

SISTEM MONITORING TAHFIDZUL QUR'AN BERBASIS WEB

**Azril Delfrian Adi Putra; Aris Rakhmadi, S.T., M.Eng.
Program Studi Teknik Informatika, Fakultas Komunikasi dan
Informatika, Universitas Muhammadiyah Surakarta**

Abstrak

Kemajuan perkembangan teknologi membawa inovasi kedalam kegiatan pendidikan salah satunya adalah kegiatan Tahfidzul Qur'an di Pondok Pesantren, yang dimana kegiatan ini masih terdapat beberapa hal yang mempengaruhi keterhambatan seperti keterbatasan waktu dan guru sehingga diperlukannya suatu sistem yang optimal untuk membantu melancarkan keberlangsungan kegiatan ini. Dalam penelitian ini penulis bertujuan untuk mengembangkan sebuah Sistem Monitoring Tahfidzul Qur'an dengan memanfaatkan metode Agile. Sistem ini dirancang untuk memudahkan proses penyeteroran hafalan santri secara modern dan membantu ustad dalam memantau perkembangan santri melalui riwayat hafalan yang terdokumentasi secara digital. Pengembangan sistem ini menggunakan teknologi PHP 8 sebagai bahasa pemrograman utama, dengan dukungan dari framework Laravel versi 10 untuk mempercepat proses pengembangan. Phpmyadmin/mysql digunakan sebagai alat penyimpanan data dalam bentuk database, dan XAMPP digunakan sebagai platform pengembangan lokal. Pengujian sistem dilakukan dengan metode *black box* dengan hasil keluaran sistem sesuai dengan tujuan. Hasil uji *System Usability Scale* dari 30 responden dengan 3 peran yaitu admin, ustad, dan santri mendapatkan skor 78.8 dengan tingkat penerimaan dapat diterima.

Kata Kunci: Sistem Monitoring, Tahfidzul Quran, Agile.

Abstract

Advances in technological development have brought innovation into educational activities, one of which is the Tahfidzul Qur'an activity at the Islamic Boarding School, where this activity still has several things that influence obstacles such as limited time and teachers, so an optimal system is needed to help smooth the continuity of this activity. In this research the author aims to develop a Tahfidzul Qur'an Monitoring System using the Agile method. This system is designed to facilitate the process of depositing students' memorization in a modern way and assist ustad in monitoring students' progress through digitally documented memorization history. This system development uses PHP 8 technology as the main programming language, with support from the Laravel version 10 framework to speed up the development process. Phpmyadmin/mysql is used as a data storage tool in the form of a database, and XAMPP is used as a local development platform. System testing is carried out using the black box method with system output results in accordance with the objectives. The System Usability Scale test results from 30 respondents with 3 roles, namely admin, ustad, and santri, got a score of 78.8 with an acceptable acceptance level.

Keywords: Monitoring System, Tahfidzul Quran, Agile.

1. PENDAHULUAN

Al-Qur'an merupakan kitab bagi umat Islam sebagai sumber petunjuk dalam beragama dan pembimbing dalam menjalani kehidupan di dunia dan akhirat. Al-Qur'an adalah kalamullah, yaitu firman Allah yang diturunkan oleh nabi Muhammad selama 23 tahun lamanya. Sehingga merupakan suatu kewajiban bagi seorang muslim untuk selalu berinteraksi aktif dengan al-Qur'an,

menjadikannya sebagai sumber inspirasi, berpikir dan bertindak (Gazali, 2010).

Tahfidzul Qur'an adalah kegiatan menghafal ayat-ayat Al-Qur'an sesuai dengan pedoman tajwid, tajwid merupakan aturan-aturan atau hukum-hukum cara membaca ayat Al-Qur'an. Hukum-hukum ini merupakan pedoman dalam membaca Al-Quran. Lalu juga terdapat Makhraj, menurut bahasa Makhraj merupakan tempat keluarnya huruf, Sedangkan menurut istilah, Makhraj adalah suatu nama tempat, yang pada huruf dibentuk (diucapkan) (Syah et al., 2018).

Namun saat ini kegiatan tahfidz qur'an di Indonesia yang terbilang masih banyak menggunakan metode konvensional dengan sistem Talaqqi. Sistem Talaqqi yaitu dengan murid melakukan setoran hafalan langsung ke guru secara tatap muka, sehingga metode tersebut mempunyai hambatan berupa keterbatasan guru dan waktu (Assisi et al., 2022). Dalam upaya untuk memajukan pendidikan Tahfidzul Qur'an, Pondok Pesantren perlu memanfaatkan teknologi untuk mengembangkan Sistem Monitoring Tahfidzul Qur'an berbasis web. Sistem ini diharapkan dapat membantu pemantauan dan evaluasi kemajuan hafalan Al-Qur'an siswa dengan lebih efisien, sekaligus memudahkan akses ke sumber daya Al-Qur'an yang diperlukan dalam proses pembelajaran.

Tujuan dari pengembangan sistem monitoring ini adalah sebagai wadah alternatif untuk memudahkan murid yang ingin menyetorkan hafalan tanpa harus menggunakan metode lama, melakukan pengelolaan data dengan lebih cepat. Pada pengembangan sistem ini diperuntukkan agar dapat memberi manfaat memudahkan dalam pelaksanaan setoran hafalan Al-Qur'an tanpa harus mengantri untuk menunggu giliran yang menghabiskan waktu lebih banyak dan memudahkan dalam manajemen pengelolaan data. Sebelumnya telah dilakukan penelitian oleh Suryana (2021) yang membuat sistem serupa dengan fitur *memorization data*. Perbedaannya dengan penelitian sebelumnya dengan saat ini adalah penambahan fitur aktor ustad dan santri dimana tiap aktor dapat melakukan *login* dan melakukan kegiatan setoran dan *review* tahfidzul quran. Kemudian fitur halaqoh untuk mengatur santri sesuai halaqoh.

2. METODE

Metode digunakan untuk mencapai tujuan dengan cara yang terorganisir dengan pedoman metode yang digunakan sehingga dapat meraih hasil yang ingin dicapai. Pengumpulan data dilakukan dengan melakukan observasi terhadap pondok pesantren dimana melakukan kegiatan tahfidzul quran secara konvensional, kemudian penulis melakukan identifikasi kebutuhan untuk merancang sistem monitoring.

Agile Development Methodology

Seiring dengan evolusi teknologi dan dinamika tuntutan pasar yang semakin kompleks, organisasi pengembangan perangkat lunak menghadapi tantangan dalam memilih model SDLC yang paling tepat untuk memenuhi kebutuhan proyek secara efisien dan efektif. Untuk membuat keputusan yang

terinformasi, pengembang, manajer proyek, dan insinyur kualitas perlu memahami secara komprehensif kelebihan, kelemahan, dan kesesuaian aplikasi dari berbagai model SDLC dalam konteks rekayasa kualitas perangkat lunak (Pargaonkar, 2023). Terdapat lima tahap Agile menurut Naufal Faruq & Maryam (2024) yaitu: Perencanaan, Desain, Implementasi, Pengujian, dan Penyebaran.

2.1 Perencanaan

2.1.1 Analisis Kebutuhan

Fase analisis kebutuhan/*requirement analysis* adalah langkah awal dalam pengembangan perangkat lunak yang bertujuan untuk mengumpulkan informasi terkait dengan kebutuhan yang diperlukan sebagai acuan pengembangan perangkat lunak. Ini melibatkan wawancara, pemetaan proses, validasi kebutuhan, dan hasilnya adalah dokumen spesifikasi yang akan menjadi panduan bagi tim pengembangan. Pada tahapan ini terdapat dua jenis analisis kebutuhan: fungsional dan non-fungsional.

Dalam penelitian yang dilakukan di Pondok Pesantren Serambi Al-Aqsho Karanganyar, fase analisis kebutuhan menjadi kunci dalam memahami kebutuhan yang spesifik untuk pengembangan perangkat lunak yang sesuai dengan konteks pesantren tersebut. Dalam konteks ini, analisis kebutuhan fungsional akan mengidentifikasi fungsi-fungsi yang harus ada dalam perangkat lunak, seperti fitur-fitur untuk manajemen hafalan santri dan evaluasi oleh ustad. Sementara itu, analisis kebutuhan non-fungsional akan menyoroti aspek-aspek seperti ketersediaan dan keamanan sistem, yang sangat penting untuk menjaga kualitas dan integritas perangkat lunak yang dikembangkan.

2.1.2 Kebutuhan Fungsional

Kebutuhan fungsional untuk platform ini meliputi kebutuhan dari berbagai pihak terlibat, mulai dari santri hingga admin. Bagi santri, platform harus menyediakan halaman setoran hafalan yang memungkinkan mereka untuk mengunggah atau mencatat hafalan berbagai format audio mp3 dikarenakan format yang jernih dan tidak terlalu berat, serta menampilkan riwayat hafalan untuk memantau perkembangan mereka. Alasan tidak memakai format video dikarenakan keterbatasan pada penyimpanan dan pemantauan tidak selalu dilakukan dengan sistem. Untuk ustad, diperlukan halaman review hafalan yang memungkinkan mereka untuk menilai hafalan santri dan memberikan saran atau catatan tambahan untuk memperbaiki hafalan tersebut, serta dapat melihat daftar santri yang diampunya berdasarkan halaqoh. Bagi admin, penting untuk memiliki kemampuan manajemen akun pengguna, termasuk membuat, menghapus, atau mengelola akun santri dan ustad, serta manajemen data hafalan seperti pembuatan dan pengelolaan kelas atau halaqoh santri dan ustad.

Sementara itu, kebutuhan bersama mencakup halaman *login* yang diperlukan oleh semua pengguna untuk mengakses akun mereka, serta halaman utama atau *dashboard* yang menyajikan informasi

sekilas hafalan. Di sisi lain, kebutuhan non-fungsional mencakup aspek perangkat keras dan lunak yang diperlukan untuk mengoperasikan platform ini, termasuk komputer dan *smartphone* sebagai perangkat keras, serta akses internet dan *browser* sebagai perangkat lunak yang harus tersedia untuk penggunaan yang lancar dan efektif.

