard admin terdapat guru, santri, data kelas yang dapat dilihat pada gambar 4.2



Gambar 4.2 Halaman Dashboard Admin

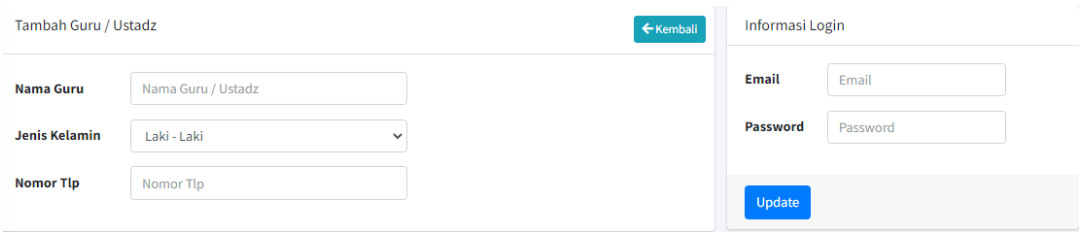
3) Halaman Guru

Pada tampilan halaman guru memiliki fitur edit data guru, tambah guru dan hapus data yang bisa dilihat pada gambar 4.3



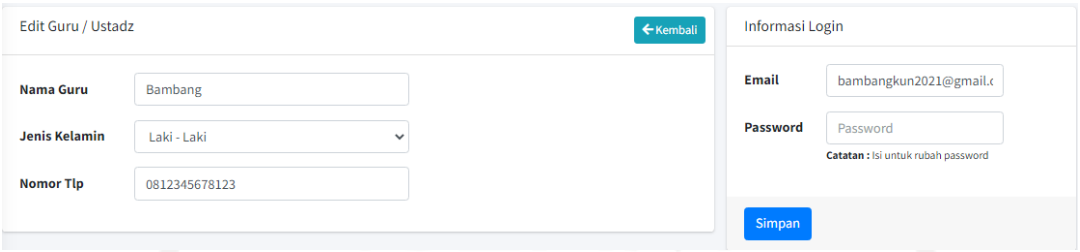
Gambar 4.3 Halaman Guru

Pada sistem terdapat fitur tambah guru/ustadz. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan nama guru, jenis kelamin, nomor telpon, email dan password bisa dilihat pada gambar 4.4



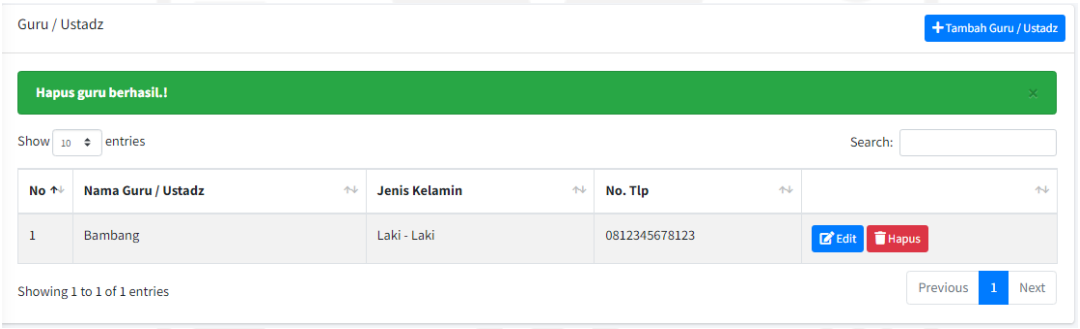
Gambar 4.4 Halaman Tambah Guru

Pada sistem terdapat fitur edit guru/ustadz. Sistem akan menampilkan kolom yang sudah terisi dan tinggal mengedit data yang akan diubah baik itu nama guru, jenis kelamin, nomor telpon, email dan password bisa dilihat pada gambar 4.5



Gambar 4.5 Halaman Edit Guru

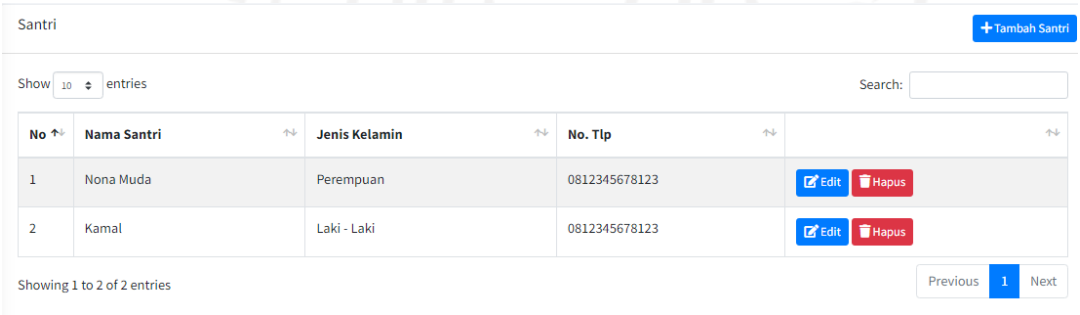
Pada sistem terdapat fitur hapus data guru. Sistem akan menampilkan pesan "Hapus guru berhasil" seperti yang terlihat pada gambar 4.6.



Gambar 4.6 Halaman Hapus data Guru

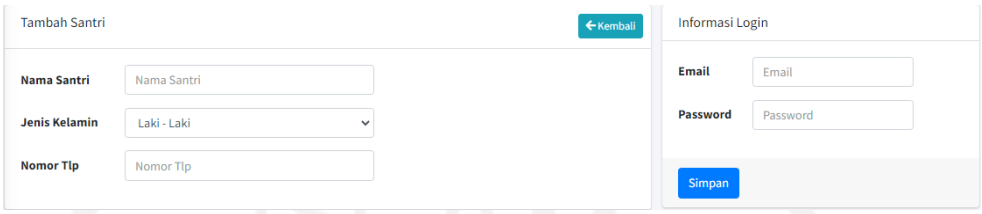
4) Halaman Santri

Pada tampilan halaman santri memiliki fitur edit data santri, tambah santri dan hapus data yang bisa dilihat pada gambar 4.7.



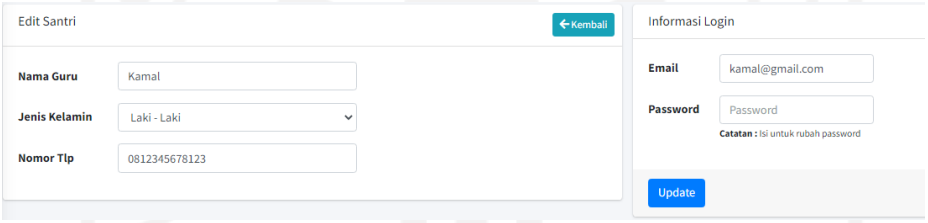
Gambar 4.7 Halaman Santri

Pada sistem terdapat fitur tambah santri. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan nama santri, jenis kelamin, nomor telpon, email dan password seperti yang terlihat pada gambar 4.8



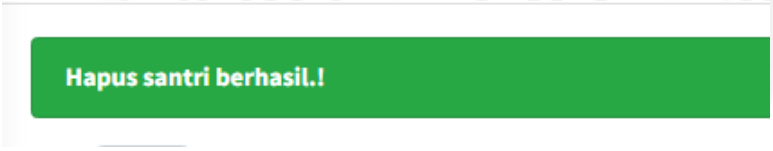
Gambar 4.8 Halaman Tambah Santri

Pada sistem terdapat fitur edit santri. Sistem akan menampilkan kolom yang sudah terisi dan tinggal mengedit data yang akan diubah baik itu nama santri, jenis kelamin, nomor telpon, email dan password seperti yang terlihat pada gambar 4.9



