

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Kedudukan selama menjalani proses kerja magang di PT Matica Inovasi Edukasi adalah sebagai *Mobile Application Developer Intern*. Tanggung jawab utama adalah merancang bangun aplikasi *Learning Management System (LMS)* berbasis Android. Posisi ini dijalankan di bawah arahan dan bimbingan Bapak Legi Hendra. Koordinasi dengan berbagai tim, termasuk tim pengembangan aplikasi, dilakukan secara daring melalui platform komunikasi WhatsApp dan Zoom. Proses rancang bangun dilakukan menggunakan *Android Studio* sebagai *Integrated Development Environment (IDE)* utama, yang mendukung pembuatan aplikasi native dengan performa tinggi. Proses rancang bangun dimulai dari tahap perancangan antarmuka pengguna berdasarkan desain Figma, penghubungan dengan backend menggunakan *Firebase* untuk autentikasi dan penyimpanan data serta penghubungan dengan backend menggunakan *Supabase* untuk pengelolaan file.

Departemen IT di PT Matica Inovasi Edukasi menerapkan sistem kerja berbasis branch dan pull request untuk memantau perkembangan setiap tugas anggota tim. Setelah seorang developer menyelesaikan pekerjaannya di cabang (branch) tertentu, permintaan penggabungan (pull request) akan diajukan dan ditinjau terlebih dahulu oleh product owner. Proses ini memastikan bahwa setiap perubahan kode telah diperiksa dengan baik sebelum akhirnya digabungkan ke sistem utama.

3.2 Tugas yang Dilakukan

Dalam pelaksanaan kerja magang di PT Matica Inovasi edukasi sebagai *Mobile Application Developer Intern*, keterlibatan berfokus pada pengembangan aplikasi *Learning Management System (LMS)* berbasis Android yang dirancang untuk memfasilitasi kegiatan belajar - mengajar digital. Dalam kerja magang ini, peran yang dijalankan adalah sebagai *Mobile Fullstack Developer*, yang mencakup pengembangan aplikasi secara menyeluruh, baik pada bagian antarmuka pengguna (frontend) maupun logika dan layanan sistem (backend). Proses rancang bangun aplikasi dilakukan dengan menggunakan *Android Studio* sebagai *Integrated Development Environment (IDE)*, dengan bahasa pemrograman Kotlin. Untuk mendukung fungsi backend, *Firebase* digunakan sebagai layanan utama dalam

mengelola autentikasi pengguna dan penyimpanan data, seperti informasi akun, catatan absensi, dan hasil asesmen. Selain itu, *Supabase* dimanfaatkan sebagai tempat penyimpanan dokumen, terutama untuk fitur unggah file. *Supabase* telah diatur agar terhubung dengan *Firestore Authentication*, sehingga akses ke file dapat dibatasi sesuai peran masing-masing pengguna.

Pelaksanaan kerja magang mencakup rancang bangun antarmuka dan logika backend untuk beberapa bagian utama aplikasi, antara lain halaman *Splash Screen*, *Onboarding*, serta seluruh halaman yang digunakan oleh pengguna dengan peran sebagai *Teacher* dan *Admin*. Fitur - fitur yang dirancang meliputi tampilan awal aplikasi, alur pengenalan penggunaan aplikasi bagi pengguna baru, manajemen data guru, siswa dan *parent*, pengisian asesmen, unggahan materi belajar (*notes*), serta unggahan latihan soal (*exercise*). Seluruh proses rancang bangun dilakukan secara terstruktur, dimulai dari perancangan antarmuka di *Figma*, inisialisasi proyek menggunakan *Android Studio*, hingga implementasi dan pengujian fungsionalitas secara menyeluruh. Pengembangan juga memperhatikan integrasi antar halaman dan penyesuaian berdasarkan role pengguna, guna memastikan pengalaman penggunaan yang intuitif dan stabil di setiap fitur yang disediakan.

3.3 Uraian Pelaksanaan Magang

Tabel 3.1 memperlihatkan rincian jadwal kegiatan yang dilaksanakan setiap minggunya selama masa magang di PT Matica Inovasi Edukasi.