2.2 Desain

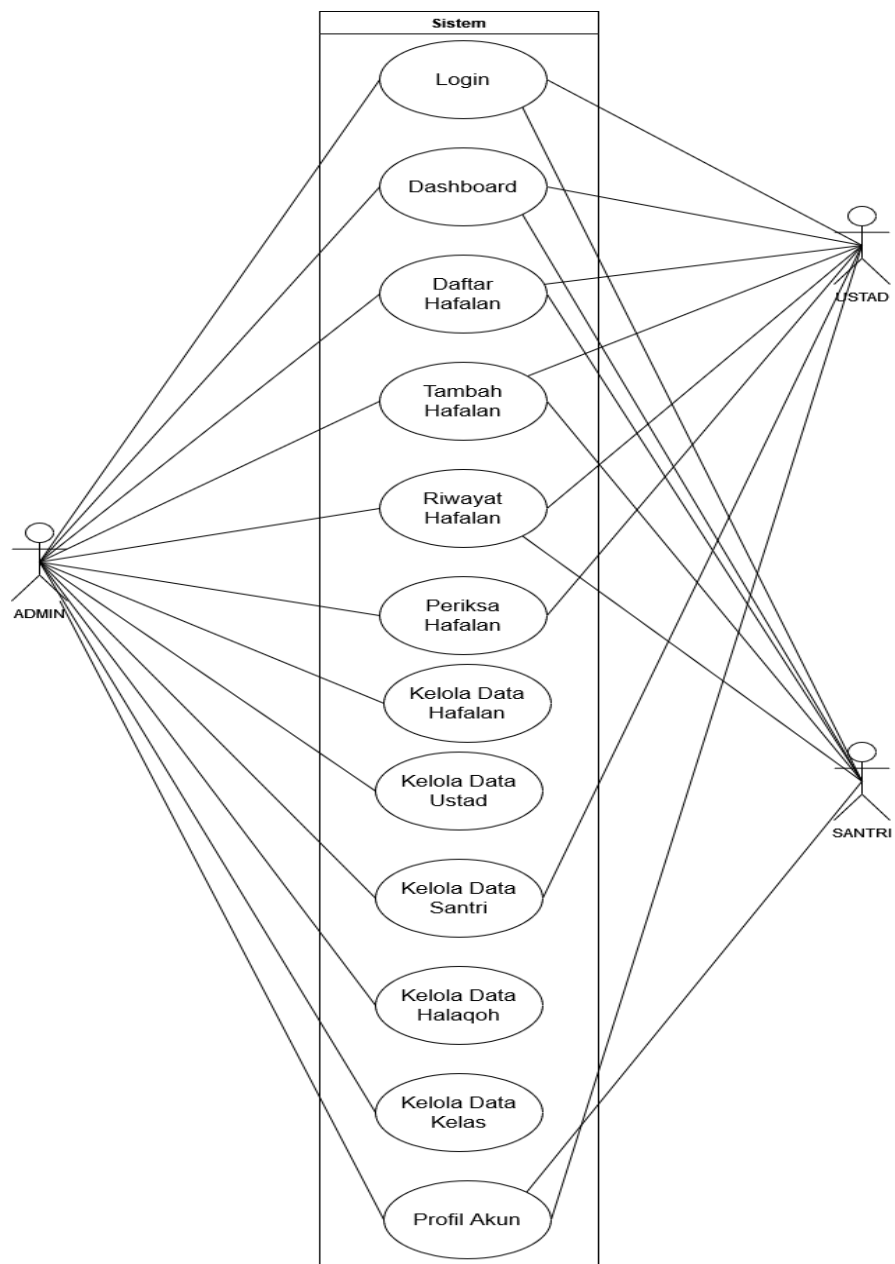
Perancangan sistem ini menggunakan *Unified Modeling Language* (UML) adalah bahasa standar yang banyak digunakan untuk mendefinisikan persyaratan, menganalisis dan desain, serta menjelaskan arsitektur pemrograman berorientasi objek dalam dunia industri. UML adalah bahasa pemodelan komputer dan komunikasi visual yang menggunakan diagram dan dukungan teks (Rambe et al., 2020). UML memungkinkan komunikasi "apa" yang dibutuhkan oleh sistem dan "bagaimana" sistem dapat direalisasikan, memungkinkan pemodelan visual sebelum implementasi, memberikan panduan seperti "*blueprint*" dalam konstruksi sistem, dan mendokumentasikan pengetahuan sistem sepanjang siklus hidupnya.

2.2.1 Use Case Diagram

Use case diagram adalah alat visualisasi yang penting dalam pengembangan sistem informasi, yang menggambarkan seluruh aktivitas yang dilakukan oleh sistem dari sudut pandang pengamatan eksternal. Pramasari & Nurgiyatna (2019) menjelaskan bahwa *use case diagram* merupakan representasi dari sekelompok aktivitas yang terhubung dan membentuk sistem secara runtut, yang diawasi oleh aktor-aktor tertentu.

Gambar 1 menampilkan *use case diagram* yang mencakup tiga aktor utama: Admin, Ustad, dan Santri. Admin, sebagai pemegang hak akses tertinggi, memiliki wewenang untuk melakukan pengelolaan data secara menyeluruh dalam sistem. Selanjutnya, pada *use case diagram* yang sama, terdapat aksi-aksi yang dapat dilakukan oleh aktor Ustad, seperti melihat data hafalan, menambah data hafalan, memeriksa hafalan, mengelola data santri, dan melihat riwayat hafalan dari santri yang diampunya. Ustad memiliki kemampuan untuk memeriksa dan melihat data hafalan santri yang sesuai dengan wilayah tanggung jawabnya, serta melakukan tindakan lainnya seperti penambahan data atau pemeriksaan riwayat berdasarkan halaqoh yang diampunya.

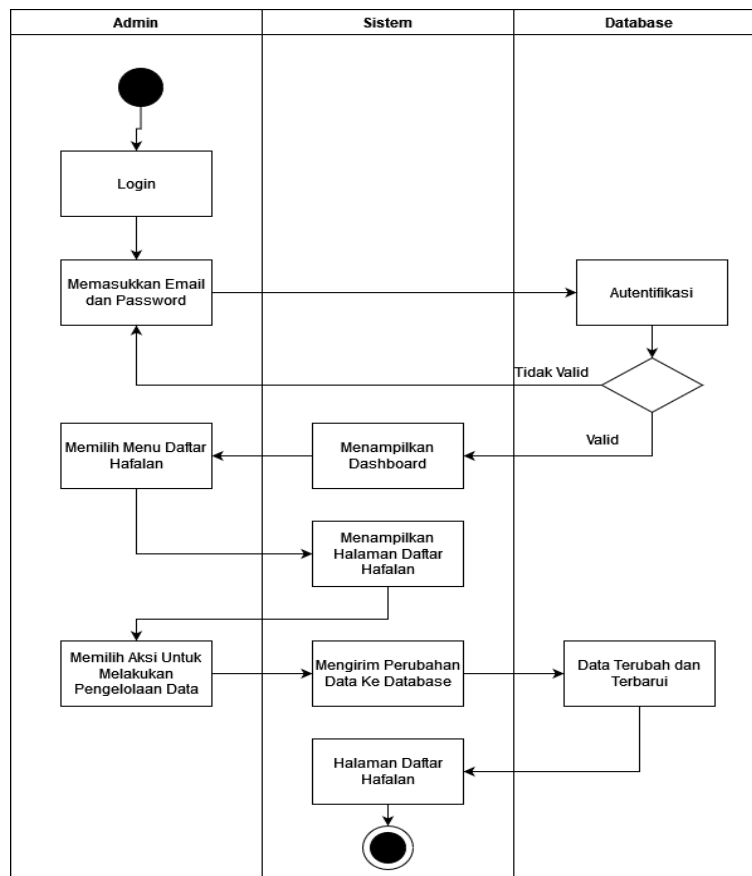
Sementara itu, *use case* dari aktor Santri memungkinkan santri untuk melakukan penambahan data hafalan dan mengirimkannya ke dalam sistem. Santri juga memiliki kemampuan untuk melihat riwayat hafalannya sendiri, serta mengunduh hasil laporan hafalan. Masing-masing dari aktor memiliki akses ke Profil untuk mengubah data diri seperti *password*. *Use Case Diagram* ini disesuaikan menurut Pondok Pesantren Serambi Al-Aqsho Karanganyar dengan sebaik-baiknya dari hasil wawancara penulis.