Gambar 4.9 Halaman Edit Santri

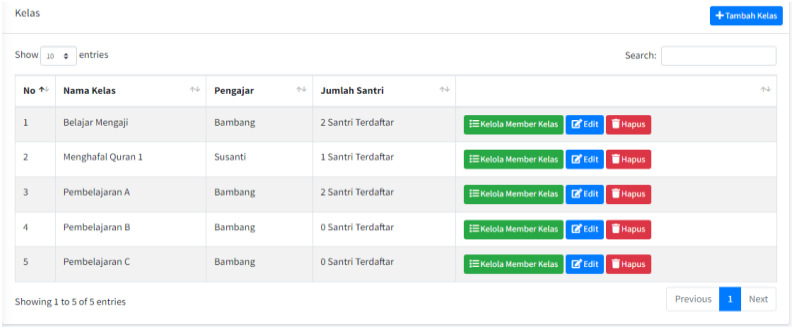
Pada sistem terdapat fitur hapus data santri. Sistem akan menampilkan pesan “Hapus santri berhasil” seperti yang terlihat pada gambar 4.10



Gambar 4.10 Halaman Hapus Santri

5) Halaman Kelas

Pada tampilan halaman kelas memiliki fitur kelola member kelas, edit data kelas, tambah kelas dan hapus data yang bisa dilihat pada gambar 4.11



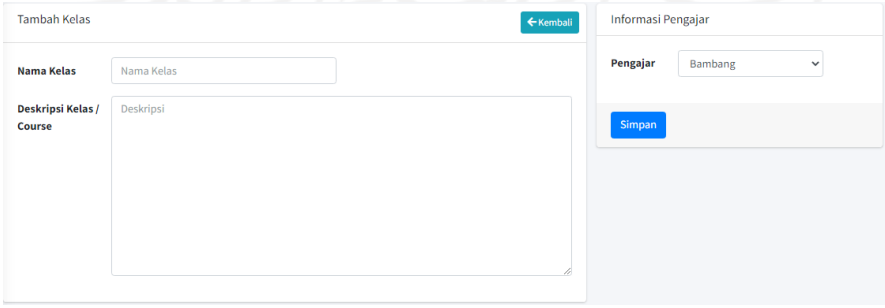
Gambar 4.11 Halaman Kelas

Pada sistem terdapat fitur kelola member kelas. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan santri belum terdaftar dan santri terdaftar seperti yang terlihat pada gambar 4.12



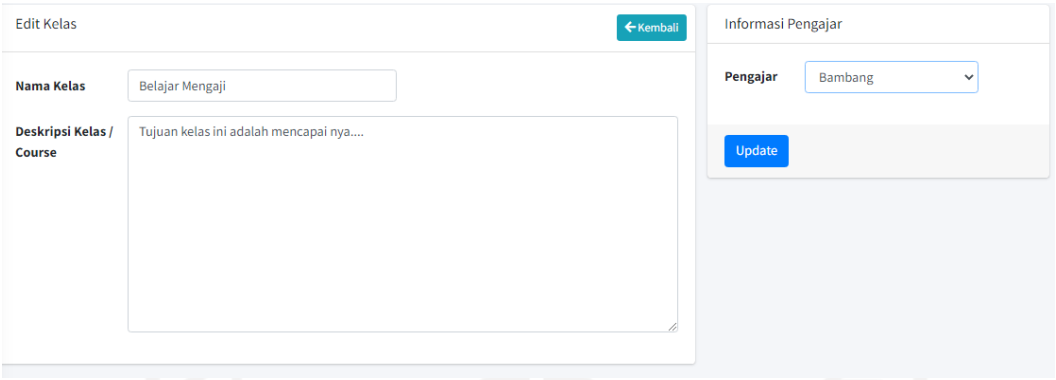
Gambar 4.12 Halaman Kelola Kelas

Pada sistem terdapat fitur tambah kelas. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan nama kelas, deskripsi kelas, dan informasi pengajar seperti yang terlihat pada gambar 4.13



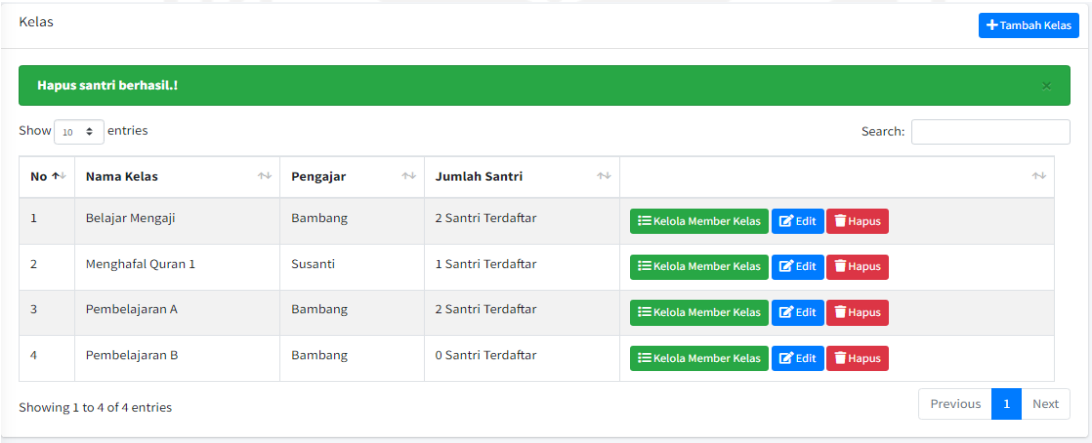
Gambar 4.13 Halaman Tambah Kelas

Pada sistem terdapat fitur edit kelas. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan nama kelas, deskripsi kelas, dan informasi pengajar seperti yang terlihat pada gambar 4.14



Gambar 4.14 Halaman Edit Kelas

Pada sistem terdapat fitur hapus data santri pada halaman kelas. Sistem akan menampilkan pesan “Hapus santri berhasil” seperti yang terlihat pada gambar 4.15

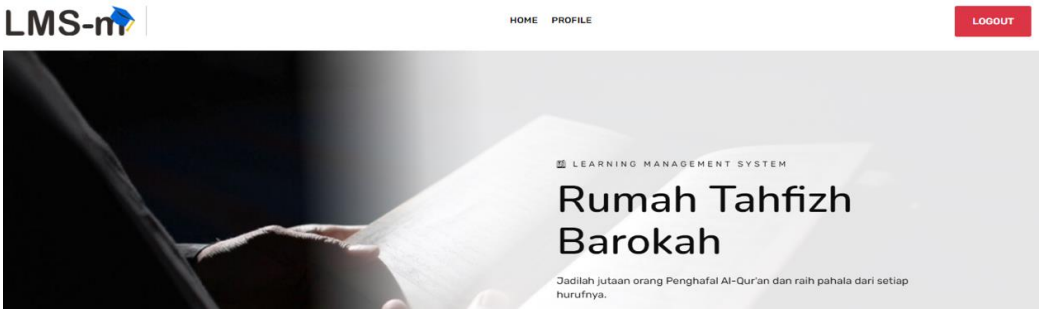


Gambar 4. 15 Halaman Hapus Data kelas

4.1.2 Guru/Ustadz

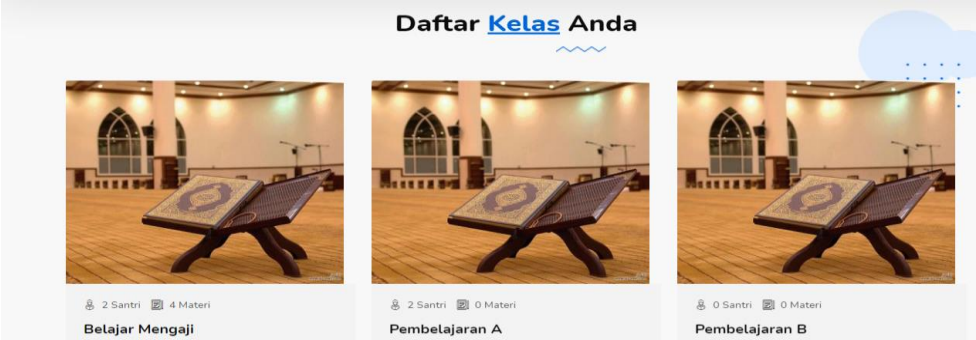
1) Halaman Dashboard

Pada tampilan dashboard guru terdapat daftar kelas yang dapat dilihat pada gambar 4.16 dan 4.17



