

# Rancang Bangun Aplikasi Tahfidz dengan Metode Rapid Application Development

**Fauza Ibrahim Farid\*<sup>1</sup>, Alon Jala Tirta Segara<sup>2</sup>**

Direktorat Kampus Purwokerto Universitas Telkom

e-mail: <sup>1</sup>ibrahimfauza@student.telkomuniversity.ac.id, <sup>2</sup>alon@telkomuniversity.ac.id

\*Penulis Korespondensi

Diterima: 15 Juni 2025; Direvisi: 26 November 2025; Disetujui: 09 Desember 2025

## **Abstrak**

*Sistem pencatatan hafalan Al-Qur'an konvensional di SMPIT Salman Alfarisi Boarding School menimbulkan masalah seperti data tercecer, ketidakteraturan pencatatan, dan kesulitan pelaporan, dimana 64% pengampu menyatakan sistem manual tidak efektif. Penelitian ini bertujuan mengembangkan aplikasi pencatatan tahfidz berbasis Android menggunakan metode Rapid Application Development (RAD) untuk mengatasi permasalahan tersebut. Proses pengembangan meliputi identifikasi kebutuhan pengguna melalui observasi, wawancara, dan studi pustaka, dilanjutkan perancangan sistem menggunakan UML dan desain antarmuka, implementasi dengan Flutter untuk aplikasi mobile dan Laravel untuk REST API, serta evaluasi melalui kuesioner dan pengujian black-box. Hasil implementasi menunjukkan aplikasi ini mampu meningkatkan efektifitas pencatatan dan mempermudah akses data dengan hasil implementasi dan evaluasi pengguna yang terhadap kepraktisan serta kegunaan sistem. Seluruh sembilan kasus uji fungsionalitas menunjukkan hasil "Berhasil", mengindikasikan aplikasi berjalan sesuai rancangan untuk pengampu, wali santri, dan admin. Aplikasi yang terdiri dari platform Android dan web ini berhasil mengatasi kendala sistem manual dan berpotensi diadopsi secara lebih luas oleh lembaga pendidikan sejenis*

**Kata kunci:** sistem pencatatan, rad, aplikasi android, pendidikan islam

## **Abstract**

*Each article in a scientific journal must contain an abstract that is generally only one paragraph (not a summary consisting of several paragraphs) in English (required) and/or Indonesian that is concise, clear, intact, there is no reference to libraries, images, and tables, and fully describes the essence of the entire content of the writing. Abstract in Indonesian italicized with Times New Roman 11 pt. Abstract length 150-200 words. The abstract should be clear, descriptive and should give a brief overview of the problem under study given the limited number of words allowed. Abstracts of journal articles include at least objectives, brief methods, and important findings. The abstract should end with a comment on the importance of the result or a brief conclusion. A conclusion is something that has already happened, not something that is still expected. Keywords are no more than 5 words.*

**Keywords:** tahfidz, recording system, rad, android application, islamic education

## **1. PENDAHULUAN**

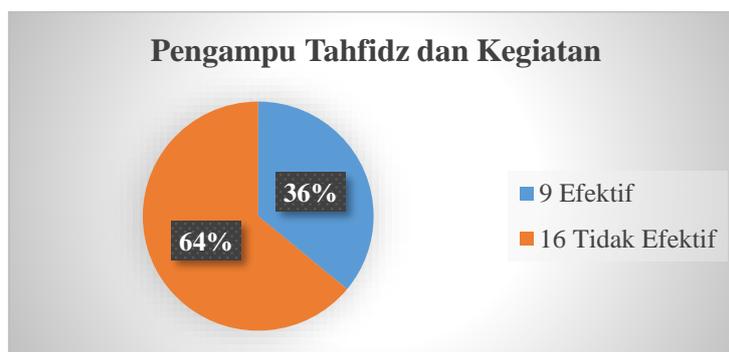
Pembelajaran memegang peran krusial dalam mengembangkan seluruh potensi peserta didik untuk menjawab tantangan zaman dan mendukung kemajuan bangsa. Di tengah

---

kompleksitas tantangan pendidikan saat ini, diperkenalkan kebijakan Merdeka Belajar Kemdikbud (2019) sebagai sebuah transformasi fundamental. Kebijakan ini bertujuan untuk mendorong kemandirian, inovasi, serta memberikan keleluasaan bagi pendidik dan peserta didik dalam menciptakan proses pembelajaran yang berkualitas dan relevan [1]. Belajar adalah tahapan perubahan perilaku individu yang komprehensif dan relatif permanen melalui pengalaman dan interaksi lingkungan, melibatkan proses kognitif. Hasil belajar mencerminkan perubahan individu, berupa peningkatan pengetahuan, pemahaman mendalam, perubahan sikap positif, serta pengembangan keterampilan dan kecakapan [2]. Penilaian hasil belajar yang komprehensif merupakan elemen esensial untuk mengukur pencapaian tujuan pembelajaran secara utuh. Idealnya, penilaian ini mencakup ketiga domain pembelajaran kognitif, afektif, dan psikomotor karena penguasaan aspek kognitif saja tidak selalu mencerminkan kemampuan siswa untuk menerapkan pengetahuan atau mengatasi masalah secara praktis. Oleh karena itu, evaluasi yang menyeluruh diperlukan untuk memahami perkembangan siswa dalam aspek pengetahuan, sikap, dan keterampilannya [3].

Maka pencatatan belajar siswa berperan penting dalam proses penilaian hasil. Pencatatan sistematis dan komprehensif memfasilitasi pemantauan perkembangan siswa, evaluasi efektivitas metode pembelajaran, identifikasi kendala belajar siswa, dan penyesuaian strategi pembelajaran. Pencatatan berfungsi sebagai alat komunikasi antara pendidik, siswa, dan orang tua, memungkinkan pemahaman progres belajar dan kolaborasi untuk mencapai hasil yang optimal [4]. Keberhasilan belajar siswa adalah fokus utama pendidikan di sekolah, dan banyak hal bisa memengaruhinya. Oleh karena itu, diperlukan tolak ukur yang jelas untuk mengukur dan mengevaluasi keberhasilan belajar siswa, sehingga dapat diketahui bagaimana pencapaian yang diharapkan dan bagaimana gambaran keberhasilan yang sesungguhnya [5]. Salah satu sistem pencatatan yang digunakan di Indonesia adalah seperti yang digunakan SMP Islam Terpadu Mutiara Kota Pariaman. Sistem informasi pencatatan hafalan Al-Qur'an berbasis website dirancang untuk memfasilitasi pengelolaan data hafalan siswa. Sistem ini berfungsi sebagai alat bantu interaktif dalam proses pencatatan hafalan dan penyusunan laporan akhir semester. Sistem ini dapat memudahkan guru dalam melakukan pencatatan hafalan siswa, serta meningkatkan akurasi dan efisiensi dalam pengolahan data hafalan [6].