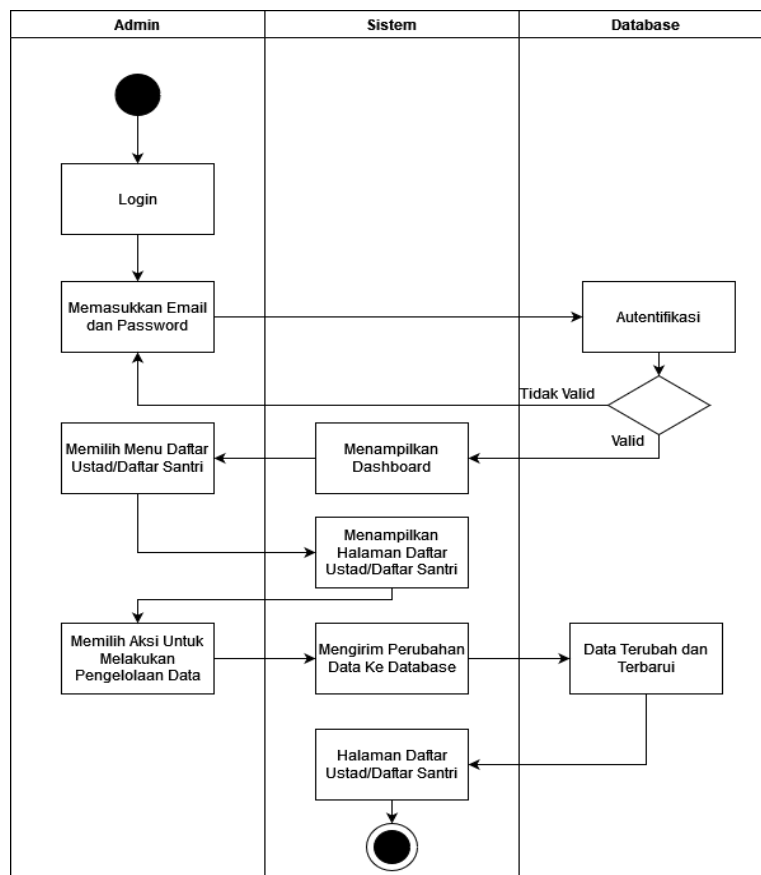
Gambar 1. Use Case Diagram

2.2.2 Activity Diagram

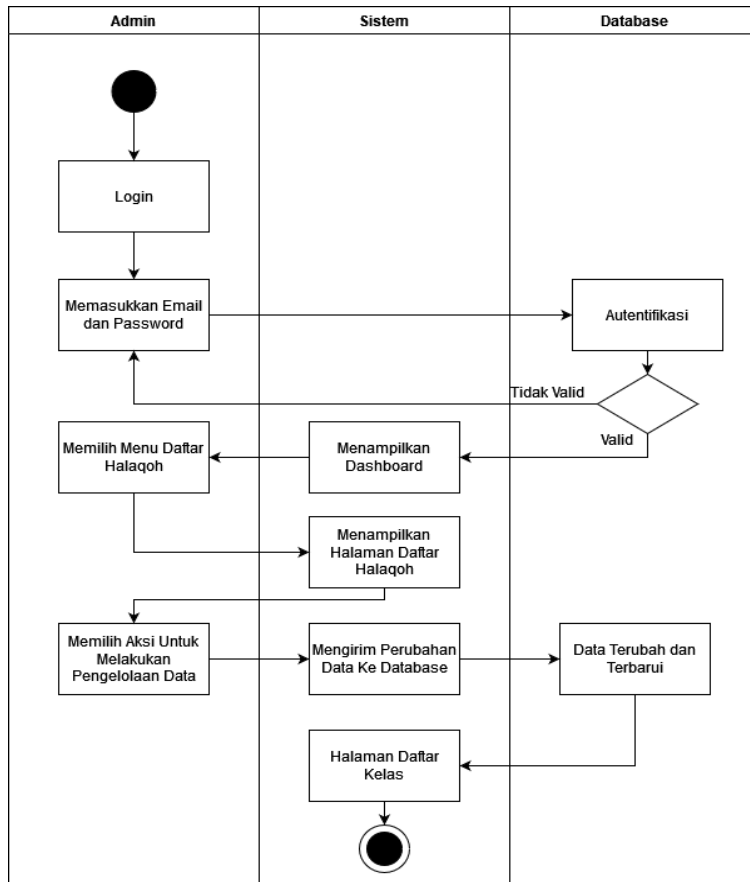
Gambar 2.a menunjukkan sebuah alur dari *activity diagram* dari kegiatan Hafalan yang memiliki langkah proses dari sistem monitoring. Alur proses dimulai dari membuka halaman *website* kemudian dilakukannya aktivitas *login* dengan memasukkan *email* dan *password* yang kemudian jika benar maka akan dialihkan ke halaman awal yaitu *dashboard*. Pada bagian dalam sistem setelah melakukan *login* terdapat sidebar yang berisi menu-menu yang dapat digunakan sesuai kebutuhan dan aktivitas yang diperlukan, seperti jika memilih menu “Daftar Hafalan” maka akan menuju halaman daftar hafalan dan akan muncul daftar hafalan para santri. Alur berikutnya juga merupakan tahapan dari masing-masing *Activity Diagram*, seperti: Gambar 2.b. Kelola Akun Ustad/Santri, Gambar 2.c. Kelola Halaqoh, Gambar 2.d. Kelola Kelas, Gambar 2.e. Tambah Data, Gambar 2.f. Periksa Hafalan.



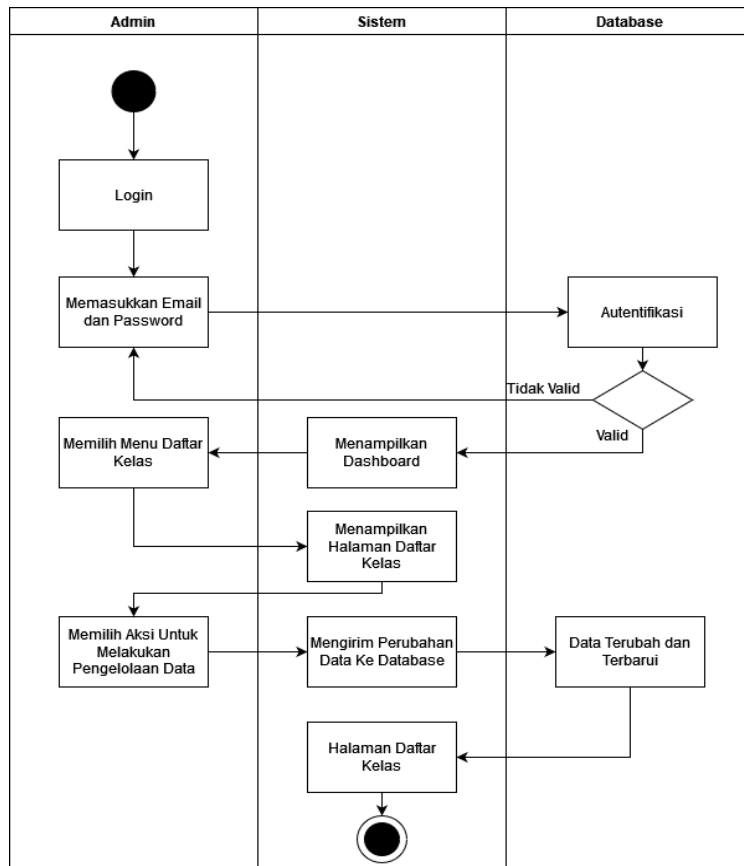
Gambar 2.a. Activity Diagram Hafalan



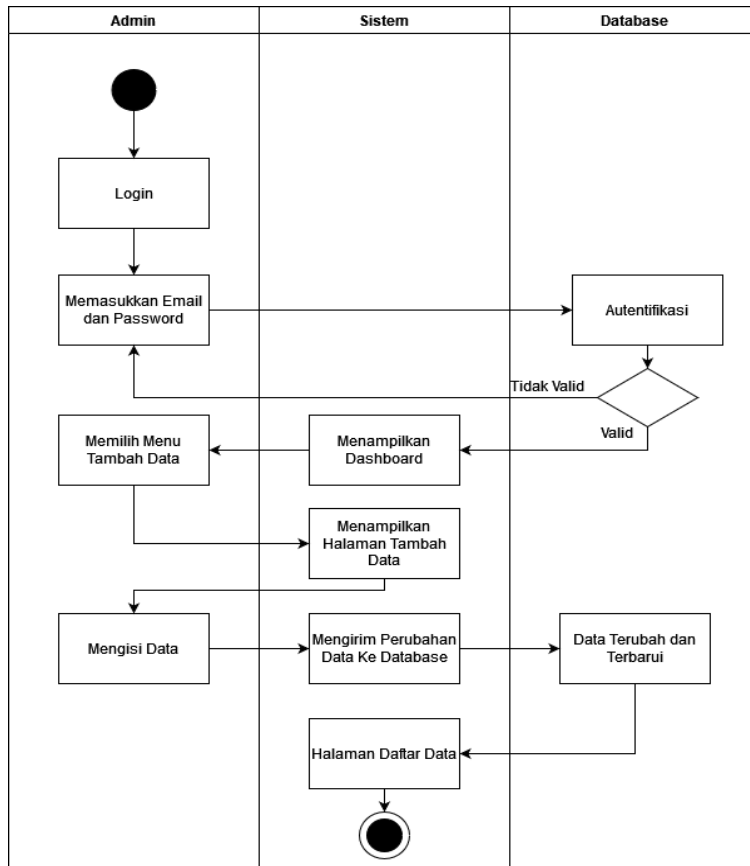
Gambar 2.b. Activity Diagram Akun Ustad dan Akun Santri



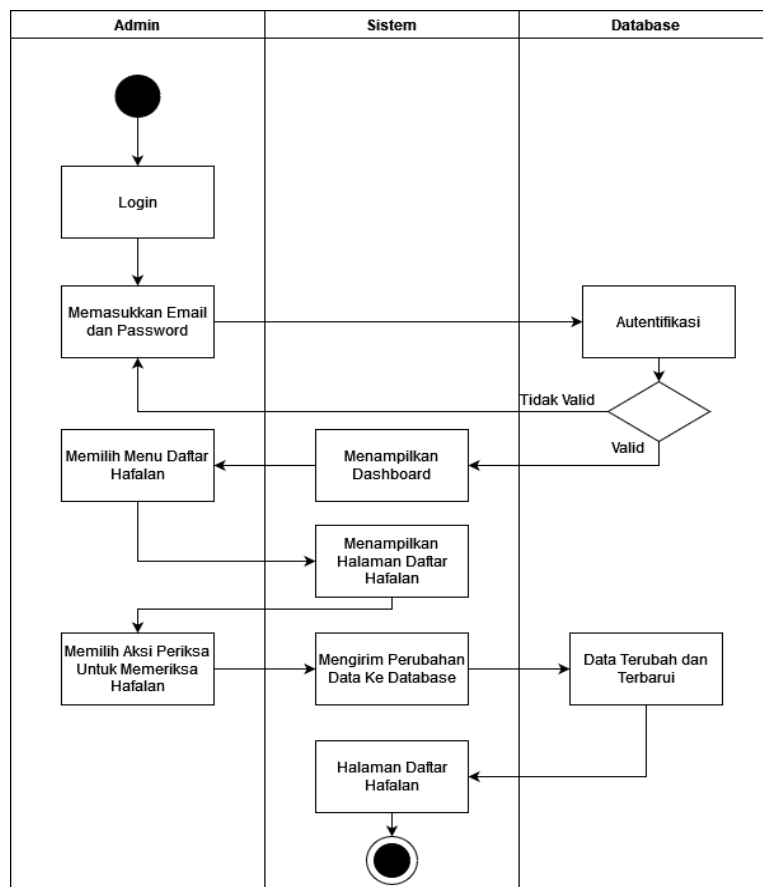
Gambar 2.c. Activity Diagram Kelola Halaqoh