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu selama periode magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Melakukan orientasi proyek, memahami kebutuhan sistem LMS, mengidentifikasi peran pengguna (<i>admin</i> , <i>teacher</i> , <i>student</i> , <i>parent</i>), serta menyusun struktur navigasi awal. Membuat wireframe awal dalam bentuk sketsa kasar di kertas sebagai dasar struktur antarmuka aplikasi
2	Melakukan riset referensi UI/UX, menerapkan desain UI/UX awal ke dalam <i>Figma</i> untuk <i>splash screen</i> , <i>onboarding</i> , dan navigasi utama serta mendiskusikan hasil desain awal dengan tim dan supervisor untuk validasi alur tampilan dan konsistensi visual.
Lanjut pada halaman berikutnya	

Lanjutan Tabel 3.1

Minggu Ke -	Pekerjaan yang dilakukan
3	Mendesain halaman admin yang mencakup fitur tambah, edit, dan hapus data untuk student, teacher, dan parent. Selain itu, disiapkan alur penghubungan akun student ke parent melalui UI yang terstruktur
4	Membuat desain halaman teacher yang terdiri dari halaman daftar siswa (<i>student list</i>), laporan perkembangan siswa (<i>assessment</i>), serta struktur awal untuk fitur upload catatan dan file latihan.
5	Melakukan perbaikan desain berdasarkan <i>feedback</i> yang diberikan oleh tim dan supervisor, termasuk penyempurnaan warna, navigasi, dan konsistensi komponen. Prototipe disusun lengkap di Figma untuk semua halaman admin dan teacher.
6	Mengimplementasikan hasil desain ke dalam project Android Studio serta mulai setup struktur proyek. Dilakukan inisialisasi tampilan splash screen, onboarding, dan halaman login.
7	Mengembangkan halaman admin untuk fitur tambah, edit, dan hapus data <i>student, teacher, dan parent</i> . Setiap data disimpan sementara secara lokal untuk pengujian fungsionalitas awal.
8	Menyusun relasi data untuk menghubungkan akun <i>student</i> dan <i>parent</i> pada halaman admin. Tampilan form dan alur penyimpanan dibuat agar mudah dimengerti oleh pengguna.
9	Menginisiasi integrasi <i>Firebase</i> sebagai backend utama aplikasi. Melakukan konfigurasi proyek <i>Firebase</i> di <i>Android Studio</i> untuk mendukung fitur autentikasi dan penyimpanan data. Selain itu, mulai menyusun struktur data di <i>Firebase Realtime Database</i> agar sesuai dengan kebutuhan halaman admin dan teacher.
10	Mengimplementasikan fitur autentikasi login menggunakan <i>Firebase Authentication</i> , termasuk login berbasis peran (admin dan teacher). Setiap pengguna akan dikenali melalui input role dan diarahkan ke tampilan sesuai rolenya.
	Lanjut pada halaman berikutnya

Lanjutan Tabel 3.1

Minggu Ke -	Pekerjaan yang dilakukan
11	Menyambungkan data pengguna (akun student, teacher, dan parent) dari halaman admin ke <i>Firestore</i> . Admin dapat menyimpan, mengedit, dan menghapus data yang akan langsung tersimpan di <i>Realtime Database</i> . Relasi data seperti penghubungan siswa ke orang tua juga mulai diterapkan pada struktur database.
12	Mengembangkan halaman teacher dengan fitur daftar siswa yang diambil langsung dari <i>Firestore</i> berdasarkan relasi yang telah ditentukan oleh admin. UI dibuat agar dinamis dan responsif, sehingga memudahkan teacher dalam menelusuri data siswa secara <i>real-time</i> .
13	Mengembangkan fitur form assessment untuk teacher. Hasil penilaian siswa disimpan ke <i>Firestore</i> secara terstruktur berdasarkan ID siswa dan mata pelajaran. Desain form mempertimbangkan kemudahan input sekaligus kejelasan tampilan data saat ditinjau ulang.
14	Menambahkan fitur unggah <i>file notes dan exercise</i> (PDF) untuk teacher. Untuk kebutuhan penyimpanan file, mulai dilakukan integrasi <i>Supabase</i> sebagai pelengkap <i>Firestore</i> . <i>Supabase</i> digunakan untuk <i>file storage</i> karena mendukung akses langsung ke berkas dan efisien dalam manajemen dokumen.
15	Menyempurnakan integrasi <i>Supabase</i> untuk menampilkan file PDF yang diunggah oleh parent ke halaman teacher. Tampilan daftar file diurut berdasarkan nama siswa dan waktu unggah, serta disesuaikan dengan struktur penyimpanan di <i>Supabase</i> agar mudah dikelola.
16	Melakukan pengujian menyeluruh pada seluruh fitur, baik di halaman admin maupun teacher. Fokus pengujian mencakup autentikasi, manajemen data pengguna, pengisian <i>assessment</i> , dan proses upload/download file. Ditemukan beberapa <i>bug minor</i> dan langsung diperbaiki untuk memastikan aplikasi berjalan stabil.
Lanjut pada halaman berikutnya	