SMPIT Salman Alfarisi Boarding School Yogyakarta, yang berlokasi di Kabupaten Sleman, fokus pada pengembangan program Tahfidz Qur'an dan pendidikan karakter. Sistem pencatatan hafalan yang digunakan saat ini masih bersifat manual dengan menggunakan buku catatan. Hal ini menimbulkan tantangan dalam proses pelaporan dan pemantauan, seperti hilangnya data, kesulitan rekapitulasi, dan keterbatasan akses data secara *real time*.



Gambar 1. Efektifitas pencatatan manual menurut pengampu tahfidz dan kegiatan smpit salman alfarisi boarding school.

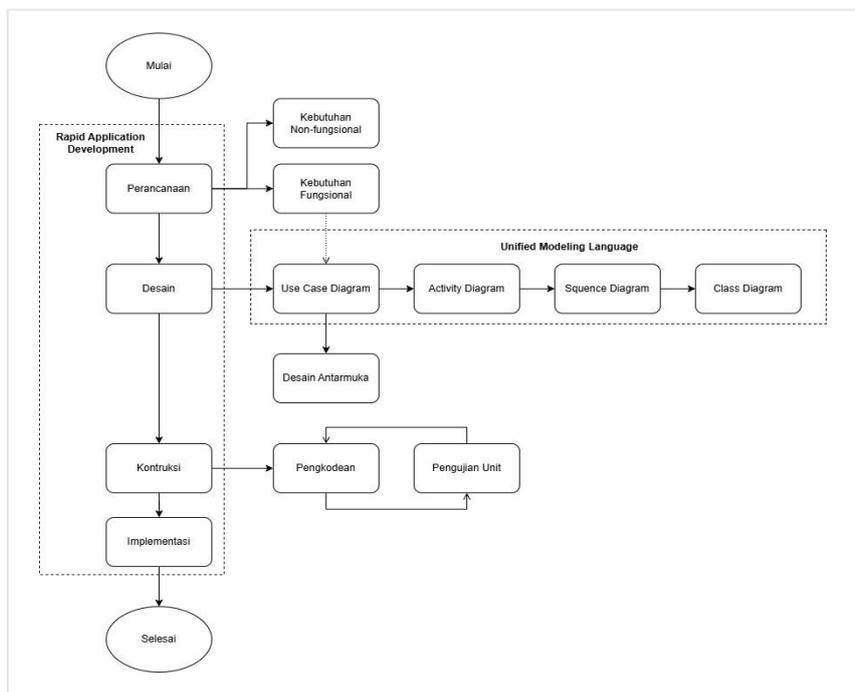
Hasil dari survei yang didapatkan seperti yang ditampilkan dalam Gambar 1, hasil survei terhadap 25 pengampu, sebanyak 64% menyatakan tidak efektif. Maka berdasarkan permasalahan tersebut akan dikembangkan sebuah aplikasi pencatatan tahfidz yang dapat digunakan untuk mencatat hasil belajar santri, mengelola secara otomatis dan efisien, sehingga memberikan

kemudahan mencatat dan mencari data termasuk kenyamanan dan keamanan data yang tidak perlu khawatir tercecer atau rusak dibandingkan menggunakan catatan kertas.

Penelitian ini mengadopsi metode Rapid Application Development (RAD), sebuah strategi pengembangan sistem yang menekankan pada kecepatan penyelesaian melalui keterlibatan aktif pengguna secara berkelanjutan, RAD mengharuskan pengembang untuk merancang sistem informasi secara menyeluruh [7]. Tahapan pada metode RAD mempermudah pengembangan sistem karena tahapannya yang sistematis dan cepat, pengujian dengan menunjukkan sistem informasi yang dibangun memenuhi syarat pada analisis kebutuhan [8]. Metode ini sangat mengutamakan partisipasi pengguna dalam setiap fase analisis dan desain, sehingga memastikan kebutuhan mereka dapat terpenuhi secara tepat dan transparan. Dengan demikian, penerapan metode RAD tidak hanya mempercepat proses pengembangan tetapi juga bertujuan meningkatkan kepuasan pengguna terhadap sistem informasi yang dihasilkan karena sesuai untuk pengembangan dalam waktu singkat dengan kolaborasi erat bersama pengguna [9].

## 2. METODE PENELITIAN

Subyek penelitian ini adalah aktifitas tahfidz SMPIT Salman Al Farisi Boarding School Yogyakarta. Objek penelitian adalah sistem pencatatan tahfidz mobile yang akan di kembangkan. Langkah penelitian adalah bagian terpenting dalam upaya pencapaian tujuan dengan membuat aplikasi tahfidz. Metodologi atau cara penelitian yang digunakan diilustrasikan di dalam gambar 2.



Gambar 2. Alir penelitian

### 2.1. Rapid Application Development (RAD)

RAD mengacu pada metode pengembangan yang dirancang untuk memberikan pengembangan yang jauh lebih cepat dan hasil yang lebih berkualitas dibandingkan dengan metode pengembangan waterfall. Tujuan dari proses RAD adalah untuk membuat pengguna merasa "memiliki" produk sebelum produk tersebut selesai. Hal ini memastikan kemitraan yang tepat dan membantu menghasilkan kualitas fungsional [10]. Untuk memastikan bahwa pengembang membangun apa yang benar-benar dibutuhkan oleh pengguna RAD terdiri dari

empat fase seperti yang digambarkan pada gambar 3.



Gambar 3. Fase *rapid application development* by james martin. Sumber [10]

## 2.2. Requirement Planing Phase

*Requirements planning phase* mengharuskan pengguna menentukan fungsi-fungsi sistem yang dibutuhkan. Proses ini melibatkan diskusi yang terstruktur mengenai masalah didalam proses yang perlu diselesaikan. Perencanaan kebutuhan paling efektif dilakukan melalui observasi dan wawancara [10]. Hasil dari observasi dan wawancara adalah dokumen *requirements planning* yang disusun berdasarkan spesifikasi kebutuhan perangkat lunak menjabarkan kebutuhan fungsional dari program yang akan dikembangkan [11].

Pada tahap perencanaan penelitian ini, dilakukan identifikasi kebutuhan pengguna melalui tiga teknik pengumpulan data. Teknik tersebut meliputi observasi langsung di objek penelitian, wawancara dengan bagian Tahfidz serta beberapa santri, dan studi pustaka yang difokuskan pada permasalahan sistem pencatatan tahfidz di SMPIT Salman AlFarisi [12][13].

## 2.3. User Design Phase

Fase ini menekankan keterlibatan aktif pengguna akhir dalam merancang aspek non-teknis dari sistem. Maksudnya bukan fokus pada kode atau infrastruktur, tetapi pada Antarmuka Pengguna (*User interface* - UI): Bagaimana pengguna akan berinteraksi dengan sistem? Tampilan seperti apa yang paling intuitif dan efisien?. Alur Kerja (*Workflow*): Bagaimana tugas-tugas akan dilakukan dalam sistem? Apakah alur ini sesuai dengan proses bisnis yang ada?. Pengalaman Pengguna (*User Experience* - UX): Apakah sistem mudah digunakan dan memuaskan bagi pengguna? [10].