Gambar 2.d. Activity Diagram Kelola Kelas



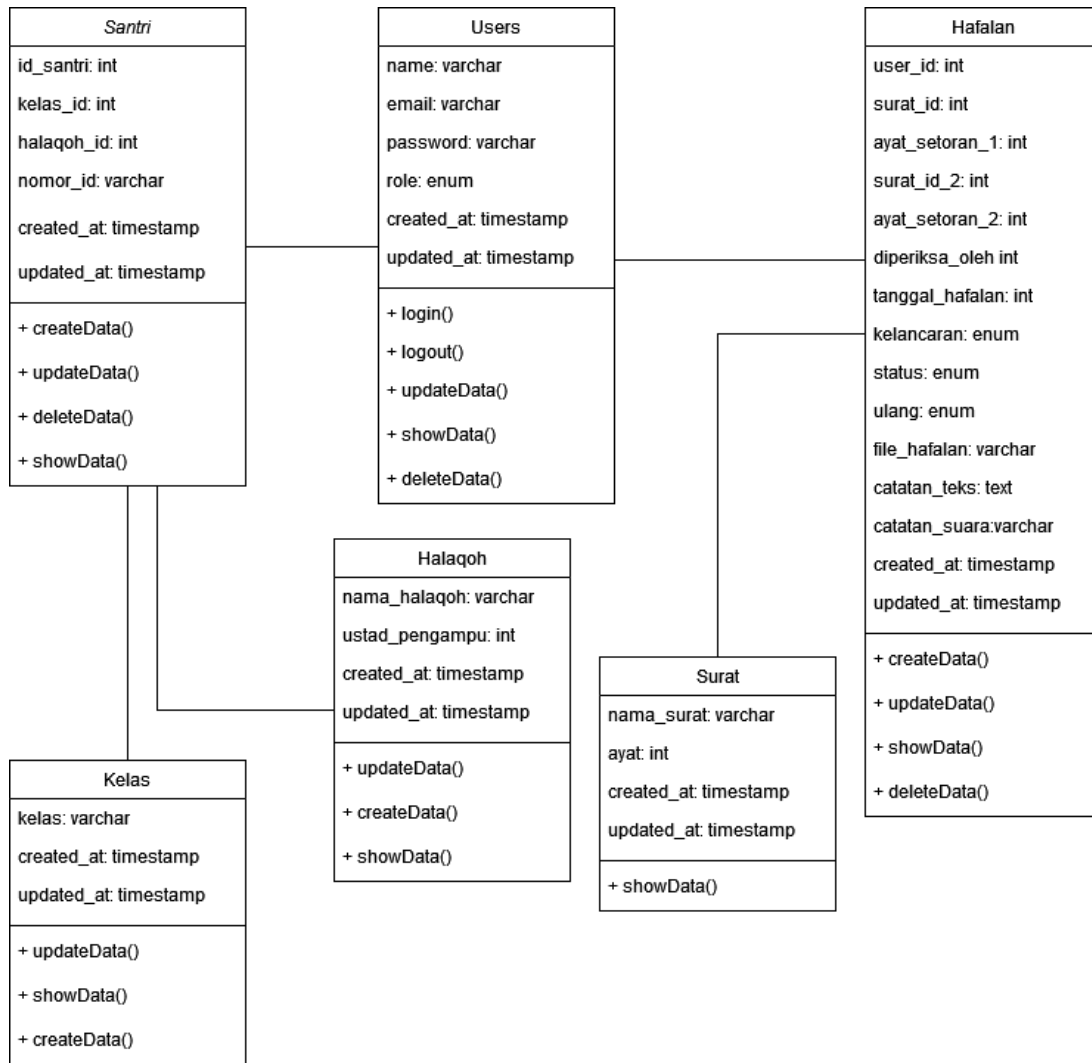
Gambar 2.e. Activity Diagram Tambah Data



Gambar 2.f. Activity Diagram Periksa Hafalan

2.2.3 Class Diagram

Diperlihatkan pada Gambar 3 merupakan sebuah gambar *class diagram* yang berisikan penggambaran kelas-kelas pada sistem seperti antar satu kelas dengan kelas lainnya sehingga menciptakan hubungan yang berkaitan satu sama lain.



Gambar 3. Class Diagram

2.3 Implementasi

Pengimplementasian ialah tahap penting dalam pengembangan sistem. Pada tahap ini, perencanaan diubah menjadi realisasi dalam bentuk sistem yang dapat dijalankan oleh pengguna. Dengan perencanaan yang matang, implementasi yang cermat, dan pengujian yang menyeluruh, sistem yang berkualitas tinggi dapat dihasilkan. Pengembang menggunakan Laravel dalam melakukan implementasi ke dalam bentuk web. Menurut McCool (2012), Laravel merupakan sebuah rangkaian pengembangan web Model-View-Controller (MVC) yang ditulis menggunakan bahasa pemrograman PHP. Tujuan utama dari Laravel adalah untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya awal pengembangan dan biaya pemeliharaan jangka panjang, sekaligus meningkatkan pengalaman pengembang dalam bekerja dengan aplikasi. Ini dilakukan melalui penyediaan sintaks yang jelas dan

ekspresif serta kumpulan fitur inti yang menyeluruh, yang pada akhirnya akan menghemat waktu implementasi secara signifikan.

2.4 Pengujian

Metode *blackbox* dipilih pada penelitian ini karena fokusnya terhadap evaluasi kesesuaian aplikasi yang dikembangkan dengan spesifikasi yang telah ditetapkan, dengan mempertimbangkan baik masukan maupun keluaran (Ahrizal et al., 2020). Penguji tidak perlu mengetahui tentang kode internal aplikasi, sehingga disebut dengan pengujian berbasis spesifikasi dan pengujian perilaku (Verma, Khatana, & Chaudhary, 2017). Kemudian sistem akan dilakukan uji skala usability sistem atau disebut dengan *System Usability Scale* (SUS). *System Usability Scale* (SUS) merupakan sebuah skala yang digunakan untuk mengukur kegunaan atau *usability* suatu sistem. Ini adalah kuesioner singkat berisi sepuluh pertanyaan yang masing-masing menggunakan skala Likert dari sangat setuju hingga sangat tidak setuju. Skala ini dirancang untuk memberikan skor yang berkisar dari 0 hingga 100 (Hyzy et al., 2022).

2.5 Penyebaran

Tahap penyebaran adalah saat perangkat lunak atau sistem telah selesai dikembangkan, diuji, dan didokumentasikan, dan siap untuk digunakan oleh pengguna akhir. Penyebaran melibatkan proses mengunggah atau menginstal perangkat lunak ke lingkungan produksi atau produksi, di mana pengguna akhir dapat mengaksesnya. Ini juga melibatkan konfigurasi perangkat lunak dan infrastruktur yang diperlukan untuk menjalankannya secara efisien dan aman di lingkungan produksi.

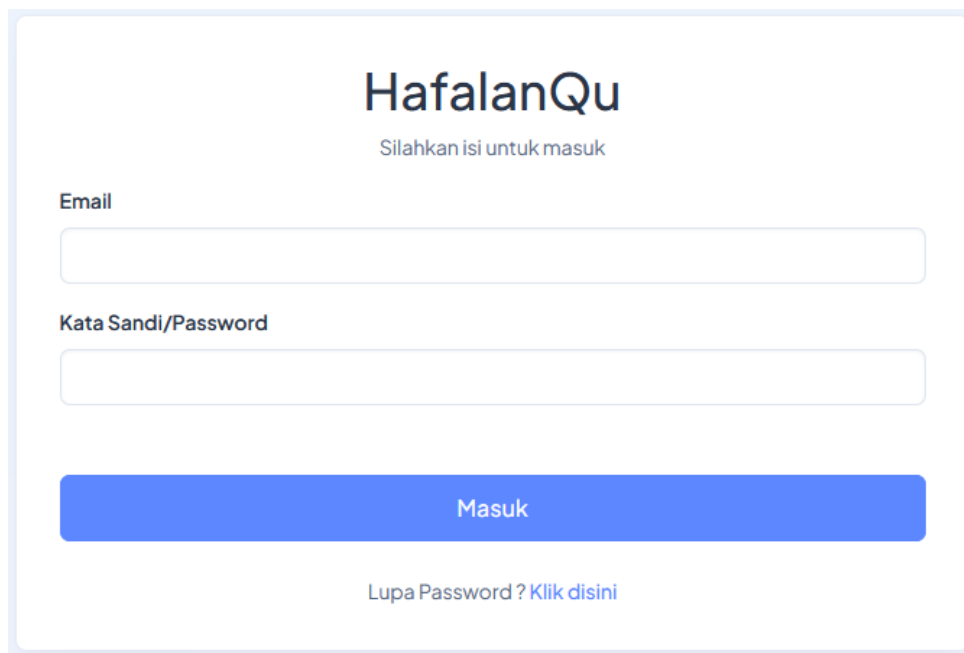
3. HASIL DAN PEMBAHASAN

Pengembangan sistem dalam penelitian ini menghasilkan sebuah sistem monitoring berbasis web yang dapat membantu untuk melakukan setoran hafalan santri secara daring. Sistem ini dibuat sebagai alternatif untuk melakukan setoran hafalan tahfidzul quran tanpa harus melakukan setoran secara manual. Berikut ini adalah hasil dari pengembangan sistem dipenelitian ini.

3.1 Hasil Tampilan Sistem

3.1.1 Formulir Login

Sebelum memasuki ke dalam sistem terdapat halaman *login* untuk menentukan apakah pengguna sudah terdaftar dalam sistem. Pengguna diminta untuk memasukkan informasi *login* seperti *email* dan *password* mereka untuk masuk ke dalam sistem. Jika benar dan berhasil masuk maka setiap akun akan diarahkan ke halaman sesuai dengan peran mereka masing-masing.



HafalanQu
Silahkan isi untuk masuk

Email

Kata Sandi/Password

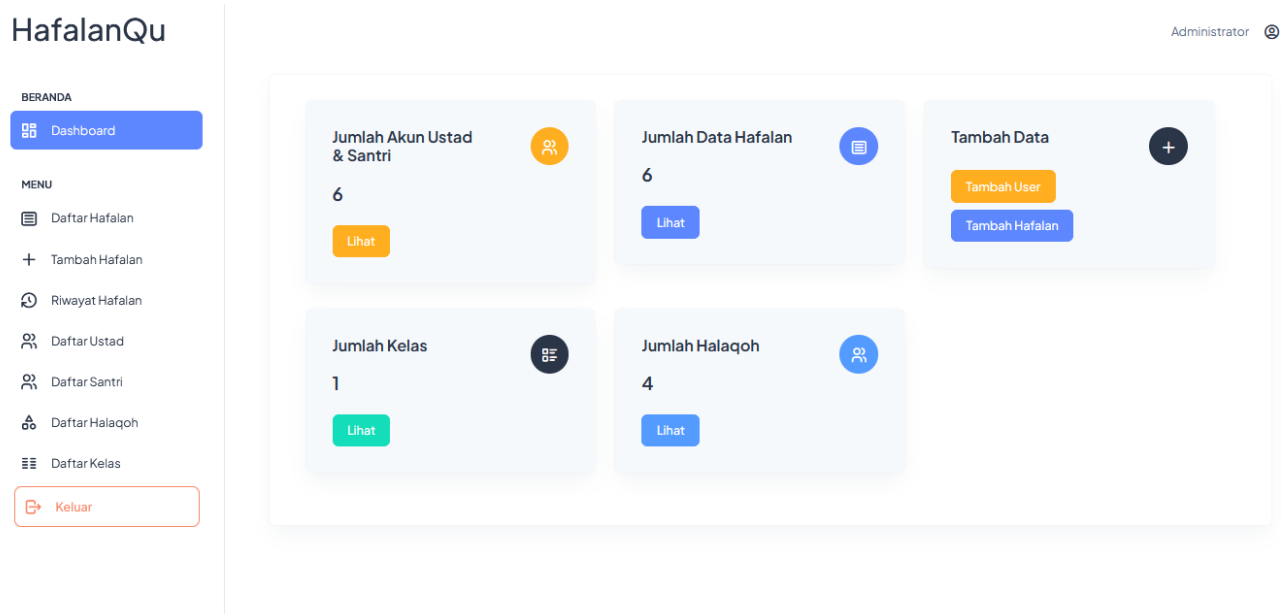
Masuk

[Lupa Password ? Klik disini](#)