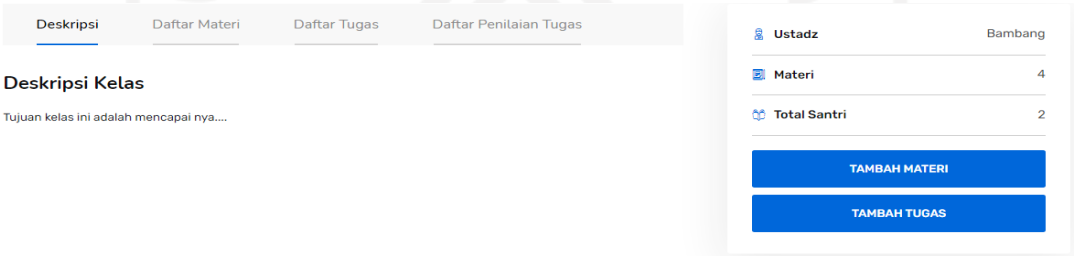
Gambar 4.16 Halaman Dashboard Guru

2) Halaman Daftar Kelas



Gambar 4.17 Halaman Fitur Daftar Kelas

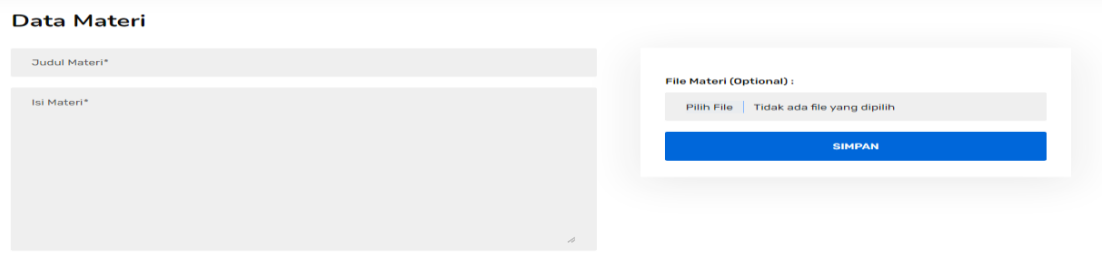
Pada tampilan dalam daftar kelas terdapat tambah materi,tambah tugas,daftar materi,daftar tugas dan daftar penilaian tugas yang dapat dilihat di gambar 4.18



Gambar 4.18 Halaman Daftar Kelas

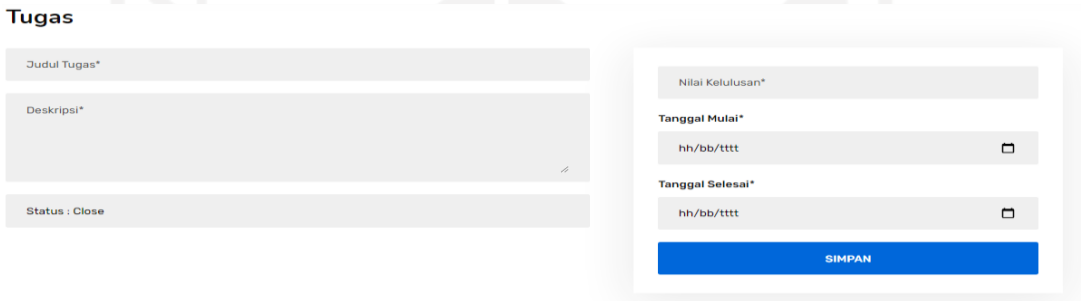
3) Halaman Daftar Materi

Pada sistem terdapat fitur tambah materi. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan data materi yaitu judul materi, isi materi dan pilih file seeprti pada gambar 4.19



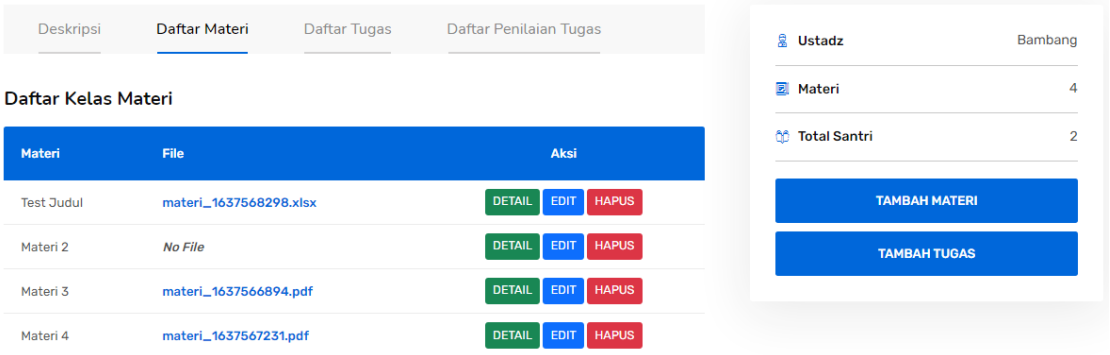
Gambar 4.19 Halaman Tambah Materi

Pada sistem terdapat fitur tambah tugas. Sistem akan menampilkan kolom kosong untuk mengisi informasi berkaitan dengan tugas yaitu judul tugas, deskripsi status, nilai kelulusan, tanggal mulai dan tanggal selesai seperti pada gambar 4.20



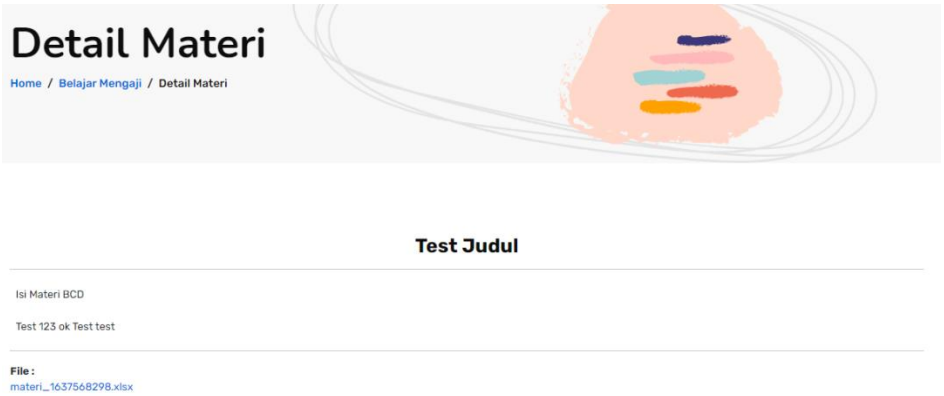
Gambar 4.20 Halaman Tambah Tugas

Pada sistem terdapat fitur daftar materi. Sistem akan menampilkan daftar kelas materi, Pada tampilan halaman daftar materi memiliki fitur edit materi,detail materi dan hapus materi yang bisa dilihat pada gambar 4.21