Pada tahap ini dilakukan pembuatan *Unified Modelling Language* (UML) untuk membuat berbagai desain sistem yang akan dibuat seperti *use case*, *activity*, *sequence*, *class diagram* dan *design user interface* [14][15].

1. *Use case diagram* digunakan untuk memperlihatkan bagaimana pengguna (aktor) berinteraksi dengan sistem.
2. *Activity diagram* di gunakan untuk memodelkan proses atau alur kerja di dalam sistem.
3. *Sequence diagram* di gunakan untuk memperlihatkan urutan interaksi antar objek di dalam sistem.
4. *Class diagram* di gunakan untuk menunjukkan bagaiman struktur dari sistem, dengan mengidentifikasi kelas, atribut dan hubungan antar kelas.
5. *Design user interface* digunakan untuk menggambarkan rancangan visual dari aplikasi, termasuk tata letak elemen, komponen interaktif, dan alur navigasi yang akan dialami oleh pengguna saat berinteraksi dengan system.

## 2.4. Construction phase

*Construction phase* adalah momen krusial dalam RAD, rancangan aplikasi dari fase desain pengguna diubah menjadi perangkat lunak fungsional melalui pengembangan dan pengujian berulang. Selama proses ini, pengguna berperan krusial dalam memvalidasi tampilan dan alur transaksi yang dibangun, dengan perubahan kebutuhan signifikan akan segera diberitahukan [10]. Pengujian sistem yang berkelanjutan memastikan pengembangan cepat dengan umpan balik langsung dari pengguna, sehingga produk akhir sesuai harapan sejak awal implementasi [16]. Tahap ini melibatkan pengembangan perangkat lunak berdasarkan desain yang telah disetujui.

### A. Pengkodean

Pada tahap ini dilakukan proses penulisan kode program sesuai yang telah di setujui dari fase design. Sistem ini dikembangkan menggunakan Flutter untuk platform aplikasi mobile, sebuah perangkat UI yang memungkinkan penggunaan lintas platform untuk web, mobile, dan desktop, sehingga dapat berjalan di sistem operasi iOS maupun Android [17]. Untuk mengakses data dari server, aplikasi ini akan memanfaatkan REST API yang berfungsi mengambil data dari database melalui internet. Pembangunan REST API ini sendiri akan menggunakan Laravel, sebuah kerangka kerja *Hypertext Preprocessor* (PHP) yang menerapkan pola *Model, View, Controller* (MVC) dan dipilih karena kesesuaian PHP dengan konsep OOP [18], [19].

## B. Pengujian

Setiap modul yang dikembangkan diuji secara terpisah untuk memastikan fungsionalitasnya. Penelitian ini menerapkan metode *black-box testing* setelah seluruh fitur aplikasi selesai dikembangkan, yang berfokus pada evaluasi pemenuhan syarat fungsional melalui berbagai kondisi input tanpa perlu meninjau struktur kode internal. Pengujian ini secara spesifik memverifikasi detail fungsionalitas seperti antarmuka pengguna, alur penggunaan (*use case*), fungsi-fungsi aplikasi, dan validasi tipe data. Hal ini bertujuan untuk memastikan bahwa kinerja sistem telah sesuai dengan kebutuhan dan perspektif pengguna [20].

### 2.5. Cutover Phase

Cutover phase atau fase implementasi adalah tahap terakhir dalam pengembangan sistem yang fokus pada transisi dari sistem lama ke sistem baru [10]. Pada proses pengembangan sistem pencatatan tahfidz setelah semua modul diuji dan berfungsi dengan baik. Implementasi dapat dilakukan secara bertahap, dengan merilis fitur-fitur yang sudah siap terlebih dahulu.

## 3. HASIL DAN PEMBAHASAN

### 3.1. Requirement Planing Phase

Pada tahap perencanaan dilakukan pengumpulan data melalui wawancara, observasi dan studi pustaka dengan orientasi permasalahan pada sistem pencatatan tahfidz santri. Pengumpulan data yang dilakukan menghasilkan kebutuhan-kebutuhan sistem dengan pengguna yaitu dari pengampu, wali santri, dan admin [13]. Hasil analisis kebutuhan pengguna dapat dilihat pada tabel 1, tabel 2, dan tabel 3.

Tabel 1. Kebutuhan fungsional pengampu

| No | Daftar Kebutuhan                                     |
|----|--|
| 1  | Pengampu tahfidz mencatat tahfidz santri.            |
| 2  | Pengampu dapat merekap catatan tahfidz santri.       |
| 3  | Pengampu dapat mempresensi aktifitas tahfidz santri. |

Tabel 2. Kebutuhan fungsional wali santri

| No | Daftar Kebutuhan   |
|----|--|
| 1  | Wali santri dapat melihat hasil rekap sesuai waktu yang diinginkan |
| 2  | Wali santri dapat melihat aktifitas tahfidz santri.                |

Tabel 3. Kebutuhan fungsional admin

| No | Daftar Kebutuhan                             |
|----|--|
| 1  | Admin dapat mengatur dan mengelola hak user. |
| 2  | Admin dapat mengelola data user.             |
| 3  | Admin dapat membuat laporan rekapitulasi.    |

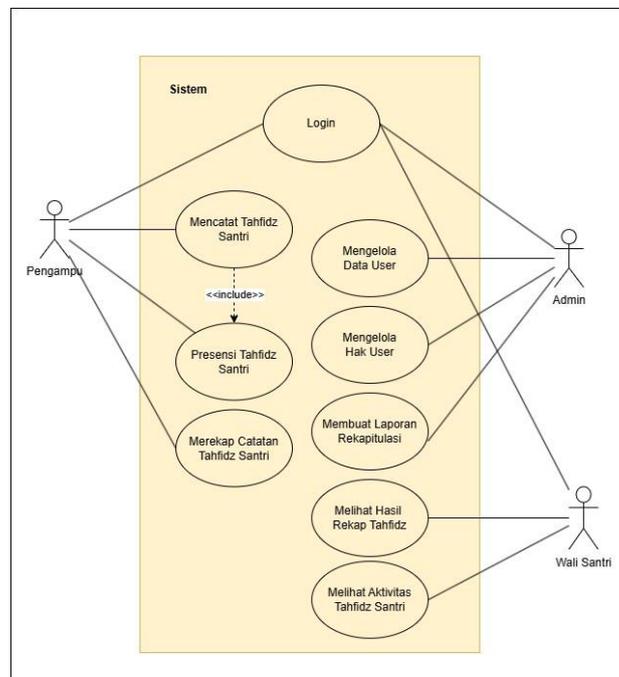
Secara keseluruhan dari tabel 1, tabel 2, dan tabel 3 menggambarkan kebutuhan fungsional dari berbagai pihak yang terlibat dalam suatu sistem. Sistem tersebut perlu menyediakan fitur-fitur yang memungkinkan pengampu, wali santri, dan admin untuk menjalankan tugas dan memperoleh informasi yang mereka butuhkan. Sistem ini akan dirancang untuk memudahkan pengelolaan data tahfidz santri dari fitur yang dibuat dan meningkatkan transparansi informasi kepada wali santri, dengan tetap menjaga keamanan dan kemudahan penggunaan.