Lanjutan Tabel 3.1

Minggu Ke -	Pekerjaan yang dilakukan
17	Menyusun dokumentasi proyek secara lengkap, meliputi struktur sistem, alur fitur, relasi data, serta proses integrasi <i>Firestore</i> dan <i>Supabase</i> . Dokumentasi ini dilampirkan sebagai bagian dari laporan magang dan menjadi acuan untuk pengembangan versi selanjutnya.

3.4 Analisis dan Perancangan Sistem

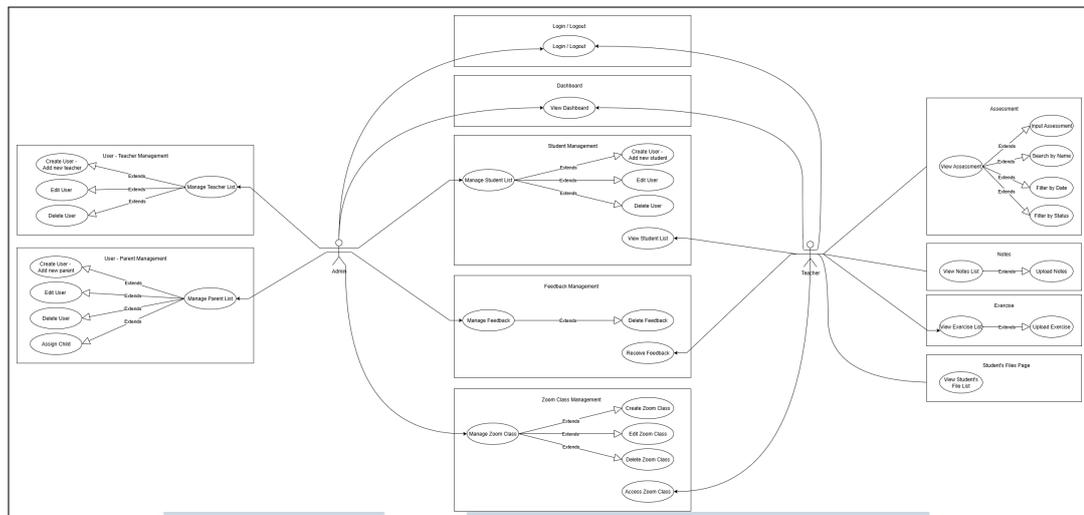
Uraian dibawah ini menjelaskan tentang Rancang bangun platform LMS berbasis Android di PT Matica Inovasi Edukasi yang mencakup diagram seperti *Use Case* dan *Sequence*, serta Struktur Tabel, hasil implementasi berupa tangkapan layar tampilan aplikasi, dan Pengujian Sistem.

3.4.1 Use Case Diagram

Diagram use case merupakan elemen penting dalam pemodelan perangkat lunak yang berfungsi untuk menggambarkan interaksi antara aktor (baik pengguna maupun sistem luar) dengan sistem yang sedang dikembangkan. Diagram ini menggambarkan fungsi-fungsi sistem dari sudut pandang pengguna, memperlihatkan interaksi aktor dengan sistem untuk mencapai tujuan yang spesifik [16]. Melalui diagram ini, pengembang dapat mengerti dan merencanakan tujuan serta fungsi sistem dengan lebih fokus, sehingga memperkuat proses desain dan pengembangan sistem secara keseluruhan [17].

Meskipun Use Case Diagram pada aplikasi LMS PT Matica Inovasi Edukasi mencakup empat aktor utama, yaitu Teacher, Student, Parent, dan Admin, namun pada laporan ini hanya akan dijelaskan bagian Admin dan Teacher, karena pengembangan fitur yang berkaitan dengan Parent dan Student dikerjakan oleh anggota tim lainnya.