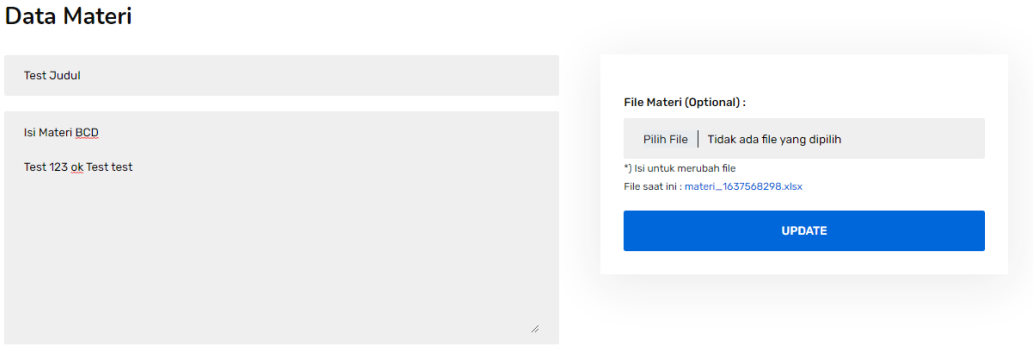
Gambar 4.21 Halaman Daftar Materi

Pada sistem terdapat fitur detail. Sistem akan menampilkan rincian materi yang berisi tentang isi materi dan file materi seperti pada gambar 4.22



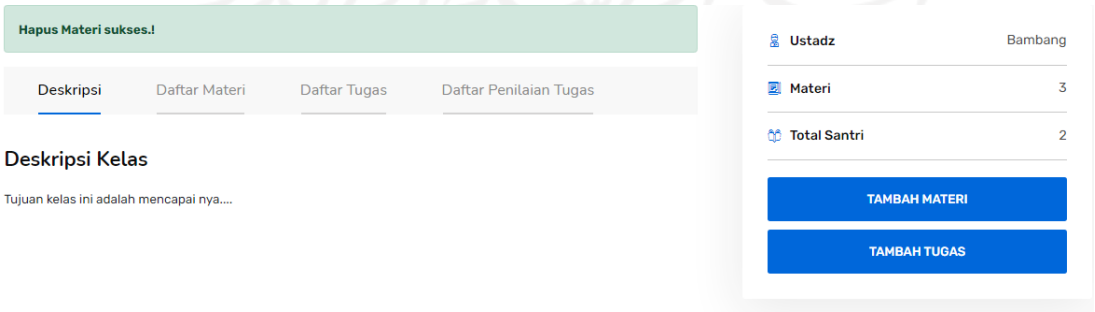
Gambar 4.22 Halaman Detail Materi

Pada sistem terdapat fitur edit materi. Sistem akan menampilkan kolom yang sudah terisi dan tinggal mengedit data materi yang akan diubah baik itu judul materi, isi materi dan file materi seperti pada gambar



Gambar 4.23 Halaman Edit Materi

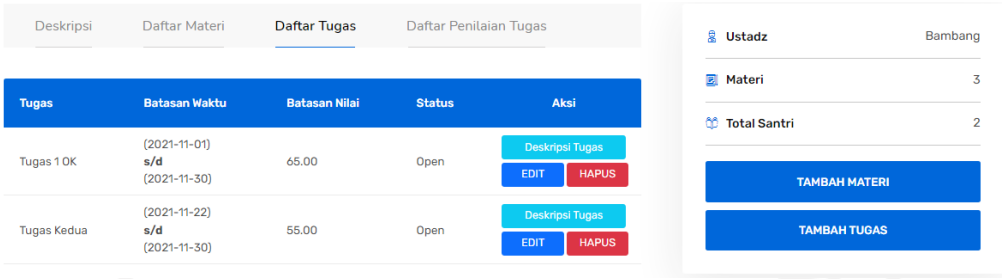
Pada sistem terdapat fitur hapus data materi. Sistem akan menampilkan pesan “Hapus materi sukses”. Seperti pada gambar 4.24



Gambar 4.24 Halaman Hapus Materi

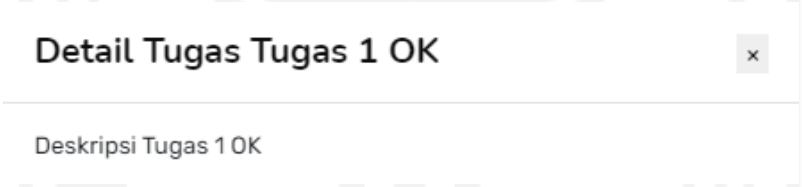
4) Halaman Daftar Tugas

Pada sistem terdapat fitur daftar tugas. Sistem akan menampilkan daftar tugas, Pada tampilan halaman daftar tugas yang memiliki fitur edit tugas,deskripsi tugas dan hapus tugas yang bisa dilihat pada gambar 4.25



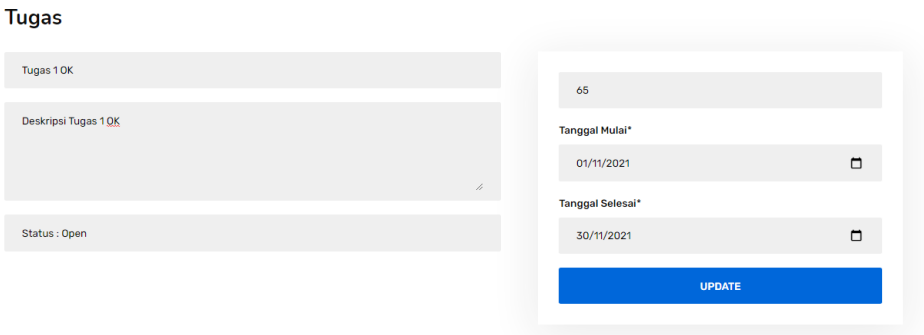
Gambar 4.25 Halaman Daftar Tugas

Pada sistem terdapat fitur deskripsi tugas. Sistem akan menampilkan detail tugas yang berisi tentang deskripsi tugas yang bisa dilihat pada gambar 4.26



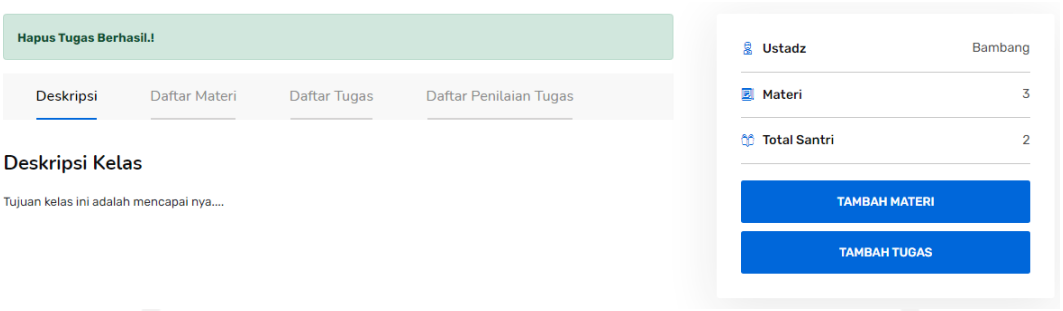
Gambar 4.26 Halaman Detail Tugas

Pada sistem terdapat fitur edit tugas. Sistem akan menampilkan kolom yang sudah terisi dan tinggal mengedit data tugas yang akan diubah baik itu judul tugas, deskripsi tugas status tugas, batasan nilai, tanggal mulai dan tanggal selesai yang bisa dilihat pada gambar 4.27