### 3.2. User Design Phase

*Design* dilakukan pembuatan *Unified Modelling Language (UML)* seperti *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* yang disesuaikan dengan fase perencanaan yaitu dengan aktor pengampu, wali santri, dan admin.

#### A. Use Case Diagram

*Use case diagram* digunakan untuk memvisualisasikan fungsionalitas utama sistem [21]. Gambar 4 adalah bentuk *visualisasi* dari sistem informasi tahfidz dan interaksi pengguna.



Gambar 4. Use case diagram sistem

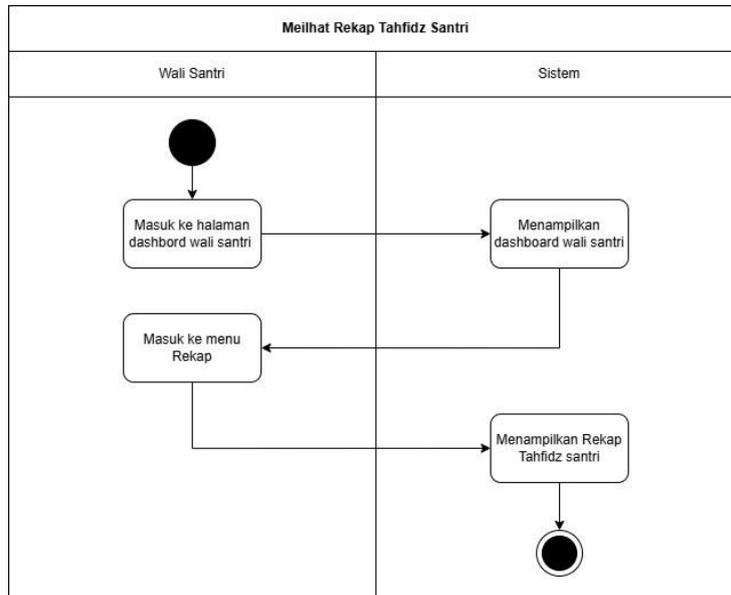
Pengampu bertugas login, mencatat tahfidz santri, dan merekap catatan tahfidz santri. Admin bertugas login, mengelola data user, mengelola hak user, membuat laporan rekapitulasi, dan dapat melihat hasil rekap tahfidz serta wali santri memantau perkembangan tahfidz anak mereka, semua diawali dengan proses logi

#### B. Activity Diagram

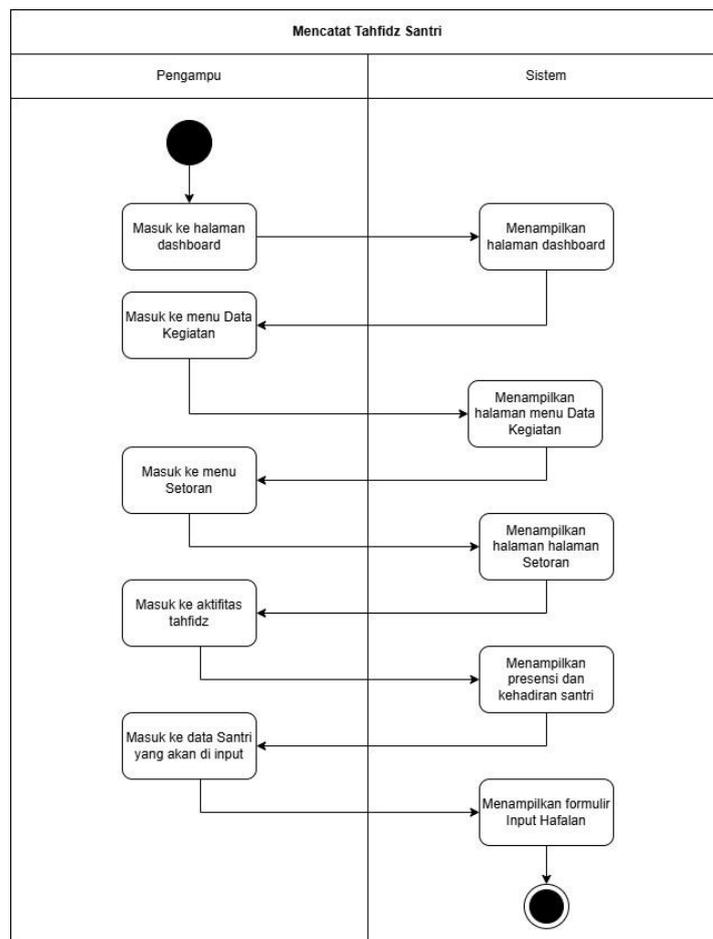
*Activity diagram* ini menggambarkan alur proses atau langkah-langkah yang dilakukan oleh actor saat melakukan aktivitas dalam sistem [21].

Gambar 5 Wali Santri dirancang untuk akses informasi yang mudah, mereka cukup menavigasi dari dashboard ke menu Rekap untuk melihat hasil tahfidz santri. Gambar 6 Pengampu mengikuti proses terstruktur dari dashboard, menavigasi ke menu Data Kegiatan dan Setoran untuk mencatat hafalan santri, yang juga mencakup input presensi. Admin Gambar 7 juga memulai dari dashboard untuk mengakses menu Data dan Rekap Kegiatan guna membuat laporan

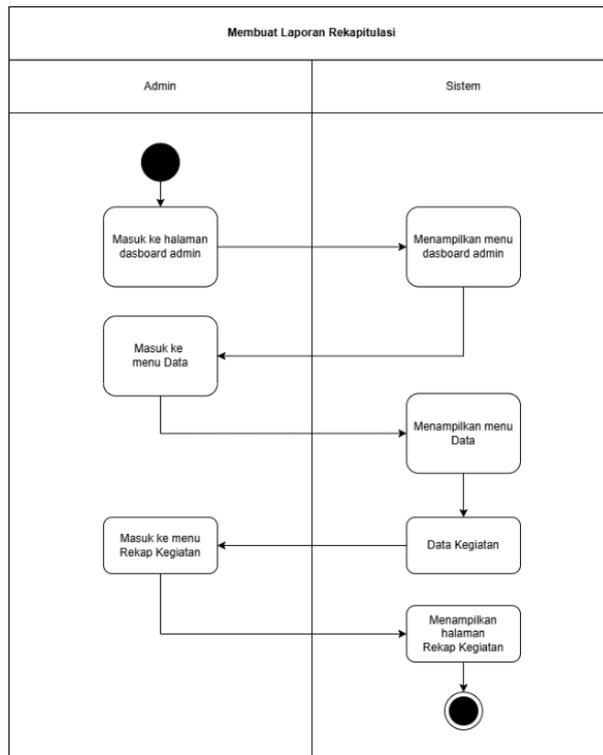
rekapitulasi.