Use Case Diagram menggambarkan interaksi guru dan admin dengan Learning Management System (LMS) pada aplikasi mobile PT Matica Inovasi Edukasi, seperti yang terlihat pada Gambar 3.1.



Gambar 3.1. Use Case Diagram Aplikasi LMS PT Matica Inovasi Edukasi

A Teacher

Teacher atau guru memiliki akses terhadap berbagai fitur yang berkaitan dengan proses pembelajaran dan evaluasi siswa dalam sistem *Learning Management System* (LMS). Setelah melakukan login, guru dapat melihat dashboard untuk melihat fitur (menu) yang dapat diakses. Salah satu fitur utama yang dapat digunakan guru adalah pengelolaan assessment siswa, di mana guru dapat melihat data penilaian, melakukan *input assessment*, serta memanfaatkan fitur pencarian dan filter berdasarkan nama, tanggal, dan status penilaian guna mempermudah proses evaluasi. Selain itu, guru juga dapat mengunggah materi pembelajaran dan latihan dalam bentuk file PDF melalui fitur *Upload Notes* dan *Upload Exercise*. File tersebut dikelola melalui tampilan daftar masing-masing, yakni *Notes List* dan *Exercise List*, agar dapat diorganisasi dengan baik. Guru juga memiliki akses terhadap *Student's File Page*, yang memungkinkan mereka melihat file yang telah dikirimkan oleh orang tua atau siswa.

B Admin

Admin dapat melakukan proses autentikasi melalui fitur *login* dan *logout*, serta mengakses dashboard untuk melihat informasi penting dari sistem. Dalam hal pengelolaan pengguna, Admin memiliki akses penuh untuk menambah, mengedit, dan menghapus data siswa, guru, dan orang tua melalui fitur manajemen yang terpisah, yaitu *Manage Student List*, *Manage Teacher List*, dan *Manage Parent List*. Khusus pada manajemen orang tua, Admin juga dapat menghubungkan

akun orang tua dengan akun siswa yang bersangkutan. Selain itu, Admin juga berperan dalam mengelola masukan dari pengguna melalui fitur *Manage Feedback*, yang memungkinkan Admin menghapus *feedback* yang dianggap tidak relevan atau telah ditindaklanjuti. Admin juga menangani pengaturan kelas daring melalui fitur *Manage Zoom Class*, yang mencakup pembuatan, pengeditan, dan penghapusan sesi Zoom untuk keperluan pembelajaran. Secara keseluruhan, peran Admin sangat sentral dalam menjaga keteraturan data dan kelancaran proses operasional sistem.