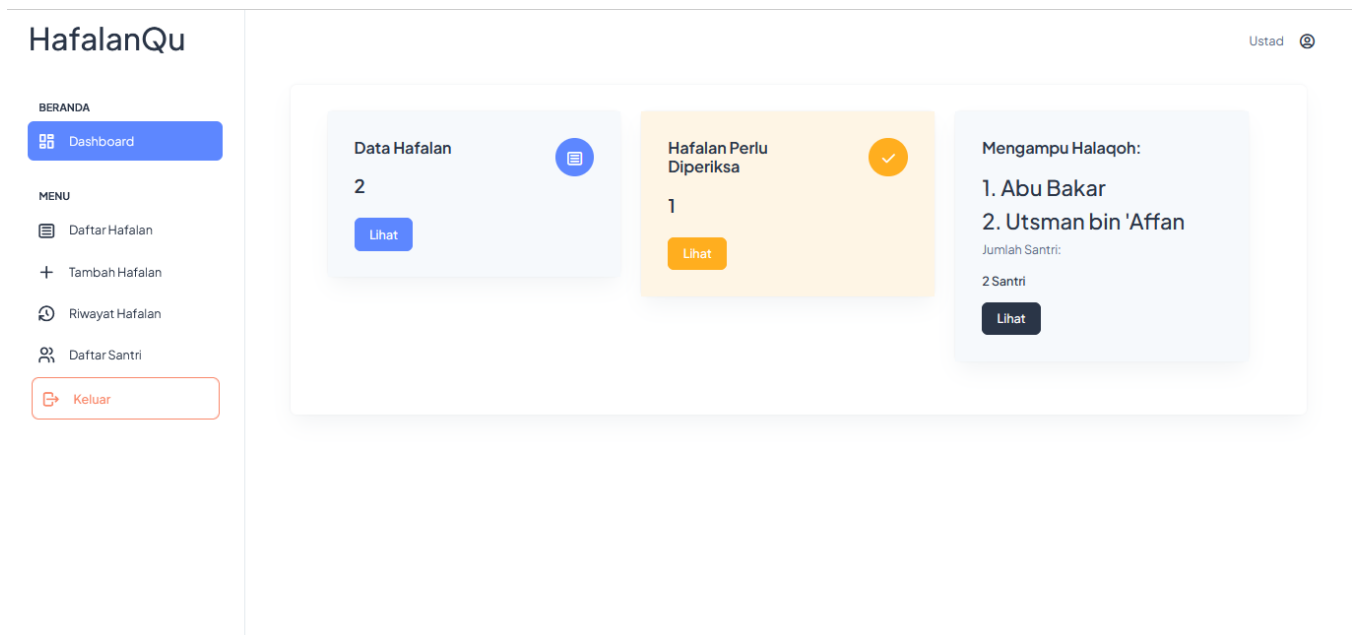
Gambar 4.a. Formulir Login

3.1.2 Halaman Dashboard

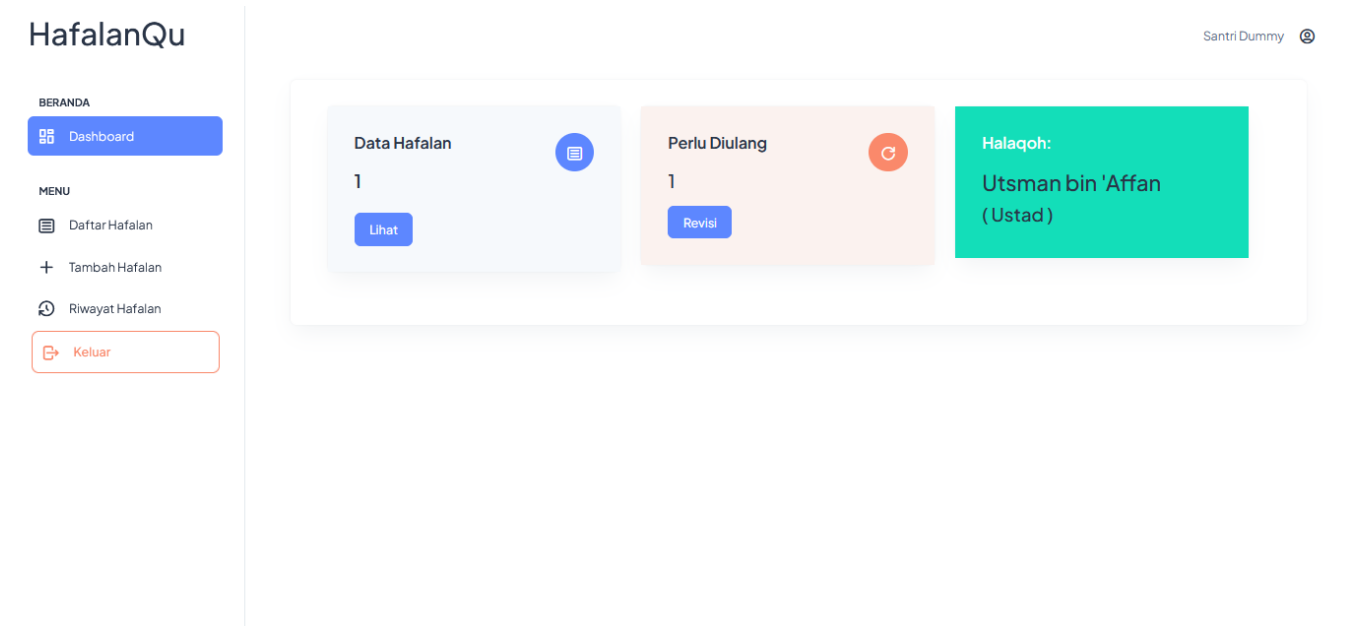
Gambar 4.b menunjukkan tampilan dari *Dashboard* dari admin yang berisi beberapa informasi terkait pengelolaan data pada sistem monitoring. Gambar 4.c tampilan dari *Dashboard* ustad memiliki informasi jumlah hafalan dan hafalan yang perlu diperiksa serta halaqoh yang diampu. Gambar 4.d merupakan *Dashboard* dari santri memiliki informasi jumlah data hafalan serta yang perlu diulang, dan kelompok halaqoh santri.



Gambar 4.b. Dashboard Admin



Gambar 4.c. Dashboard Ustad



Gambar 4.d. Dashboard Santri

3.1.3 Halaman Daftar, Tambah, dan Riwayat Hafalan

Seperti yang tertampil pada Gambar 4.e menunjukkan daftar hafalan santri dengan beberapa informasi seperti nama, dari surat apa ayat berapa sampai surat apa ayat berapa, tanggal melakukan hafalan, serta status dari hafalan tersebut. Gambar 4.f merupakan halaman untuk menambah data hafalan santri dengan mengisi data hafalan dan unggah file audio hafalan. Gambar 4.g menunjukkan riwayat dari hafalan santri, halaman ini hampir serupa dengan daftar hafalan dan yang membedakan hanya pada letak tampilan mengulangnya saja.

BERANDA

Dashboard

MENU

Daftar Hafalan

+ Tambah Hafalan

Riwayat Hafalan

Daftar Ustad

Daftar Santri

Daftar Halaqoh

Daftar Kelas

Keluar

No	Nama	Dari	Sampai	Tanggal	Status	Aksi
1	Azril Delfrian Adi Putra	Al-Ma'un (الماعون) Ayat 1	Al-Ma'un (الماعون) Ayat 7	2024-05-14	belum diperiksa	⋮
2	Azril Delfrian Adi Putra	Al-Ikhlâs (الإخلاص) Ayat 1	Al-Ikhlâs (الإخلاص) Ayat 4	2024-05-14	belum diperiksa	⋮
3	Azril Delfrian Adi Putra	An-Nashr (النصر) Ayat 1	An-Nashr (النصر) Ayat 3	2024-05-13	sudah diperiksa	⋮
4	Azril Delfrian Adi Putra	An-Naba' (النبا) Ayat 1	An-Naba' (النبا) Ayat 40	2024-05-13	belum diperiksa	⋮
5	Santri Dummy	Al-Baqarah (البقرة) Ayat 1	Al-Baqarah (البقرة) Ayat 20	2024-05-13	belum diperiksa	⋮
6	Santri Tes	An-Nashr (النصر) Ayat 1	An-Nashr (النصر) Ayat 3	2024-05-13	sudah diperiksa	⋮

Menampilkan 1 sampai 6 dari 6 hasil

Gambar 4.e. Daftar Hafalan

BERANDA

Dashboard

MENU

Daftar Hafalan

+ Tambah Hafalan

Riwayat Hafalan

Daftar Ustad

Daftar Santri

Daftar Halaqoh

Daftar Kelas

Keluar

Tambah Hafalan

Nama Santri*

Pilih Santri

Dari Surat*

Pilih Surat

Dari Ayat*

Pilih Ayat

SAMPAI

Sampai Surat*

Pilih Surat

Sampai Ayat*

Pilih Ayat

File Hafalan

Telusuri... Tidak ada berkas dipilih.

Silahkan upload file hafalan berupa file audio .mp3

Simpan

Gambar 4.f. Tambah Hafalan

BERANDA

Dashboard

MENU

Daftar Hafalan

+ Tambah Hafalan

Riwayat Hafalan

Daftar Ustad

Daftar Santri

Daftar Halaqoh

Daftar Kelas

Keluar

No	Aksi	Nama	Dari	Sampai	Tanggal	Mengulang
1	Detail	Azril Delfrian Adi Putra	Al-Ma'un (الماعون) Ayat 1	Al-Ma'un (الماعون) Ayat 7	2024-05-14	-
2	Detail	Azril Delfrian Adi Putra	Al-Ikhlâs (الإخلاص) Ayat 1	Al-Ikhlâs (الإخلاص) Ayat 4	2024-05-14	-
3	Detail	Azril Delfrian Adi Putra	An-Nashr (النصر) Ayat 1	An-Nashr (النصر) Ayat 3	2024-05-13	tidak
4	Detail	Azril Delfrian Adi Putra	An-Naba' (النبا) Ayat 1	An-Naba' (النبا) Ayat 40	2024-05-13	-
5	Detail	Santri Dummy	Al-Baqarah (البقرة) Ayat 1	Al-Baqarah (البقرة) Ayat 20	2024-05-13	-
6	Detail	Santri Tes	An-Nashr (النصر) Ayat 1	An-Nashr (النصر) Ayat 3	2024-05-13	tidak

Menampilkan 1 sampai 6 dari 6 hasil

Gambar 4.g. Riwayat Hafalan

3.1.4 Halaman Daftar Ustad dan Daftar Santri

Gambar 4.h menampilkan halaman daftar ustad yang digunakan untuk mengelola data akun ustad seperti melakukan penambahan data, pengeditan data, dan penghapusan data. Sedangkan pada Gambar 4.i menunjukkan bahwa daftar santri yang ditujukan untuk mengelola data santri sama seperti halaman daftar ustad. Halaman daftar ustad dan santri ini hanya dapat diakses oleh admin saja, sedangkan untuk bagian ustad hanya dapat melihat daftar santri yang diampu saja.