Gambar 5. Activity diagram walisantri melihat rekap tahfidz santri

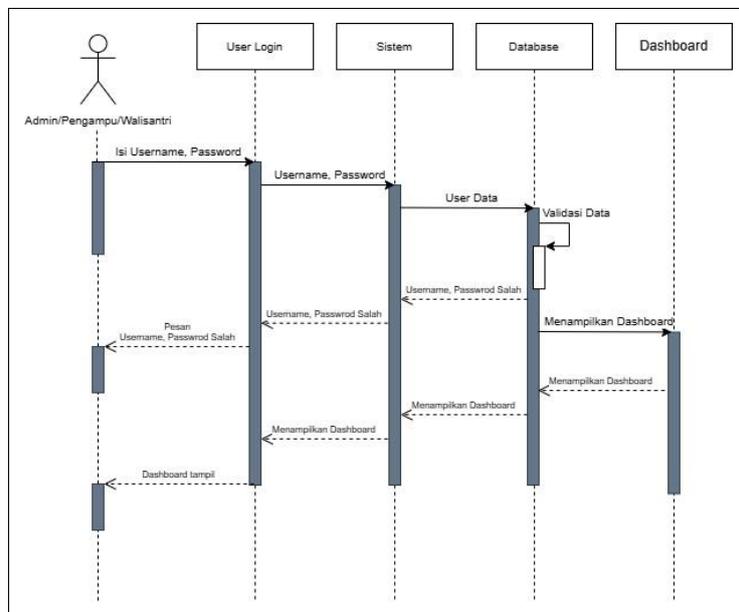


Gambar 6. Activity diagram pengampu mencatat tahfidz santri



Gambar 7. Activity diagram admin membuat laporan rekapitulasi

C. Sequence Diagram



Gambar 8. Sequence diagram login user

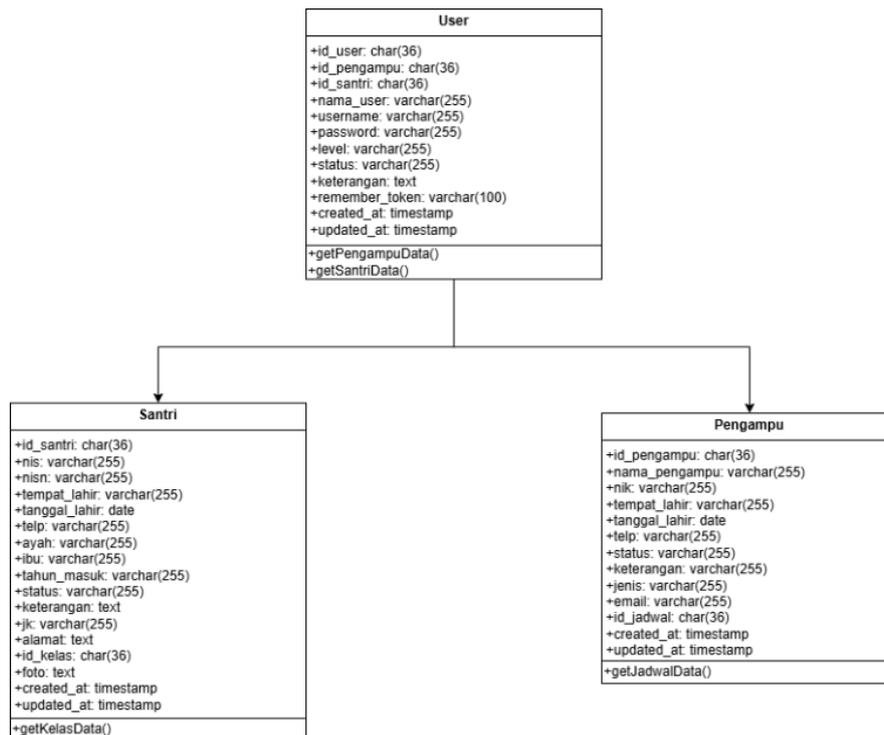
Diagram Sekuens berfungsi untuk merinci alur interaksi antar berbagai entitas dalam suatu sistem. Diagram ini menunjukkan pesan atau perintah yang dikirim antar entitas tersebut, sekaligus menggambarkan urutan waktu prosesnya [22]. Pembahasan pada hasil *sequence diagram* menunjukan login pada setiap *user*, sebelum user bisa melakukan aktifitas didalam sistem diperlukan *login*.

Pada gambar 8 mengilustrasikan alur proses login yang melibatkan aktor

Admin/Pengampu/Walisantri, objek User Login, Sistem, Database, dan Dashboard. Diagram ini menjelaskan interaksi dan urutan pesan yang terjadi selama proses autentikasi pengguna. Alur ini memastikan bahwa setiap langkah dalam proses login, mulai dari input pengguna hingga validasi dan penanganan hasilnya.

#### D. Class Diagram

Gambar 9 menggambarkan class diagram dengan struktur statis dan arsitektur data inti dari sistem informasi tahfidz. Hubungan antar entitas ini didasarkan pada konsep asosiasi, di mana class user bertindak sebagai penghubung utama [21].

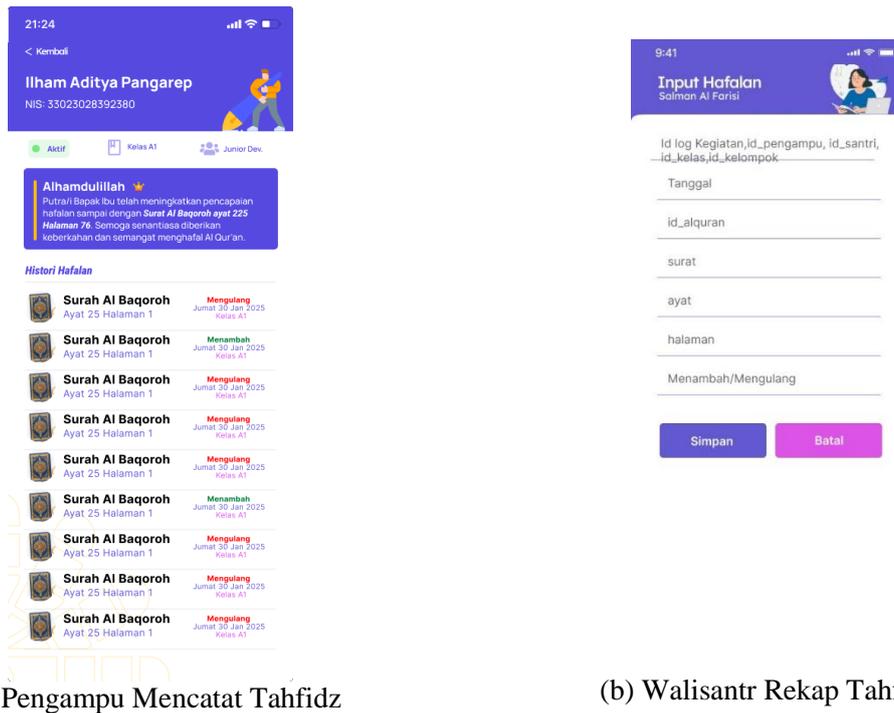


Gambar 9. Class Diagram Sistem

Kelas User memiliki relasi dengan Pengampu melalui atribut id\_pengampu dan dengan Santri melalui id\_santri. Relasi ini biasanya bersifat *one-to-zero-or-one*, yang berarti sebuah akun User dapat terkait dengan nol atau satu entitas Pengampu jika level adalah 'pengampu' atau nol atau satu entitas Santri jika level adalah 'wali santri' atau 'santri'. Adanya atribut kunci asing ini serta metode *getPengampuData()* dan *getSantriData()* di kelas *User* memastikan bahwa data detail seorang pengampu atau santri dapat diakses setelah otentikasi pengguna berhasil.