Gambar 4.27 Halaman Edit Tugas

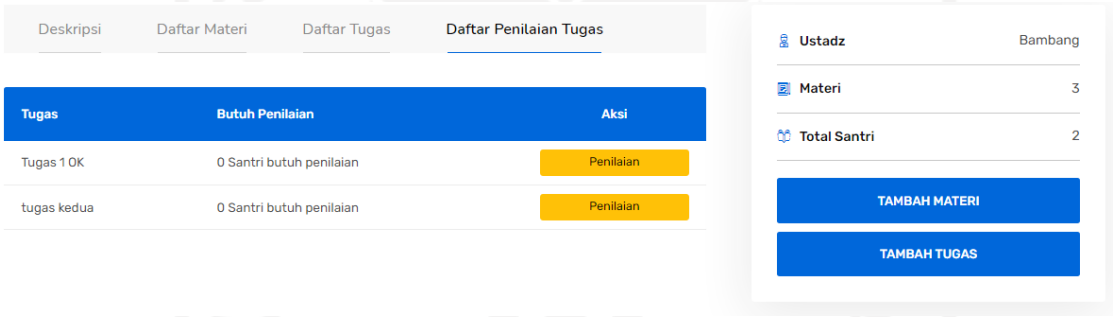
Pada sistem terdapat fitur hapus data tugas. Sistem akan menampilkan pesan “Hapus tugas sukses”. Seperti pada gambar 4.28



Gambar 4.28 Halaman Hapus Tugas

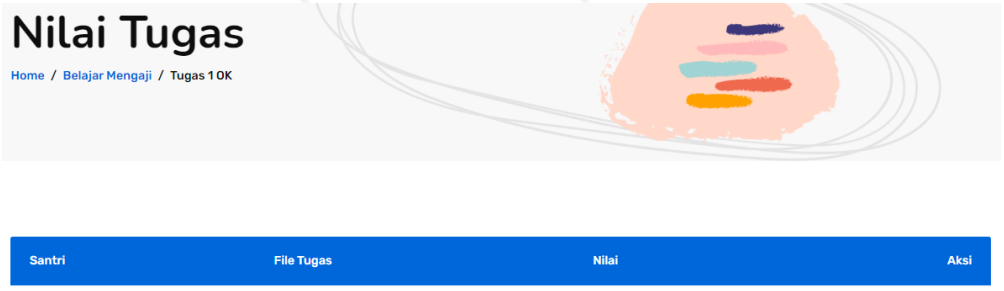
5) Halaman Daftar Penilaian Tugas

Pada sistem terdapat fitur daftar penilaian tugas. Sistem akan menampilkan keterangan tugas, butuh penilaian dan aksi Pada tampilan halaman daftar penilaian tugas yang memiliki fitur penilaian tugas yang bisa dilihat pada gambar 4.29



Gambar 4.29 Halaman Daftar Penilaian Tugas

Pada sistem terdapat fitur penilaian. Sistem akan menampilkan keterangan santri,file tugas,nilai dan aksi seperti pada gambar 4.30

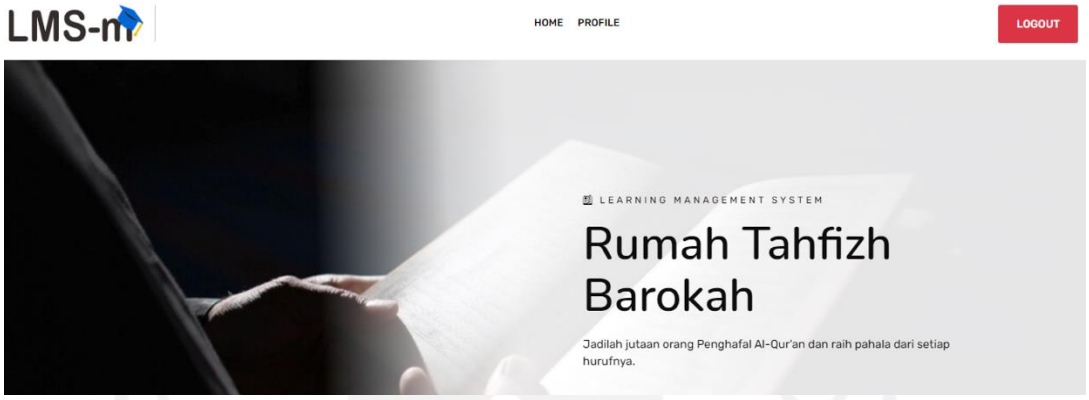


Gambar 4.30 Halaman Penilaian

4.1.3 Santri

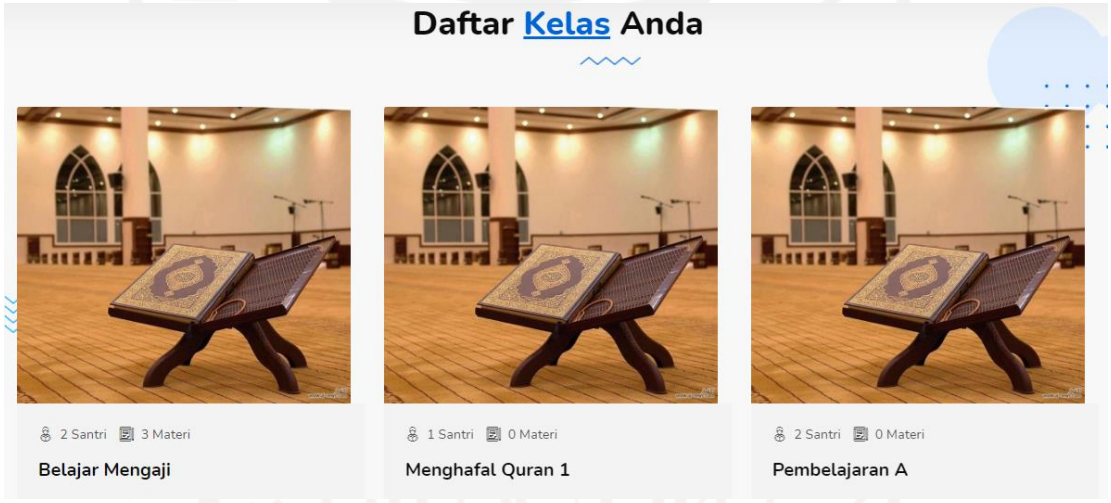
1) Halaman Dashboard

Pada tampilan dashboard santri terdapat daftar kelas yang dapat dilihat pada gambar 4.31 dan 4.32



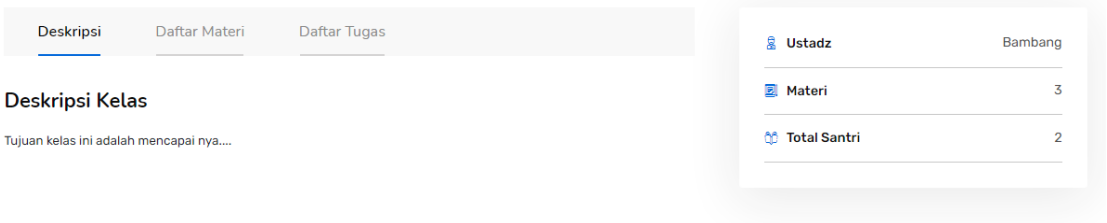
Gambar 4.31 Halaman Dashboard Santri

2) Halaman Dashboard



Gambar 4.32 Halaman Fitur Daftar Kelas

Pada tampilan dalam daftar kelas terdapat deskripsi, daftar tugas dan daftar materi yang dapat dilihat di gambar 4.33