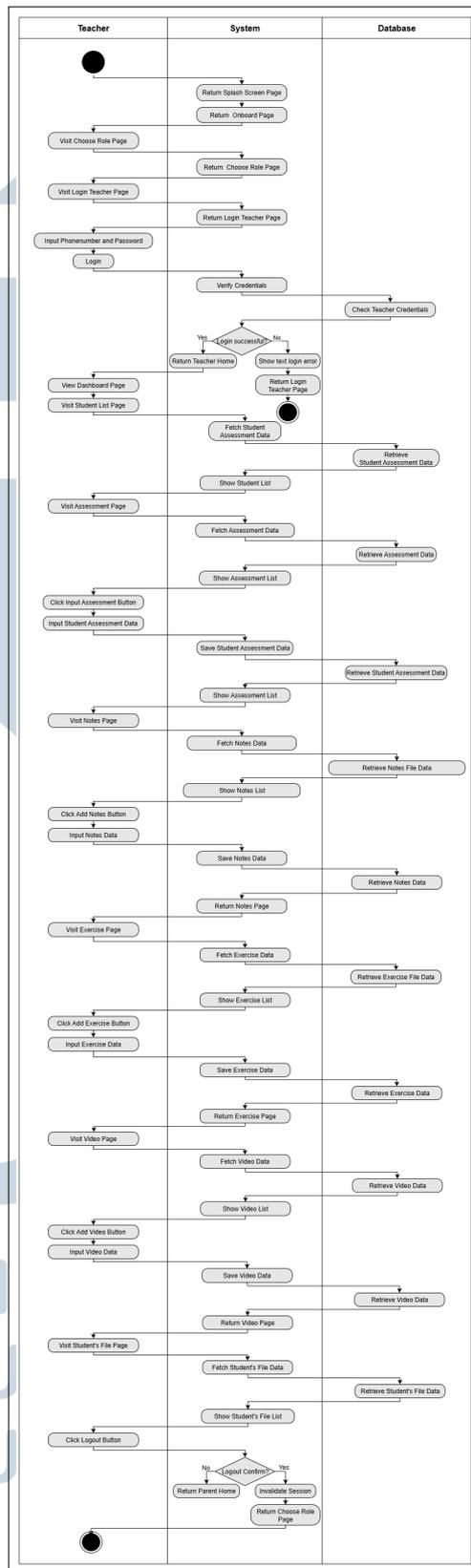
3.4.2 Activity Diagram

Activity Diagram berfungsi untuk menggambarkan alur aktivitas atau proses bisnis dalam sistem secara terstruktur dan logis. Diagram ini menampilkan urutan aktivitas yang dilakukan oleh aktor maupun sistem, termasuk kondisi percabangan, keputusan, dan aliran kerja yang terjadi. Dalam aplikasi LMS PT Matica Inovasi Edukasi, Activity Diagram digunakan untuk memvisualisasikan proses-proses utama yang dilakukan oleh pengguna, terutama pada aktivitas yang melibatkan interaksi kompleks antara admin dan sistem, sehingga memberikan gambaran menyeluruh mengenai logika alur kerja dalam aplikasi.

A Teacher

Activity Diagram Teacher menggambarkan seluruh alur proses dari awal pengguna mengakses aplikasi hingga proses keluar (logout), termasuk interaksi dengan sistem dan basis data, seperti yang terlihat pada Gambar 3.2.

Diagram ini terdiri atas tiga swimlane, yaitu Teacher, System, dan Database. Swimlane pertama mencerminkan aktivitas yang dilakukan langsung oleh guru sebagai pengguna sistem. Swimlane kedua menunjukkan bagaimana sistem merespons setiap tindakan pengguna, termasuk pengambilan keputusan, validasi data, dan pengelolaan navigasi halaman. Swimlane ketiga berfungsi merepresentasikan proses pengambilan dan penyimpanan data ke dalam basis data.



Gambar 3.2. Activity Diagram Guru Aplikasi LMS PT Matica Inovasi Edukasi

Proses dimulai ketika guru membuka aplikasi dan diarahkan ke halaman splash screen, onboarding, dan pemilihan peran. Setelah memilih peran sebagai guru dan mengisi nomor telepon serta kata sandi, sistem akan memverifikasi kredensial pengguna dengan mencocokkannya ke dalam basis data. Jika data cocok, pengguna diarahkan ke halaman utama guru; jika tidak, sistem akan menampilkan pesan kesalahan.

Setelah berhasil masuk, guru dapat mengakses berbagai fitur, seperti melihat daftar siswa dan penilaiannya, membuka halaman assessment, dan menambahkan penilaian siswa. Selain itu, guru juga dapat mengelola materi pembelajaran seperti catatan, latihan soal, dan video. Untuk setiap proses input data, sistem akan menyimpan informasi ke dalam basis data dan menampilkan hasilnya kembali dalam bentuk daftar pada halaman masing-masing. Guru juga dapat melihat file yang diunggah oleh siswa sebelum mengakhiri sesi dengan menekan tombol logout. Sistem kemudian akan mengonfirmasi proses logout dan mengembalikan pengguna ke halaman pemilihan peran.

Activity Diagram ini membantu menggambarkan secara visual dan terstruktur bagaimana alur interaksi antara guru, sistem, dan basis data berlangsung dalam aplikasi, serta memperjelas tahapan proses yang harus diikuti dalam penggunaan aplikasi oleh guru.