#### E. Design User interface

Pada tahap ini, desain antarmuka pengguna akan dibuat sesuai dengan apa yang dibutuhkan oleh pengguna [23] dan ditampilkan dalam gambar 10. Perancangan tampilan UI ini meliputi pembuatan elemen visual dan interaktif yang mudah dipahami serta gampang digunakan, sehingga interaksi antara pengguna dan sistem bisa berjalan lebih efisien. Oleh karena itu, masukan dari pengguna baik berupa kritik, saran, ataupun pujian, sangatlah penting untuk memahami kebutuhan dan preferensi mereka [24]. Antarmuka pengguna untuk Wali Santri, khususnya pada halaman rekap hafalan, dirancang untuk memberikan informasi yang jelas dan mudah diakses mengenai perkembangan tahfidz santri.



(a) Pengampu Mencatat Tahfidz (b) Walisantr Rekap Tahfidz  
 Gambar 10. Desain Antarmuka

Desain input hafalan pengampu gambar 10(a) menyediakan formulir efisien untuk mencatat detail hafalan tanggal, surat, ayat, halaman, tipe setoran dengan tombol simpan/batal. sedangkan desain rekap hafalan wali santri gambar 10(b) menyajikan ringkasan dan histori hafalan santri nama, NIS, capaian terakhir, detail setoran.

### 3.3 Construction phase

#### A. Pengkodean

```

<?php
namespace App\Models;

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

class DetailKelompok extends Model
{
    use HasFactory;

    protected $guarded = [];
    protected $table = 'detail_kelompok';

    protected $primaryKey = 'id_detail_kelompok';
    public $incrementing = false;
    protected $keyType = 'string';

    public function santri()
    {
        return $this->belongsTo(Santri::class, 'id_santri', 'id_santri');
    }
    public function kelompok()
    {
        return $this->hasOne(DataKelompok::class, 'id_kelompok', 'id_kelompok');
    }
}
    
```

Gambar 10. Detail Kegiatan Admin Model

Hasil dari pengkodean dalam fitur rekap admin terhadap kegiatan santri. Gambar 10 adalah view dalam arsitektur *Model-View-Controller* (MVC) pada Laravel bertanggung jawab untuk menyajikan antarmuka pengguna, memungkinkan interaksi dan menampilkan data kepada pengguna [25]. Dalam konteks fitur "Rekap Pilih Periode," *view* ini dirancang untuk memfasilitasi pengguna dalam memilih rentang tanggal dan induk kegiatan untuk melakukan rekapitulasi data.

```

public function rekap()
{
    $master_kegiatan = MasterKegiatan::all();
    if ($GET['_token'] == 'xx') {
        $kegiatan = LogKegiatan::with(['detail', 'kegiatan', 'kelas', 'kelompok', 'pengaspu'])
            ->withCount([
                'detail as total_santri',
                'detail as total_hadir' => function ($query) {
                    $query->where('ket', '=', 'hadir');
                },
                'detail as total_tidak_hadir' => function ($query) {
                    $query->where('ket', '=', 'tidak_hadir');
                }
            ])
            ->orderBy('created_at', 'desc')
            ->limit(10)
            ->get();
    } else {
        $t1 = $GET['tanggal_awal'];
        $t2 = $GET['tanggal_akhir'];
        $kegiatan = LogKegiatan::with(['detail', 'kegiatan', 'kelas', 'kelompok', 'pengaspu']);
        if ($t1 != null && $t2 != null) {
            $kegiatan = $kegiatan->whereBetween('tanggal', [$t1, $t2]);
        }
        if ($GET['kegiatan'] != null) {
            // dd($GET['kegiatan']);
            $kegiatan = $kegiatan->where('id_kegiatan', $GET['kegiatan']);
        }
        $kegiatan = $kegiatan->withCount([
            'detail as total_santri',
            'detail as total_hadir' => function ($query) {
                $query->where('ket', '=', 'hadir');
            },
            'detail as total_tidak_hadir' => function ($query) {
                $query->where('ket', '=', 'tidak_hadir');
            }
        ])
        ->orderBy('created_at', 'desc')
        ->get();
    }
    // dd($kegiatan);
    return view('admin.kegiatan.rekap', compact('kegiatan', 'master_kegiatan'));
}

```

Gambar 11. Rekap Kegiatan Admin View

Gambar 11 *controller* dalam arsitektur MVC pada Laravel berperan sebagai jembatan antara model *data* dan *view*. Dalam konteks fitur "Rekap Pilih Periode," *controller* ini bertanggung jawab untuk menerima permintaan dari *view*, memproses logika bisnis terkait filter dan pengambilan data rekapitulasi, serta menyiapkan data yang diperlukan untuk ditampilkan kembali ke *view* [25]. Analisis terhadap kode *controller* ini mengungkapkan bagaimana data master kegiatan diambil, serta bagaimana rekapitulasi data Log Kegiatan dilakukan berdasarkan kriteria filter yang diberikan pengguna.

```

<center>
<div class="my-3">
    Pilih Periode
</div>
</center>
<hr>
<form action="{{ route('admin.kegiatan.rekap') }}" method="get">
<div class="row m-3">
    @csrf
<div class="col-4">
<label form="tanggal_awal" class="form-label">Tanggal Awal</label>
<input type="date" class="form-control" id="tanggal_awal"
    name="tanggal_awal" value="{{ old('tanggal_awal') }}">
</div>
<div class="col-4">
<label form="tanggal_akhir" class="form-label">Tanggal Akhir</label>
<input type="date" class="form-control" id="tanggal_akhir"
    name="tanggal_akhir" value="{{ old('tanggal_akhir') }}">
</div>
<div class="class col-4">
<label form="kegiatan" class="form-label">Induk Kegiatan</label>
<select class="form-select" id="kegiatan" name="kegiatan">
<option value="">Pilih Induk Kegiatan</option>
@foreach ($master_kegiatan as $item)
<option value="{{ $item->id_kegiatan }}"
    {{ old('kegiatan') == $item->id_kegiatan ? 'selected' : '' }}>
    {{ $item->nama_kegiatan }}</option>
</foreach>
</select>
</div>
<div class="col-4">
<label form="submit" class="form-label"></label>
<button type="submit" class="btn btn-primary form-control" id="submit"
    name="submit">Tampilkan</button>
</div>
</div>
</form>
</div>

```

Gambar 12. Rekap Kegiatan Admin Controller

Hasil dari pengkodean dalam fitur rekap admin terhadap kegiatan santri. Gambar 12 adalah model dalam arsitektur MVC pada Laravel yang merepresentasikan tabel *detail\_kelompok* dalam database dan bertanggung jawab untuk berinteraksi dengan tabel database [25]. Dalam konteks fitur "Rekap Pilih Periode," *Model*.