HafalanQu

BERANDA

Dashboard

MENU

Daftar Hafalan

Tambah Hafalan

Riwayat Hafalan

Daftar Ustad

Daftar Santri

Daftar Halaqoh

Daftar Kelas

Keluar

Administrator

Cari...

Cari

Tambah Akun

Jumlah: 10

No	Nama	Email	Role	Tanggal Dibuat	Aksi
1	Ustad	ustad@gmail.com	ustad	2023-12-07 17:39:47	
2	Azril Delfrian Adi Putra (Ustad)	azril.ustad@hafalanqu.com	ustad	2024-01-16 07:34:19	
3	Arhan Yudha Pamungkas	yudha@gmail.com	ustad	2024-05-13 18:29:51	

Menampilkan 1 sampai 3 dari 3 hasil

Gambar 4.h. Daftar Ustad

HafalanQu

BERANDA

Dashboard

MENU

Daftar Hafalan

Tambah Hafalan

Riwayat Hafalan

Daftar Ustad

Daftar Santri

Daftar Halaqoh

Daftar Kelas

Keluar

Administrator

Cari...

Cari

Tambah santri

Jumlah: 10

No	Nama	Email	Halaqoh	Kelas	Nomor Induk	Aksi
1	Azril Delfrian Adi Putra	azril@hafalanqu.com	Ali bin Abi Thalib	Kelas Contoh	L200200202	
2	Santri Dummy	santri@gmail.com	Utsman bin 'Affan	Kelas Contoh	L200200300	
3	Santri Tes	santri2@gmail.com	Abu Bakar	Kelas Contoh	L200200301	

Menampilkan 1 sampai 3 dari 3 hasil

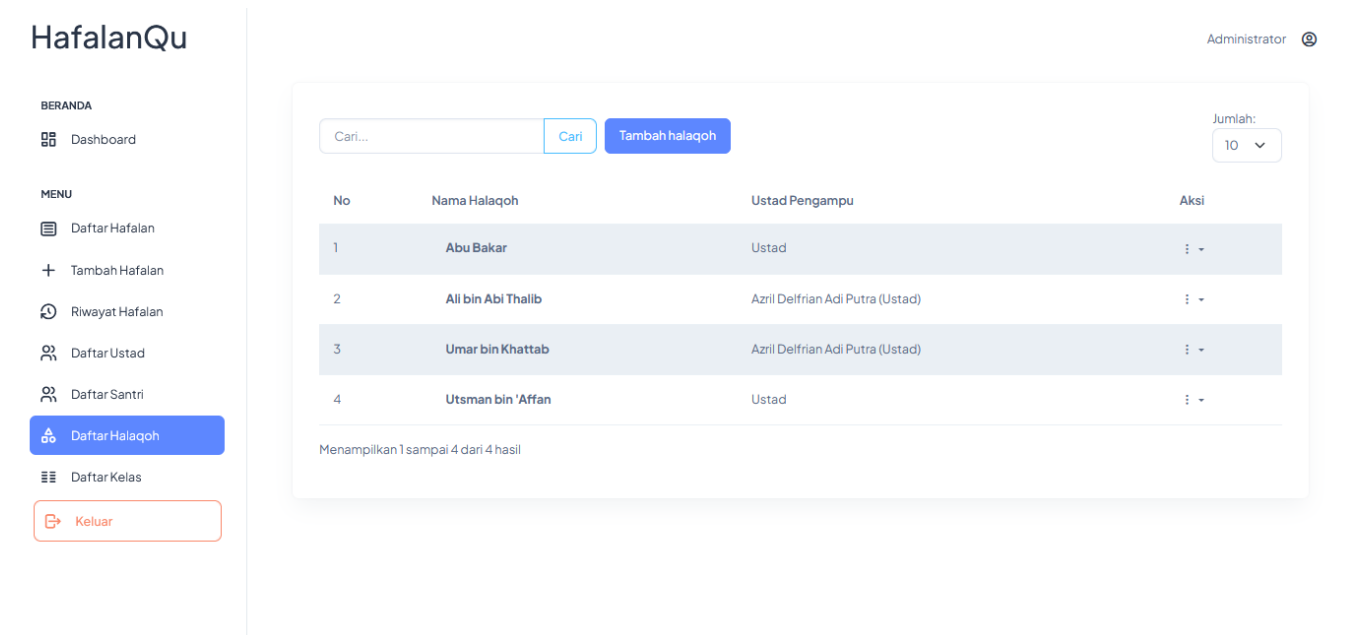
Gambar 4.i. Daftar Santri

3.1.5 Halaman Daftar Halaqoh dan Kelas

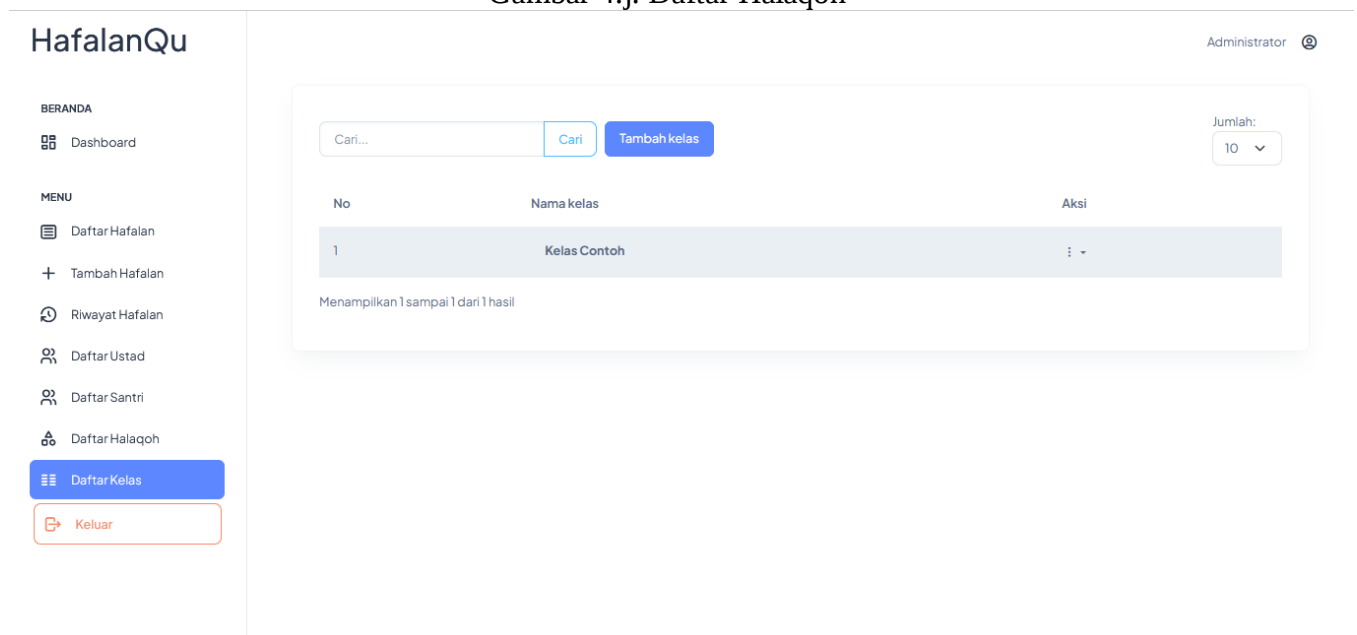
Halaman daftar halaqoh yang terlihat pada Gambar 4.j berfungsi penting dalam pengelolaan halaqoh. Di halaman ini, admin dapat menambah data halaqoh baru serta memilih ustad yang akan menjadi pengampu atau pembimbing halaqoh tersebut. Proses pemilihan ustad ini sangat penting untuk

memastikan halaqoh berjalan dengan baik dan mencapai tujuannya.

Sedangkan, halaman Daftar Kelas pada Gambar 4.k digunakan untuk pengelolaan kelas secara keseluruhan. Di halaman ini, admin bisa menambah kelas baru, memperbarui informasi yang ada, dan mengubah data kelas yang sudah ada. Fitur ini memastikan informasi kelas selalu terkini dan memudahkan koordinasi kegiatan pembelajaran.