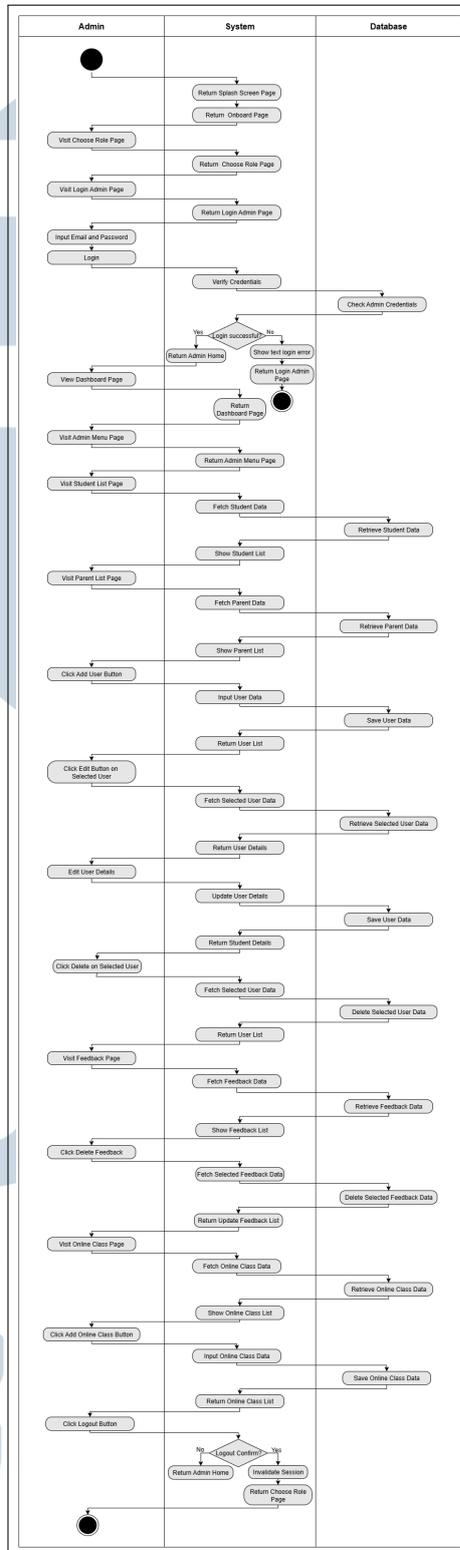
B Admin

Activity Diagram Admin menggambarkan seluruh alur proses dari awal pengguna mengakses aplikasi hingga proses keluar (logout), termasuk interaksi dengan sistem dan basis data, seperti yang terlihat pada Gambar 3.3.

Activity Diagram pada aktor Admin bertujuan untuk menggambarkan keseluruhan alur proses penggunaan sistem oleh admin, dari awal masuk hingga keluar dari aplikasi. Diagram ini terdiri atas tiga swimlane, yaitu Admin, System, dan Database. Swimlane pertama menggambarkan aktivitas yang dilakukan langsung oleh admin, swimlane kedua menunjukkan bagaimana sistem merespons interaksi tersebut, sedangkan swimlane ketiga berfungsi untuk menangani proses pengambilan dan penyimpanan data ke basis data.

Proses dimulai ketika admin membuka aplikasi dan diarahkan ke halaman splash screen, onboarding, dan halaman pemilihan peran. Setelah memilih peran sebagai admin dan mengisi email serta kata sandi, sistem akan memverifikasi kredensial pengguna dengan mencocokkannya ke basis data. Jika berhasil, admin akan

diarahkan ke halaman beranda admin.



Gambar 3.3. Activity Diagram Admin Aplikasi LMS PT Matica Inovasi Edukasi

Setelah berhasil masuk, admin dapat mengakses berbagai fitur utama sistem. Admin dapat membuka menu admin, melihat daftar siswa, dan melihat daftar orang tua. Admin juga memiliki wewenang untuk menambahkan pengguna baru, yang mencakup proses pengisian data pengguna dan penyimpanan ke basis data, serta dapat mengedit data pengguna yang telah dipilih dan menyimpannya kembali. Selain itu, admin juga dapat menghapus data pengguna yang sudah ada dari sistem. Admin dapat mengakses halaman feedback untuk melihat umpan balik dari pengguna, mengambil data feedback dari basis data, dan memiliki kemampuan untuk menghapus umpan balik tertentu yang dipilih. Di samping itu, admin juga dapat mengelola data kelas daring (online class), termasuk melihat daftar kelas daring dan menambahkan data kelas baru ke sistem.

Pada akhir sesi, admin dapat memilih untuk keluar dari sistem melalui tombol logout. Sistem kemudian akan mengonfirmasi tindakan logout dan jika dikonfirmasi, sesi akan diakhiri dan admin akan diarahkan kembali ke halaman pemilihan peran.