Detail Kelompok dirancang untuk memfasilitasi pengambilan dan pengelolaan data yang relevan dengan rentang waktu tertentu yang dipilih oleh admin.

## B. Pengujian

Pengujian dilakukan secara berkala dari setiap modul yang telah selesai dibuat. Kemudian dilakukan pengujian *blackbox* dengan skenario pengujian padat tabel 4

Tabel 4. Rancangan pengujian sistem tahfidz

| No | Deskripsi Pengujian                | Hasil yang diharapkan  |
|----|------------------------------------|--|
| T1 | <i>Login User</i>                  | <i>User</i> dapat <i>login</i> sesuai dengan role dan <i>username</i> password |
| T2 | Mengelola Data <i>User</i>         | Admin dapat menggolongkan data <i>user</i> kedalam kelas atau kelompok         |
| T3 | Mengelola Hak <i>User</i>          | Admin dapat <i>menginputkan</i> data <i>user</i> pengampu atau santri          |
| T4 | Membuat Laporan Rekapitulasi       | Admin dapat merekap aktifitas santri   |
| T5 | Melihat Aktifitas Tahfidz          | Walisantri dapat melihat aktifitas harian santri                               |
| T6 | Meihat Rekap Tahfidz Santri        | Walisantri dapat melihat rekap tahfidz santri                                  |
| T7 | Mencatat Tahfidz Santri            | Pengampu dapat <i>menginput</i> data hafalan santri sesuai dengan kondisinya   |
| T8 | Prensensi Aktifitas Tahfidz Santri | Pengampu dapat mempresensikan santri dari setiap aktifitas                     |
| T9 | Merekap Catatan Tahfidz            | Pengampu dapat menampilkan data tahfidz santri                                 |

Tabel 5. Hasil pengujian sistem tahfidz

| No | Hasil Pengujian  | Kesimpulan |
|----|--|------------|
| T1 | Pengguna 'Adm' berhasil <i>login</i> dengan password '12345678' dan diarahkan ke <i>dashboard</i> Admin. Pengguna 'pengampu5' berhasil <i>login</i> dengan password '12345678' dan diarahkan ke <i>dashboard</i> Pengampu. Walisantri masuk dengan generate dari NIS santri. | Berhasil   |
| T2 | Admin berhasil menambahkan <i>user</i> santri baru dan mengassign santri tersebut ke Kelompok A. Admin juga berhasil memindahkan santri dari Kelompok A ke Kelompok B.   | Berhasil   |
| T3 | Admin berhasil mengubah hak akses <i>user</i> Pengampu dari Aktif menjadi <i>Non-Aktif</i> .   | Berhasil   |
| T4 | Admin berhasil membuat laporan rekapitulasi aktivitas seluruh santri dari dari Waktu dan penggolongan Master Kegiatan. Laporan menampilkan data dengan benar.  | Berhasil   |
| T5 | Dapat menampilkan hasil aktifitas harian santri  | Berhasil   |
| T6 | Dapat menampilkan hasil rekap tahfidz santri   | Berhasil   |
| T7 | Pengampu 'PengampuA' berhasil <i>menginput</i> data setoran hafalan baru (Surat An-Nas, Ayat 1-6, Keterangan: Menambah) untuk santri 'SantriC'. Data tersimpan.  | Berhasil   |
| T8 | Pengampu 'PengampuB' mencoba melakukan presensi (menandai Hadir) untuk santri 'SantriD' pada aktivitas tahfidz tanggal 28 Mei 2025, namun setelah klik 'Simpan', status presensi tidak berubah dan kembali ke 'Belum dipresensi'.  | Berhasil   |
| T9 | Pengampu dapat menampilkan data tahfidz santri.  | Berhasil   |

Berdasarkan tabel 5 hasil pengujian dari 9 kasus uji yang dilakukan, 9 kasus uji menunjukkan hasil "Berhasil". Sistem menunjukkan tingkat fungsionalitas yang baik untuk sebagian besar skenario yang diuji.

### 3.4 Cutover phase

Setelah proses pengembangan dan pengujian tuntas, sistem kini memasuki tahap implementasi atau cutover. Sistem ini memanfaatkan dua platform utama yaitu aplikasi Android

yang ditujukan untuk para pengampu dan walisantri, serta platform website khusus untuk kebutuhan admin. Untuk sementara waktu, instalasi server dilakukan pada server lokal milik pengembang, yang nantinya akan dimigrasikan ke server yang lebih stabil dan mulai konsisten digunakan. Demi menjamin kenyamanan dan keamanan pengguna, khususnya walisantri, aplikasi Android akan didistribusikan melalui Google Play Store

#### 4. KESIMPULAN

Penelitian ini menghasilkan aplikasi pencatatan tahfidz berbasis Android menggunakan metode *Rapid Application Development* (RAD) untuk mengatasi permasalahan sistem manual di SMPIT Salman Alfarisi Boarding School, seperti data tercecer, ketidak teraturan pencatatan, dan kesulitan pelaporan. Aplikasi yang dihasilkan diharapkan mampu meningkatkan efektifitas pencatatan dan mempermudah akses data dengan hasil implementasi dan evaluasi pengguna yang terhadap kepraktisan serta kegunaan sistem. Sembilan kasus uji fungsionalitas sistem menunjukkan hasil "Berhasil", mengindikasikan bahwa aplikasi berjalan sesuai dengan rancangan dan kebutuhan fungsional pengguna yang meliputi pengampu, wali santri, dan admin. Sistem ini terdiri dari aplikasi Android untuk pengampu dan wali santri, serta platform *website* untuk admin. Dengan demikian, sistem ini berpotensi untuk diadopsi secara lebih luas oleh lembaga pendidikan sejenis yang menghadapi tantangan serupa.