Gambar 4.33 Halaman Deskripsi Kelas

3) Halaman Materi

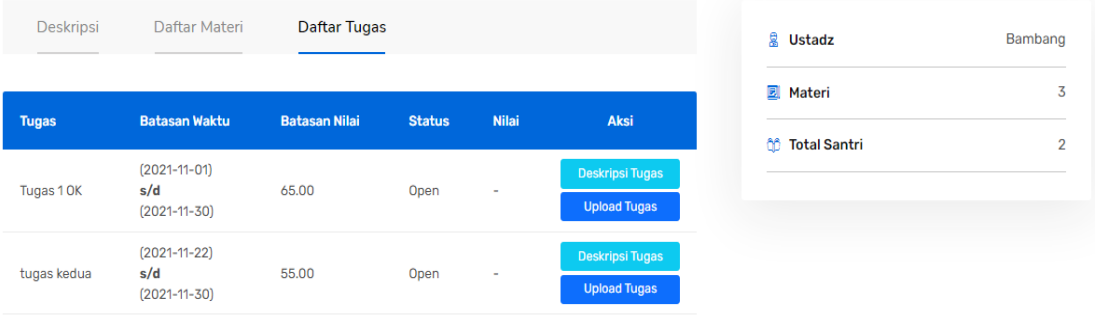
Pada sistem terdapat fitur daftar materi. Sistem akan menampilkan daftar kelas materi yang berisi nama materi dan download file materi pada gambar 4.34



Gambar 4.34 Halaman Daftar Materi

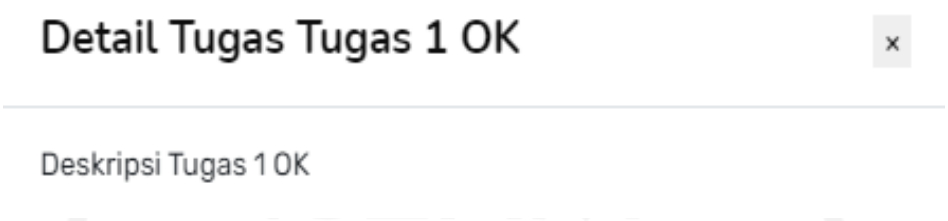
4) Halaman Daftar Tugas

Pada sistem terdapat fitur daftar tugas. Sistem akan menampilkan daftar tugas yang berisi keterangan tugas, batasan waktu, batasan nilai, status, nilai dan aksi bisa dilihat pada gambar 4.35



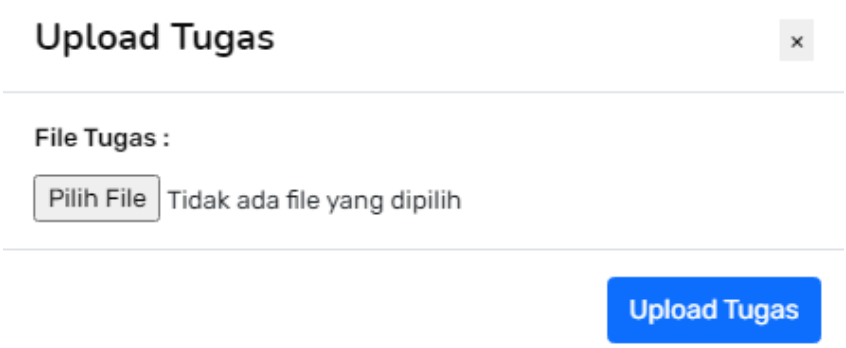
Gambar 4.35 Halaman Daftar Tugas

Pada sistem terdapat fitur deskripsi tugas dan upload tugas. Jika memilih salah satu contoh deskripsi tugas maka sistem akan menampilkan keterangan yang dapat dilihat pada gambar 4.36



Gambar 4.36 Halaman Deskripsi Tugas

Pada sistem terdapat fitur deskripsi tugas dan upload tugas. Jika memilih salah satu contoh upload tuga maka sistem akan menampilkan keterangan yang bisa dilihat pada gambar 4.37



Gambar 4.37 Halaman Upload Tugas

4.2 Pengujian

Pengujian sistem dijalankan melalui semua aktivitas di dalam sistem dengan memakai pengujian black box. Apabila ketika pengujian terdapat kesalahan maka akan di lakukan perbaikan untuk memperbaiki kesalahan.

4.2.1 Pengujian *Blackbox*

Pengujian *blackbox* di lakukan untuk menguji apakah sistem yang di buat sejalan dengan apa yang tertuang dalam spesifikasi fungsional sistem. *Blackbox* juga digunakan untuk menguji fungsi-fugsi khusus dari perangkat lunak yang di rancang kebenaran perangkat

lunak yang di uji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan fungsi yang ada tanpa melihat bagaimana proses mendapat keluaran tersebut.

Terdapat beberapa rencana pengujian *blackbox* pada tiap pengguna pada sistem:

a. Rencana Pengujian Admin yang dapat dilihat pada Tabel 4.1.
b. Rencana Pengujian Guru/Ustadz yang dapat dilihat pada Tabel 4.2.
c. Rencana Pengujian Santri yang dapat dilihat pada Tabel 4.3.

Tabel 4.1 Rencana Pengujian Admin

No	Detail Uji	Detail Pengujian	Jenis Pengujian
1	Halaman Login	Verifikasi data yang dimasukan oleh admin dengan memasukan email dan password	*Blackbox*
2	Halaman *dashboard* admin	Menguji semua fungsi yang ada pada halaman utama dan perbedaan menu yang ada pada menu admin,guru serta santri	*Blackbox*
3	Halaman guru	Menguji semua proses input dan output yang ada pada halaman kelas	*Blackbox*
4	Halaman santri	Menguji semua proses input dan output yang ada pada halaman Enroll Kelas	*Blackbox*
5	Halaman kelas	Menguji semua proses input dan output yang ada pada halaman kriteria	*Blackbox*

Tabel 4.2 Rencana pengujian Guru/Ustadz

No	Detail Uji	Detail Pengujian	Jenis Pengujian
1	Halaman Login guru	Verifikasi data yang di berikan oleh admin dengan memasukan email dan password	*Blackbox*
2	Halaman *dashboard* guru	Menguji semua fungsi yang ada pada halaman utama dan perbedaan menu yang ada pada menu admin dan santri	*Blackbox*
3	Halaman daftar kelas	Menguji semua proses input dan output yang ada pada halaman daftar kelas	*Blackbox*
4	Halaman tambah materi	Menguji semua proses input dan output yang ada pada halaman tambah materi	*Blackbox*
5	Halaman tambah tugas	Menguji semua proses input dan output yang ada pada halaman tambah tugas	*Blackbox*

6	Halaman daftar materi	Menguji semua proses input dan output yang ada pada halaman daftar materi	*Blackbox*
7	Halaman daftar tugas	Menguji semua proses input dan output yang ada pada halaman daftar materi	*Blackbox*
8	Halaman daftar penilaian tugas	Menguji semua proses input dan output yang ada pada halaman daftar materi	*Blackbox*

Tabel 4.3 Rencana Pengujian Santri

No	Detail Uji	Detail Pengujian	Jenis Pengujian
1	Halaman Login santri	Verifikasi data yang di berikan oleh admin dengan memasukan email dan password	*Blackbox*
2	Halaman dashboard santri	Menguji semua fungsi yang ada pada halaman utama dan perbedaan menu yang ada pada menu admin dan guru	*Blackbox*
3	Halaman daftar kelas	Menguji semua proses input dan output yang ada pada halaman daftar kelas	*Blackbox*
4	Halaman daftar materi	Menguji semua proses input dan output yang ada pada halaman daftar materi	*Blackbox*
5	Halaman daftar tugas	Menguji semua proses input dan output yang ada pada halaman daftar tugas	*Blackbox*

4.2.2 Kasus dan Hasil Pengujian Blackbox

Pada tahap ini di lakukan pengujian pada rencana yang telah di susun, maka dapat dilakukan pengujian sebagai berikut :

4.2.2.1 Hasil Pengujian Admin

a. Hasil pengujian *blackbox login* admin dapat dilihat pada Tabel 4.4.