Gambar 4.j. Daftar Halaqoh

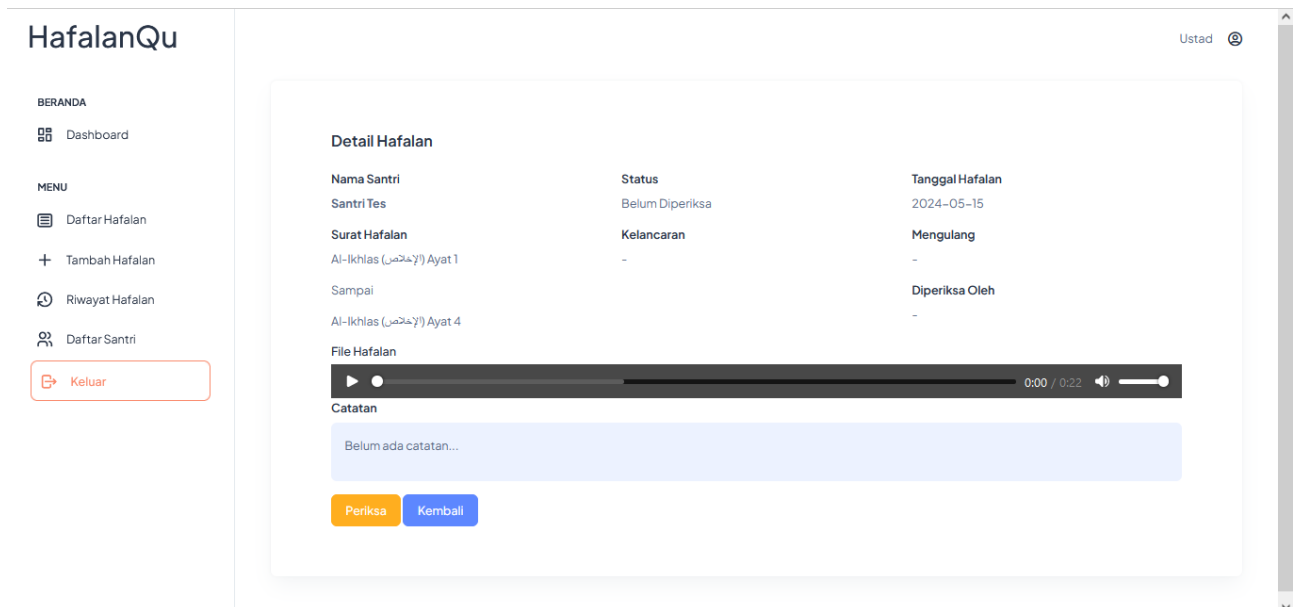


Gambar 4.k. Daftar Kelas

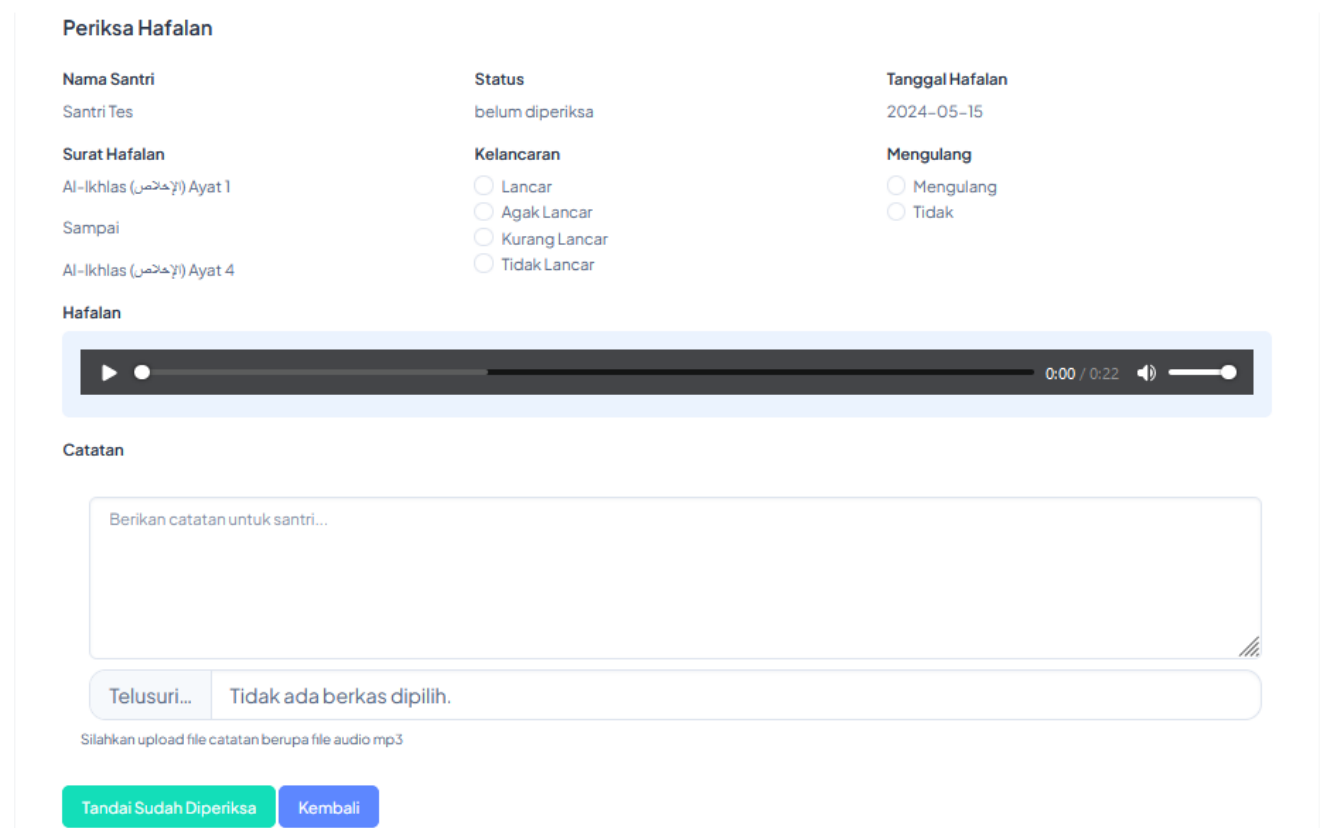
3.1.6 Halaman Detail, Periksa, dan Revisi Hafalan

Halaman Detail Hafalan pada Gambar 4.l menyediakan informasi rinci mengenai hafalan seorang santri. Di halaman ini, admin, ustad, dan santri dapat melihat detail hafalan yang telah dicapai oleh santri, termasuk surat yang dihafal, tanggal hafalan, serta komentar atau catatan yang diberikan oleh pengampu hafalan. Gambar 4.m yaitu Halaman Periksa Hafalan digunakan untuk memeriksa hafalan yang telah

dilakukan oleh santri. Di sini, ustad dapat mendengarkan hafalan santri, memberikan penilaian, dan mencatat koreksi yang diperlukan. Proses pemeriksaan ini penting untuk memastikan bahwa hafalan dilakukan dengan benar dan sesuai dengan standar yang ditetapkan. Halaman Revisi Hafalan pada Gambar 4.n difungsikan untuk melakukan koreksi terhadap hafalan yang sudah diperiksa sebelumnya. Pada halaman ini, ustad dapat memberikan revisi kepada santri, mencatat bagian yang perlu diperbaiki, dan memberikan arahan untuk peningkatan hafalan. Fitur ini membantu santri dalam memperbaiki kesalahan dan meningkatkan kualitas hafalannya.



Gambar 4.l. Detail Hafalan



Gambar 4.m. Periksa Hafalan

BERANDA

Dashboard

MENU

Daftar Hafalan

+ Tambah Hafalan

Riwayat Hafalan

Keluar

Edit/Revisi Hafalan

Surat

Al-Baqarah (البقرة) - 286 ayat

Ayat

1

SAMPAI

Surat

Al-Baqarah (البقرة) - 286 ayat

Ayat

20

File Hafalan Sebelumnya



Upload File Hafalan

Telusuri...

Tidak ada berkas dipilih.

Silahkan upload file hafalan berupa file audio

Simpan

Gambar 4.n. Revisi Hafalan

3.2 Pengujian *Black Box*

Pengembangan sistem ini menggunakan pengujian dengan jenis *black box* yang melibatkan tiga langkah utama. Langkah-langkah tersebut dijelaskan sebagai berikut: a) Mendefinisikan *test case*, b) Jalankan kasus uji, c) Manajemen hasil pengujian dan pembuatan laporan pengujian (Supriyono, 2020).

Tabel 1. Hasil Pengujian Black Box Formulir *Login*

No	Bagian	Komponen yang diuji	Skenario	Hasil yang diharapkan	Hasil
1.	Login	Formulir Login	Memasukkan email dan password lalu klik “Masuk”	Pengguna masuk ke sistem dan menampilkan halaman dashboard	Valid
2.	Logout	Logout/Keluar	Memilih tombol “Keluar”	Akun keluar dari sesi dan kembali ke Login	Valid

Tabel 2. Hasil Pengujian Black Box Halaman Admin

No	Bagian	Komponen yang diuji	Skenario	Hasil yang diharapkan	Hasil
1.	Dashbo ard	Dashboard	Mengklik tombol “Dashboard”	Pengguna beralih ke halaman dashboard	Valid
2.	Daftar Hafalan	Tabel Daftar Hafalan	Mengklik tombol “Daftar Hafalan”	Menampilkan seluruh data hafalan santri	Valid
3.	Daftar Hafalan	Edit	Mengubah data hafalan lalu “Simpan”	Data berubah sesuai yang diisi	Valid

Tabel 2. Lanjutan Hasil Pengujian Black Box Halaman Admin

No	Bagian	Komponen yang diuji	Skenario	Hasil yang diharapkan	Hasil
4.	Daftar Hafalan	Detail	Klik tombol “Detail”	Menampilkan halaman detail data yang dipilih	Valid
5.	Daftar Hafalan	Periksa Hafalan	Mengisi data dan klik “Tandai sudah diperiksa”	Data tersimpan dan tampil	Valid
6.	Daftar Hafalan	Hapus	Menghapus salah satu data	Data yang dipilih terhapus	Valid
7.	Tambah Data Hafalan	Tambah Data Hafalan	Mengisi formulir data hafalan dan klik “Simpan”	Data bertambah sesuai dengan yang diisi	Valid
8.	Riwayat Hafalan	Riwayat Hafalan	Memilih menu “Riwayat Hafalan”	Menampilkan tabel riwayat data hafalan santri	Valid
9.	Daftar Santri	Tambah Akun	Mengisi data formulir tambah akun Santri	Data tersimpan dan dapat digunakan	Valid
10.	Daftar Santri	Edit	Mengubah data akun Santri	Data berubah dan dapat digunakan	Valid
11.	Daftar Santri	Delete	Menghapus data akun Santri	Data yang terpilih terhapus	Valid
12.	Daftar Halaqoh	Daftar Halaqoh	Memilih menu “Daftar Halaqoh”	Menampilkan tabel yang berisi daftar halaqoh	Valid
13.	Daftar Halaqoh	Tambah Akun	Mengisi data formulir tambah halaqoh	Data tersimpan dan dapat digunakan	Valid
14.	Daftar Halaqoh	Edit	Mengubah data halaqoh	Data berubah dan dapat digunakan	Valid
15.	Daftar Kelas	Daftar Kelas	Memilih menu “Daftar Kelas”	Menampilkan tabel yang berisi daftar kelas	Valid
16.	Daftar Kelas	Tambah Kelas	Mengisi data formulir tambah kelas	Data tersimpan dan dapat digunakan	Valid
17.	Daftar Kelas	Edit	Mengubah data kelas	Data berubah dan dapat digunakan	Valid

Tabel 3. Hasil Pengujian Black Box Halaman Ustad

No	Bagian	Komponen yang diuji	Skenario	Hasil yang diharapkan	Hasil
1.	Dashboard	Dashboard	Memilih menu “Dashboard”	Beralih ke halaman Dashboard	Valid
2.	Daftar Hafalan	Tabel Daftar Hafalan	Mengklik tombol “Daftar Hafalan”	Menampilkan data hafalan santri sesuai halaqoh yang diampu ustad	Valid
3.	Daftar Hafalan	Detail	Klik tombol “Detail”	Menampilkan halaman detail data yang dipilih	Valid
4.	Daftar Hafalan	Periksa Hafalan	Mengisi data dan klik “Tandai sudah diperiksa”	Data tersimpan dan tampil	Valid
5.	Tambah Data Hafalan	Tambah Data Hafalan	Mengisi formulir data hafalan dan klik “Simpan”	Data bertambah sesuai dengan yang diisi	Valid
6.	Riwayat Hafalan	Riwayat Hafalan	Memilih menu “Riwayat Hafalan”	Menampilkan tabel riwayat data hafalan santri sesuai yang diampu	Valid

Tabel 4. Hasil Pengujian Black Box Halaman Santri

No	Bagian	Komponen yang diuji	Skenario	Hasil yang diharapkan	Hasil
1.	Dashboard	Dashboard	Memilih menu “Dashboard”	Beralih ke halaman Dashboard	Valid
2.	Daftar Hafalan	Tabel Daftar Hafalan	Mengklik tombol “Daftar Hafalan”	Menampilkan data hafalan sesuai santri yang sedang login	Valid
3.	Daftar Hafalan	Detail	Klik tombol “Detail”	Menampilkan halaman detail data yang dipilih	Valid
4.	Daftar Hafalan	Revisi	Klik tombol “Detail” lalu klik “Revisi”	Data terubah dan tersimpan	Valid

Tabel 4. Lanjutan Hasil Pengujian Black Box Halaman Santri

No	Bagian	Komponen yang diuji	Skenario	Hasil yang diharapkan	Hasil
5.	Tambah Data Hafalan	Tambah Data Hafalan	Mengisi formulir data hafalan dan klik “Simpan”	Data bertambah sesuai dengan yang diisi	Valid
6.	Riwayat Hafalan	Riwayat Hafalan	Memilih menu “Riwayat Hafalan”	Menampilkan tabel riwayat data hafalan sesuai santri yang sedang login	Valid

3.3 System Usability Scale (SUS)

Pengujian SUS memiliki 10 (sepuluh) pernyataan dimana setiap pertanyaan memiliki skala jawaban rentang angka 1 sampai dengan 5. Skor 1 berarti sangat tidak setuju, skor 2 berarti tidak setuju, skor 3 berarti agak setuju, skor 4 berarti setuju, dan skor 5 berarti sangat setuju. Penilaian aplikasi melibatkan tiga aspek utama: *acceptability* (penerimaan), *grade scale* (skala kelas), dan *adjective rating* (peringkat).