#### 5. SARAN

Saran untuk pengembangan dan penyempurnaan lebih lanjut dapat difokuskan pada tiga area Utama, evaluasi mendalam, peningkatan fungsionalitas teknis, dan integrasi sistem. Pertama, penelitian lanjutan perlu mengukur dampak langsung penggunaan aplikasi secara kuantitatif dan kualitatif terhadap peningkatan kualitas hafalan santri dan motivasi belajar, mungkin melalui perbandingan dengan kelompok kontrol. Selain itu, meskipun evaluasi awal positif, disarankan untuk melakukan studi usabilitas yang lebih mendalam menggunakan metrik kuantitatif seperti *System Usability Scale* (SUS), tingkat kesalahan, dan waktu penyelesaian tugas, melibatkan sampel pengguna yang lebih beragam. Kedua, dari sisi teknis, perlu dikembangkan dan diuji fitur fungsionalitas offline yang robas, lengkap dengan mekanisme sinkronisasi data yang efisien, mengingat aplikasi saat ini bergantung pada koneksi internet via REST API. Fokus juga harus diperluas pada analisis risiko keamanan siber yang spesifik (termasuk pengujian penetrasi dan enkripsi yang lebih kuat) untuk melindungi data sensitif santri dan pengampu. Terakhir, untuk efisiensi administratif, penelitian selanjutnya sebaiknya merancang dan mengevaluasi model integrasi aplikasi tahfidz ini dengan Sistem Informasi Akademik (SIA) atau *Learning Management System* (LMS) lain yang sudah digunakan institusi sejenis

#### DAFTAR PUSTAKA

- [1] A. Angga dan S. Iskandar, "Kepemimpinan Kepala Sekolah dalam Mewujudkan Merdeka Belajar di Sekolah Dasar," *Jurnal Basicedu*, vol. 6, no. 3, hlm. 5295–5301, Mei 2022.
- [2] E. Mulia, S. Zakir, C. Rinjani, dan S. Annisa, "Kajian Konseptual Hasil Belajar Siswa dalam Berbagai Aspek dan Faktor yang Mempengaruhinya," 2021.
- [3] S. W. Nasution, "Assesment Kurikulum Merdeka Belajar Di Sekolah Dasar," *Prosiding Pendidikan Dasar*, 2021.
- [4] J. Pendidikan dan D. Konseling, "Analisis Kesulitan Guru Dalam Mengukur Aktivitas Belajar Siswa Pada Pembelajaran Jarak Jauh.," 2022

- [5] A. Rahmawati dan U. Islam Jakarta, “Pengaruh Kompetensi Guru terhadap Keberhasilan Proses Pembelajaran (Studi Survei di SMK Nurul Iman Jakarta),” *Jurnal Pendidikan dan Studi Islam*, vol. 5, no. 2, hlm. 114, 2024.
- [6] A. Arman, S. Sotar, dan K. Ulya, “Sistem Informasi Pencatatan Hafalan Al-Qur’an Pada SMP Islam Terpadu Mutiara Kota Pariaman Berbasis Website,” *Rang Teknik Journal*, vol. 4, no. 2, hlm. 316–324, Jun 2021.
- [7] P. Rahardiyanto dan A. Alfatiha, “Pengembangan Aplikasi Monitoring Evaluasi Pembelajaran Siswa SMP Negeri 2 Gempol Berbasis Android Dengan Metode Rad (Rapid Application Development),” 2022.
- [8] F. A. Purwanto, “Sistem Informasi Arsip Surat dengan Metode Rapid Application Development (RAD),” *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi*, vol. 3, hlm. 84–88, 2021.
- [9] M. Hasyimi dan I. Ali, “Rancang Bangun Aplikasi Wallpaper Berbasis Android Menggunakan Metode Rapid Application Development,” *Jurnal Device*, vol. 13, no. 1, hlm. 59–69, 2023.
- [10] James Martin, *Rapid Application Development by James Martin*. Macmillan Publishing Company, 1991.
- [11] H. Faqih, A. B. Hikmah, W. Azizah, S. I. Kampus, dan K. Tegal, “Implementasi Metode Rapid Application Development Pada Pengembangan Aplikasi e-Fin Mosque Z,” *Indonesian Journal on Software Engineering (IJSE)*, vol. 8, no. 1, 2022.
- [12] D. Prastyo dan D. Irawan, “Rancang Bangun Aplikasi Inspeksi dan Checklist Terintegrasi menggunakan Metode Rapid Application Development (RAD),” *bit-Tech*, vol. 7, no. 1, hlm. 162–173, Agu 2024.
- [13] V. F. Fuadiah, T. Yuniati, dan C. Ramdani, “Rancang Bangun E-Katalog Pada Perusahaan Distributor Produk Periklanan Menggunakan Metode Application Development (Virida Farkhatul Fuadiah) Rancang Bangun E-Katalog Pada Perusahaan Distributor Produk Periklanan Menggunakan Metode Rapid Application Development,” 2025.
- [14] Kenneth E. Kendall dan Julie E. Kendall, *Systems Analysis and Design*. Prentice Hall, 2006.
- [15] S. Sismadi, S. Dalis, S. Bahri, dan D. Setiawan, “Model Rapid Application Development Untuk Rancang Bangun Sistem Pengelolaan Transaksi Perdagangan Indosurrati Sukses Makmur,” *INTI Nusa Mandiri*, vol. 18, no. 1, hlm. 13–22, Agu 2023.
- [16] Ph. D. Roger S. Pressman, *Software Engineering: A Practitioner’s Approach*, Seventh. McGraw-Hill, 2010.
- [17] F. Rizky Maulidy dan D. Rahman Prehanto, “Rancang Bangun Aplikasi Penjualan Barang Toserba Menggunakan Metode OOAD (Object Oriented Analysis Design) Berbasis Mobile Platform Pada CV FCH Kema Pole Indonesia,” 2022.
- [18] Jamailah dan Arif Ali Ibrahim, “Rancang Bangun Aplikasi E-Commerce Furnitur Berbasis Android,” *Jurnal Ilmiah Teknik*, vol. 2, no. 2, 2023.
- [19] Bayu Rezky Ramadhan, Ahmad Faisol, dan Deddy Rudhistiar, “Rancang Bangun Sistem Ticketing Parkir Menggunakan Ktm Berbasis Android Menggunakan Restful Api,” 2023.
- [20] A. Amalia, S. W. Putri Hamidah, dan T. Kristanto, “Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web,” *Building of Informatics, Technology and Science (BITS)*, vol. 3, no. 3, hlm. 269–274, Des 2021.
- [21] W. Khafa Nofa, M. Rafly, dan N. Ichsan, “Rancang Bangun Aplikasi Website Penjualan Makanan Beku Menggunakan Laravel,” *Jurnal Teknik dan Science*, vol. 1, no. 2, hlm.125–132, 2022.
- [22] I. Dwi Jaya Tanjung, Y. Elin Yuspita, H. Antoni Musril, S. Derta, dan A. Kata Kunci “Perancangan Sistem Informasi Tahfidz Al-Qur’an Berbasis Web di Pesantren Modern Daar Al-Uluum Asahan Kisaran,” 2024.

- [23] Luther Aldo Christian dan Siska Narulita, “Perancangan User Interface (UI) Sistem Social Media Scheduler dengan Metode Rapid Application Development (RAD),” *Pixel :Jurnal Ilmiah Komputer Grafis*, vol. 17, hlm. 129–137, 2024.
  - [24] S. H. Ali Mansyur, R. Dwicahya Supriatman, dan D. Mulyana, “Aplikasi Laundry Berbasis Website Menggunakan Metode Rapid Application Development (RAD) Pada LC Jaya Clean,” 2024.
  - [25] Ambriani dan A. I. Nurhidayat, “Rancang Bangun Repository Publikasi Ilmiah Dosen Berbasis Web Menggunakan Framework Laravel,” 2020.
-