Tabel 4.4 Pengujian *Blackbox Login*

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Mengosongkan Salah satu dialog email atau password	Sistem menolak dengan mengeluarkan warning Harap isi bidang ini	Valid

2	Memasukan e-mail atau password yang salah	Menampilkan pesan “email tidak ditemukan atau password yang anda masukan salah”, dan Kembali ke halaman login	Valid
3	Semua pengisian dialog diisi dengan benar	Sistem menyimpan data dan mendirect ke halaman dashboard.	Valid

b. Hasil pengujian *blackbox* halaman guru/ustadz dapat dilihat pada Tabel 4.5.

Tabel 4.5 Pengujian *Blackbox* Halaman Guru

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman Guru	Ketika admin mengakses halaman guru maka akan memunculkan menu tambah guru/ustadz, edit data guru dan hapus data guru	Valid
2	Pilih button tambah guru/ustadz	Ketika admin menekan button tambah, maka akan menampilkan form tambah guru/ustadz	Valid
3	Mengosongkan salah satu dialog pada form tambah guru/ustadz	Menampilkan pesan “harap isi kolom ini” dan data belum dapat disimpan	Valid
4	Pilih button edit	Menampilkan form edit data yang berisi data-data guru/ustadz yang mau di ubah	Valid
5	Mengosongkan salah satu dialog pada form edit	Menampilkan pesan “harap isi bidang ini” dan data belum dapat disimpan	Valid
6	Hapus data	Menampilkan pesan “apakah anda yakin?” dan menampilkan button OK dan batal. Jika pilih OK maka data guru akan terhapus, namun jika memilih cancel maka akan Kembali ke halaman guru	Valid

c. Hasil pengujian *blackbox* halaman santri dapat dilihat pada Tabel 4.6.

Tabel 4.6 Pengujian *Blackbox* Halaman Santri

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman santri	Ketika admin memilih menu santri, maka akan menampilkan tambah santri, edit dan hapus	Valid
2	Pilih button tambah santri	Ketika admin menekan button tambah santri, maka akan menampilkan form tambah data santri	
3	Mengosongkan salah satu dialog pada form tambah santri	Menampilkan pesan “harap isi bidang ini” dan data belum bisa disimpan	Valid
4	Pilih button edit	Menampilkan form edit data yang berisi data-data santri yang mau di ubah	Valid
5	Mengosongkan salah satu dialog pada form edit	Menampilkan pesan “harap isi bidang ini” dan data belum dapat disimpan	
6	Hapus data	Menampilkan pesan “apakah anda yakin?” dan menampilkan button OK dan batal. Jika pilih OK maka jadwal akan terhapus, namun jika memilih cancel maka akan Kembali ke halaman jadwal pelajaran	Valid

d. Hasil pengujian *blackbox* halaman kelas dapat dilihat pada Tabel 4.7.

Tabel 4.7 Pengujian *Blackbox* Halaman kelas

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman kelas	Ketika admin memilih menu kelas, maka akan menampilkan button kelola member kelas, edit, dan hapus	Valid
2	Pilih button kelola member kelas	Ketika admin menekan button tambah, maka akan menmunculkan form member kelas yang berisi santri belum terdaftar dan santri terdaftar	Valid
3	Memindahkan santri belum	Menampilkan pesan “Santri telah di	

	terdaftar ke form santri terdaftar	tambahkan dalam kelas.!" dan data sudah bisa disimpan	Valid
4	Pilih button edit	Menampilkan form edit yang berisi data-data kelas yang akan diubah	Valid
5	Mengosongkan salah satu dialog pada form edit	Menampilkan pesan "harap isi bidang ini" dan data belum bisa disimpan	
6	Hapus data kelas	Menampilkan pesan "apakah anda yakin?" dan menampilkan button OK dan batal. Jika pilih OK maka data akan terhapus, namun jika memilih cancel maka akan Kembali ke halaman santri	Valid

4.2.2.2 Hasil Pengujian Guru/Ustadz

a. Hasil pengujian *blackbox login* guru/ustadz dapat dilihat pada Tabel 4.8.

Tabel 4.8 Pengujian *Blackbox Login* Guru/Ustadz

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Mengosongkan Salah satu dialog email atau *password*	Sistem menolak dengan mengeluarkan *warning* Harap isi bidang ini	Valid
2	Memasukan e-mail atau *password* yang salah	Menampilkan pesan "email tidak terdaftar atau password yang anda masukan salah", dan Kembali ke halaman login	Valid
3	Semua pengisian dialog diisi dengan benar	Sistem menyimpan data dan Menuju ke halaman dashboard	Valid

b. Hasil pengujian *blackbox* halaman daftar kelas dapat dilihat pada Tabel 4.9.

Tabel 4.9 Pengujian *Blackbox* Halaman daftar kelas

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman daftar kelas	Ketika guru mengakses Halaman daftar kelas maka akan memunculkan menu tambah materi, tambah tugas serta beberapa fitur di dalam nya seperti daftar materi, daftar tugas dan daftar penilaian tugas	Valid

2	Pilih button tambah materi	Ketika guru menekan button tambah materi, maka akan menampilkan form tambah materi	Valid
3	Mengosongkan salah satu dialog pada form tambah materi	Menampilkan pesan “harap isi bidang ini” dan data belum dapat disimpan	Valid
4	Pilih button tambah tugas	Ketika guru menekan button tambah tugas, maka akan menampilkan form tambah tugas	Valid
5	Mengosongkan salah satu dialog pada form tambah tugas	Menampilkan pesan “harap isi bidang ini” dan data belum bisa disimpan	Valid

c. Hasil pengujian *blackbox* halaman daftar materi dapat dilihat pada Tabel 4.10.

Tabel 4.10 Pengujian *Blackbox* Halaman daftar materi

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman daftar materi	Ketika guru memilih fitur daftar materi, maka akan menampilkan button detail, edit, dan hapus	Valid
2	Pilih button detail	Ketika guru menekan button detail, maka akan memunculkan judul materi, isi materi serta file materi	Valid
3	Pilih button edit	Menampilkan form edit materi yang berisi data-data materi yang mau di ubah	Valid
4	Mengosongkan salah satu dialog pada form edit materi	Menampilkan pesan “harap isi bidang ini” dan data belum bisa disimpan	valid
5	Hapus materi	Menampilkan pesan “apakah anda yakin?” dan menampilkan button OK dan batal. Jika pilih OK maka materi akan terhapus, namun jika memilih cancel maka akan Kembali ke halaman daftar materi	Valid

d. Hasil pengujian *blackbox* daftar tugas dapat dilihat pada Tabel 4.11.

Tabel 4.11 Pengujian *Blackbox* Halaman daftar tugas

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman daftar tugas	Ketika guru memilih fitur daftar tugas, maka akan menampilkan button deskripsi tugas, edit, dan hapus	Valid
2	Pilih button deskripsi tugas	Ketika guru menekan button deskripsi tugas, maka akan memunculkan keterangan deskripsi tugas	Valid
3	Pilih button edit	Menampilkan form edit tugas yang berisi data-data tugas yang mau di ubah	Valid
4	Mengosongkan salah satu dialog pada form edit materi	Menampilkan pesan “harap isi bidang ini” dan data belum bisa disimpan	valid
5	Hapus materi	Menampilkan pesan “apakah anda yakin?” dan menampilkan button OK dan batal. Jika pilih OK maka materi akan terhapus, namun jika memilih cancel maka akan Kembali ke halaman daftar tugas	Valid

e. Hasil pengujian *blackbox* halaman daftar penilaian tugas dapat dilihat pada Tabel 4.12.