- *Acceptability*: Menentukan apakah aplikasi diterima oleh pengguna. Tingkatnya meliputi *not acceptable*, *marginal (low dan high)*, dan *acceptable*.
- *Grade scale*: Menentukan kualitas aplikasi. Tingkatnya meliputi A, B, C, D, dan E.
- *Adjective rating*: Menentukan peringkat aplikasi (berguna atau tidak). Tingkatnya meliputi *worst imaginable*, *poor*, *ok*, *good*, *excellent*, dan *best imaginable*.

Hasil evaluasi aplikasi ditentukan oleh nilai komulatif rata-rata instrumen, bukan nilai masing-masing pernyataan. Hasilnya berupa *acceptability*, *grade scale*, dan *adjective rating*. Kesimpulan diperoleh dengan *grade scale* melalui ketentuan rata-rata: $A > 80,3$, $74 < B < 80,3$, $68 < C < 74,3$, $51 < D < 68$, dan $F < 51$. (Fatmawati, 2021).

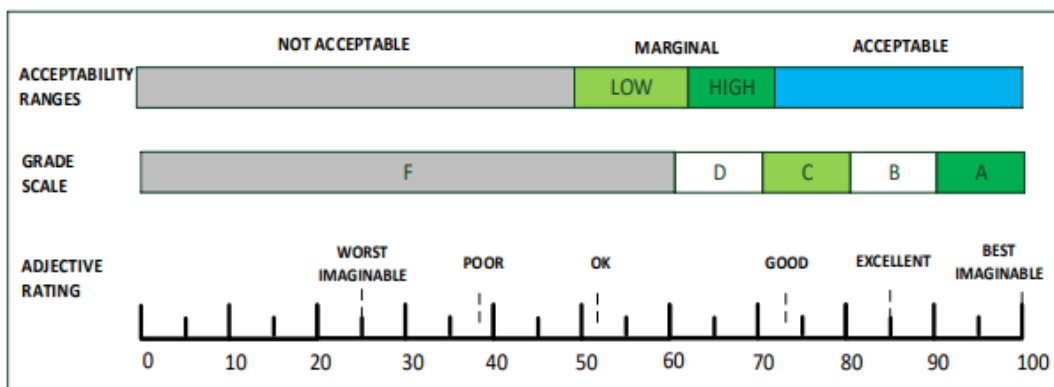
Tabel 4. Hasil Perhitungan SUS

Skor Hasil Hitung										Jumlah	Nilai (Jumlah x 2.5)
Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
0	4	4	4	4	4	4	4	0	4	32	80
4	4	4	4	4	4	4	4	4	4	40	100
3	3	4	3	3	3	3	4	1	1	28	70
4	0	4	0	4	0	4	0	4	0	20	50
1	4	0	4	0	4	0	4	0	4	21	53

Tabel 4. Lanjutan Hasil Perhitungan SUS

Skor Hasil Hitung										Jumlah	Nilai (Jumlah x 2.5)
Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
4	4	3	1	3	3	3	3	1	2	27	68
4	4	4	4	0	4	3	4	4	3	34	85
1	2	2	2	2	2	2	2	2	2	19	48
2	2	2	2	2	2	2	2	2	2	20	50
2	4	3	2	0	2	3	2	0	2	20	50
2	2	2	2	2	2	2	2	2	2	20	50
4	3	4	4	4	4	4	4	4	3	38	95
4	0	4	0	4	0	4	0	4	0	20	50
2	2	2	0	2	2	2	2	2	0	16	40
3	3	3	3	3	3	3	3	3	3	30	75
4	3	4	3	4	4	4	4	4	1	35	88
2	4	4	4	4	4	4	4	0	0	30	75
3	2	3	2	4	1	3	3	2	3	26	65
4	4	4	4	0	3	2	1	4	3	29	73
4	3	4	4	4	3	3	3	4	3	35	88
4	3	3	2	4	3	2	3	3	4	31	78
4	4	3	4	4	3	4	4	4	4	38	95
3	3	4	3	3	4	4	2	4	3	33	83
2	4	3	4	3	4	3	4	2	4	33	83
4	4	4	4	3	4	4	4	4	4	39	98
4	3	4	4	4	4	4	4	4	4	39	98
2	4	4	4	3	4	4	3	3	3	34	85
Skor Rata-rata (Hasil Akhir)											78.87

Tabel 4 merupakan hasil dari pengujian SUS dengan perolehan nilai 78.8 dengan tingkat *Acceptability Range* : "**Acceptable**", *Grade Scale* : "**B**", dan *Adjective Rating* : "**Good**"



Gambar 6. Indikator Penilaian SUS

4. KESIMPULAN

Berdasarkan hasil dari penelitian ini, penulis dapat mengembangkan Sistem Monitoring Tahfidzul Qur'an yang membantu dengan memberikan alternatif dan kemudahan dalam kegiatan Hafalan Al-

Qur'an. Memanfaatkan SDLC (*Software Development Life Cycle*) dengan metode Agile. Pengujian *Black Box* diatas menunjukkan bahwa unit sistem berjalan dengan semestinya. Hasil dari pengujian *System Usability Scale* (SUS) memperoleh nilai 78.8 dengan perolehan *acceptability, grade, dan rating "Acceptable/B/Good"* menunjukkan bahwa sistem ini dapat diterima. Meskipun sistem ini sudah dapat diterima, adapun harapan pengembangan selanjutnya. Melakukan penambahan fitur seperti progress perjuz santri, statistik hafalan, notifikasi pengingat, penambahan fungsi moitoring dibagian ustad, dan lain-lain. Pengembangan sistem ini dikembangkan supaya bermanfaat untuk instansi yang memerlukan sistem monitoring hafalan Tahfidz Al-qur'an seperti Pondok Pesantren modern ataupun Boarding School.

DAFTAR PUSTAKA

- Assisi, nisah, Septiarini, A., Harsa Kridalaksana, A., & Wati, M. (2022). Rancang Bangun Aplikasi Hafalan Al-Quran dengan Google Speech API Berbasis Android. *JURTI*, 6(1).
- Gazali, M. I. A. (2010). Keutamaan Membaca dan Menghafal Al-Qur'an. *Islamhouse.Com*, 1–9.
- Syah, F., Bustami, B., & Nisa, C. C. (2018). Sistem Pendeteksi Pola Tajwid Wajibul Ghunnah Pada Surat Al-Waqiah Menggunakan Metode Sokal & Sneath. *TECHSI - Jurnal Teknik Informatika*, 10(2), 01. <https://doi.org/10.29103/techsi.v10i2.904>
- Ahrizal, D., Miftah, M. K., Kurniawan, R., Zaelani, T., & Yulianti, Y. (2020). Pengujian Perangkat Lunak Sistem Informasi Peminjaman PlayStation dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing. *Jurnal Informatika Universitas Pamulang*, 5(1), 73. <https://doi.org/10.32493/informatika.v5i1.4338>
- Fatmawati, A. (2021). Evaluasi Usability pada Learning Management System OpenLearning Menggunakan System Usability Scale. *INOVTEK Polbeng - Seri Informatika*, 6(1), 120. <https://doi.org/10.35314/isi.v6i1.1881>
- Hyzy, M., Bond, R., Mulvenna, M., Bai, L., Dix, A., Leigh, S., & Hunt, S. (2022). System Usability Scale Benchmarking for Digital Health Apps: Meta-analysis. *JMIR MHealth and UHealth*, 10(8), 1–11. <https://doi.org/10.2196/37290>
- McCool, S. (2012). *Laravel starter*. Packt Publishing.
- Naufal Faruq, M., & Maryam, M. (2024). Implementasi Metode Agile Pada Pengembangan Aplikasi Manajemen Pengelolaan Layanan Wifi. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(6), 3472–3478. <https://doi.org/10.36040/jati.v7i6.7868>
- Pargaonkar, S. (2023). A Comprehensive Research Analysis of Software Development Life Cycle (SDLC) Agile & Waterfall Model Advantages, Disadvantages, and Application Suitability in Software Quality Engineering. *International Journal of Scientific and Research Publications*, 13(8), 120–124. <https://doi.org/10.29322/ijsrp.13.08.2023.p14015>
- Pramitasari, B., & Nurgiyatna, N. (2019). Sistem Informasi Unit Kegiatan Mahasiswa Marching Band Universitas Muhammadiyah Surakarta berbasis Web. *Emitor: Jurnal Teknik Elektro*, 19(2), 59–65. <https://doi.org/10.23917/emitor.v19i2.7998>
- Rambe, B. H., Pane, R., Irmayani, D., Nasution, M., & Munthe, I. R. (2020). UML Modeling and Black Box Testing Methods in the School Payment Information System. *Jurnal Mantik*, 4(3), 1634–1640. <https://iocscience.org/ejournal/index.php/mantik>

- Supriyono. (2020). Software Testing with the approach of Blackbox Testing on the Academic Information System. *IJISTECH: International Journal of Information System & Technology*, 3(2), 227–233.
- Suryana, T., Oktaviany, I., & Dewi, N. P. (2021). Website-based Al-Quran memorization monitoring and evaluation system design. *Journal of Physics: Conference Series*, 1764(1). <https://doi.org/10.1088/1742-6596/1764/1/012188>
- Verma, A., Khatana, A., & Chaudhary, S. (2017). A Comparative Study of Black Box Testing and White Box Testing. *International Journal of Computer Sciences and Engineering*, 5(12), 301–304. <https://doi.org/10.26438/ijcse/v5i12.301304>