Tabel 4.12 Pengujian *Blackbox* Halaman daftar penilaian tugas

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman daftar penilaian tugas	Ketika guru memilih fitur daftar penilaian tugas, maka akan menampilkan keterangan tugas,butuh penilaian button penilaian, button tambah materi dan button tambah tugas	Valid
2	Pilih button penilaian	Ketika guru menekan button penilaian, maka akan memunculkan keterangan nilai tugas yang berisi santri, file tugas, nilai dan aksi	Valid
3	Pilih button tambah materi	Ketika guru menekan button tambah materi, maka akan menampilkan form tambah materi	Valid
4	Mengosongkan salah satu	Menampilkan pesan “harap isi bidang ini” dan	valid

	dialog pada form tambah materi	data belum bisa disimpan	
5	Pilih button tambah tugas	Ketika guru menekan button tambah tugas, maka akan menampilkan form tambah tugas	
6	Mengosongkan salah satu dialog pada form tambah tugas	Menampilkan pesan “harap isi bidang ini” dan data belum bisa disimpan	valid

4.2.2.3 Hasil Pengujian Santri

a. Hasil pengujian *blackbox login* santri dapat dilihat pada Tabel 4.13.

Tabel 4.13 Pengujian *Blackbox Login* Santri

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Mengosongkan Salah satu dialog email atau password	Sistem menolak dengan mengeluarkan warning Harap isi bidang ini	Valid
2	Memasukan e-mail atau password yang Salah	Menampilkan pesan “email tidak terdaftar atau password yang anda masukan salah”, dan Kembali ke halaman login	Valid
3	Semua pengisian dialog diisi dengan benar	Sistem menyimpan data dan mendirect ke halaman dashboard	Valid

b. Hasil pengujian *blackbox* daftar tugas dapat dilihat pada Tabel 4.14.

Tabel 4.14 Pengujian *Blackbox* Halaman daftar kelas santri

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman daftar kelas	Ketika santri mengakses Halaman daftar kelas santri maka akan memunculkan menu daftar materi dan daftar tugas	Valid
2	Pilih menu daftar materi	Ketika santri memilih menu daftar materi, maka akan menampilkan keterangan materi dan file	Valid
3	Pilih menu daftar tugas	Ketika santri memilih menu daftar tugas, maka akan menampilkan keterangan tugas, batasan waktu, batasan nilai, status, nilai dan aksi	Valid

c. Hasil pengujian *blackbox* daftar tugas dapat dilihat pada Tabel 4.15.

Tabel 4.15 Pengujian *Blackbox* Halaman daftar tugas

No	Deskripsi Pengujian	Hasil yang di harapkan	Kesimpulan
1	Halaman daftar tugas	Ketika santri mengakses Halaman daftar tugas santri maka akan memunculkan keterangan tugas, batasan waktu, batasan nilai, status, nilai dan aksi	Valid
2	Halaman nilai tugas	Ketika santri mengakses Halaman nilai tugas santri maka akan memunculkan keterangan tugas, batasan waktu, batasan nilai, status dan nilai	Valid

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan wawancara, observasi, studi pustaka, perancangan, implementasi dan pengujian sistem yang telah dilakukan, penulis menyimpulkan bahwa:

1. Berhasil membangun sebuah sistem informasi yang dapat digunakan sebagai sarana untuk kegiatan pembelajaran antara para santri/santriwati dengan pengajar selama diterapkannya kebijakan dimasa pandemi.
2. Terancang nya sebuah sistem informasi berupa *Learning Management System (LMS)* berbasis web dengan memakai metode *Service Oriented Architecture (SOA)* menggunakan metode pengembangan *Waterfall.*

5.2 Saran

Terdapat saran yang diusulkan terhadap pengembangan *learning management system* rumah/pondok tahfizh dengan *service oriented architecture* untuk tahapan pengembangan selanjutnya adalah:

1. Dalam sistem ini akan lebih baik apabila ditambahkan actor orang tua santri agar dapat melihat perkembangan anak nya ketika sedang belajar tahfiz Qur’an dan jika ada beberapa hal yang ingin ditanyakan soal seputar materi pembelajaran.
2. Tampilan UI/UX bisa diperbarui lagi agar terlihat nyaman dan mudah digunakan, baik oleh santri maupun guru/ustadz.
3. Sistem dapat dikembangkan dari berbasis web juga menjadi berbasis android agar lebih flexibel.

DAFTAR PUSTAKA

Hendri, Manurung, J. W., Ferian, R. A., Hanaatmoko, W. F., & Yulianti, Y. (2020). Pengujian Black Box pada Aplikasi Sistem Informasi Pengelolaan Masjid Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi, Vol. 3, No. 2, E-ISSN: 2654-4229*, 107-113.

Wiliani, N., & Zambi, S. (2017). RANCANG BANGUN APLIKASI KASIR TIKET NONTON BOLA BARENG PADA X KASIR DI SUATU LOKASI X DENGAN VISUAL BASIC 2010 DAN MYSQL. *Jurnal Rekayasa Informasi, Vol. 6, No. 2, E-ISSN : 2252-7354* , 77-83.

Ruseno, N. (2019). IMPLEMENTASI SCRUM PADA PENGEMBANGAN APLIKASI SISTEM RESERVASI ONLINE MENGGUNAKAN PHP. *JURNAL GERBANG, Vol. 9, No. 1*, 8-15.

Kurniawan, H., Apriliah, W., Kurniawan, I., & Firmansyah, D. (2020). PENERAPAN METODE WATERFALL DALAM PERANCANGAN SISTEM INFORMASI PENGGAJIAN PADA SMK BINA KARYA KARAWANG. *Jurnal Interkom, Vol. 14, No. 4*, 13-23.

Tabrani, M., & Aghniya, I. R. (2019). IMPLEMENTASI METODE WATERFALLPADA PROGRAM SIMPAN PINJAM KOPERASI SUBUR JAYA MANDIRI SUBANG. *Jurnal Interkom, Vol. 14, No. 1*, 44-53.

Marbun, M., & Sinaga, B. (2019). *BUKU AJAR SISTEM PENDUKUNG KEPUTUSAN PENILAIAN HASIL BELAJAR DENGAN METODE TOPSIS.* Medan, Sumatera Utara: CV.Rudang Mayang.

Sianipar, R. H. (2015) *Jquery & Ajax untuk Web Designer*. 1st edn. yogyakarta: Andi Offset.

Syarif, M., & Nugraha, W. (2020). PEMODELAN DIAGRAM UML SISTEM PEMBAYARAN TUNAI PADA TRANSAKSI E-COMMERCE. *Jurnal Teknik Informatika Kaputama (JTIK), Vol. 4, No. 1, E-ISSN : 2686-0880,* 64-70.

Yauma, A., Fitri, I., & Ningsih, S. (2021). Learning Management System (LMS) pada E-Learning Menggunakan Metode Agile dan Waterfall berbasis Website. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), Vol. 5, No. 3*, 323-328.

Fitriyah, I. M., & Ghofur, M. A. (2021). Pengembangan E-LKPD Berbasis Android dengan Model Pembelajaran Problem Based Learning (PBL) untuk Meningkatkan Berpikir

Kritis Peserta Didik. *EDUKATIF : JURNAL ILMU PENDIDIKAN, Vol. 3, No. 5, E-ISSN : 2656-8071*, 1957-1970.

Mahardhika, E. B. (2013). Peran Rumah Tahfidz Zulfa Qurrata'ayun dalam Pemberdayaan Masyarakat di Desa Purbayan Kota Gede Yogyakarta. *Skripsi. UIN Sunan Kalijaga Yogyakarta*.

Khoirudin, Daru, A. F., & Nugroho, A. (2019). Hybrid Mobile Application Dengan Metode Service Oriented Architecture. *JURNAL INFORMATIKA UPGRIS, Vol. 5, No. 1, E-ISSN : 2447-6645*, 17-24.