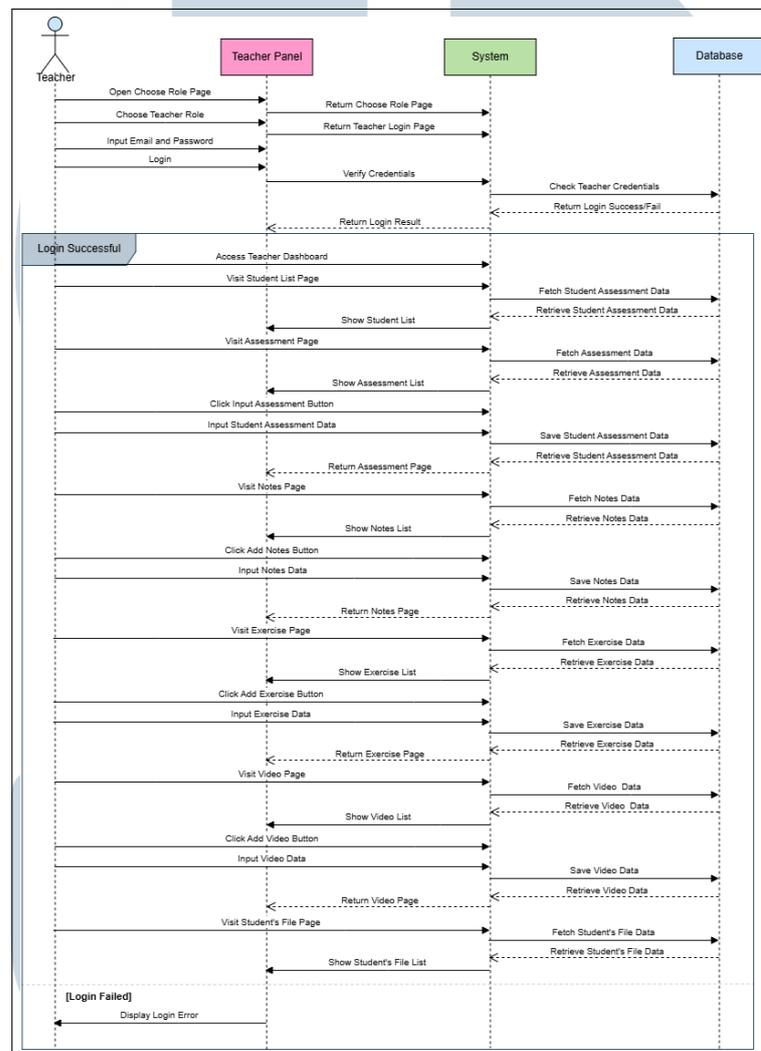
3.4.3 Sequence Diagram

Sequence Diagram berfungsi untuk menggambarkan interaksi dinamis antara objek atau elemen dalam sebuah sistem perangkat lunak. Diagram ini memperlihatkan rangkaian pengiriman pesan antara aktor dan objek sepanjang suatu proses berlangsung, sehingga memberikan pemahaman yang lebih jelas mengenai alur komunikasi di antara komponen dalam sistem. Di Aplikasi LMS PT Matica Inovasi Edukasi, *Sequence Diagram* diterapkan untuk memvisualisasikan interaksi antara pengguna dan sistem serta interaksi yang lebih rumit antara admin dan sistem.

A Teacher

Sequence Diagram Teacher menggambarkan urutan interaksi antara guru dan aplikasi LMS PT Matica Inovasi Edukasi. Sequence diagram untuk Teacher menggambarkan alur interaksi antara pengguna dengan peran sebagai guru (teacher) dan sistem *Learning Management System (LMS)* pada PT Matica Inovasi Edukasi. Proses dimulai saat guru membuka halaman pemilihan role dan memilih role sebagai Guru. Setelah itu, guru diharuskan menuju halaman login dan menginput email serta kata sandi untuk proses verifikasi. Sistem lalu memeriksa kredensial dengan mencocokkan informasi yang ada dalam database. Jika login berhasil,

guru akan diarahkan ke dashboard utama. Dalam dashboard, guru bisa mengakses berbagai fitur seperti melihat daftar murid, membuka halaman evaluasi, menambah data evaluasi, serta menyimpan informasi tersebut ke dalam basis data, seperti yang terlihat pada Gambar 3.4.



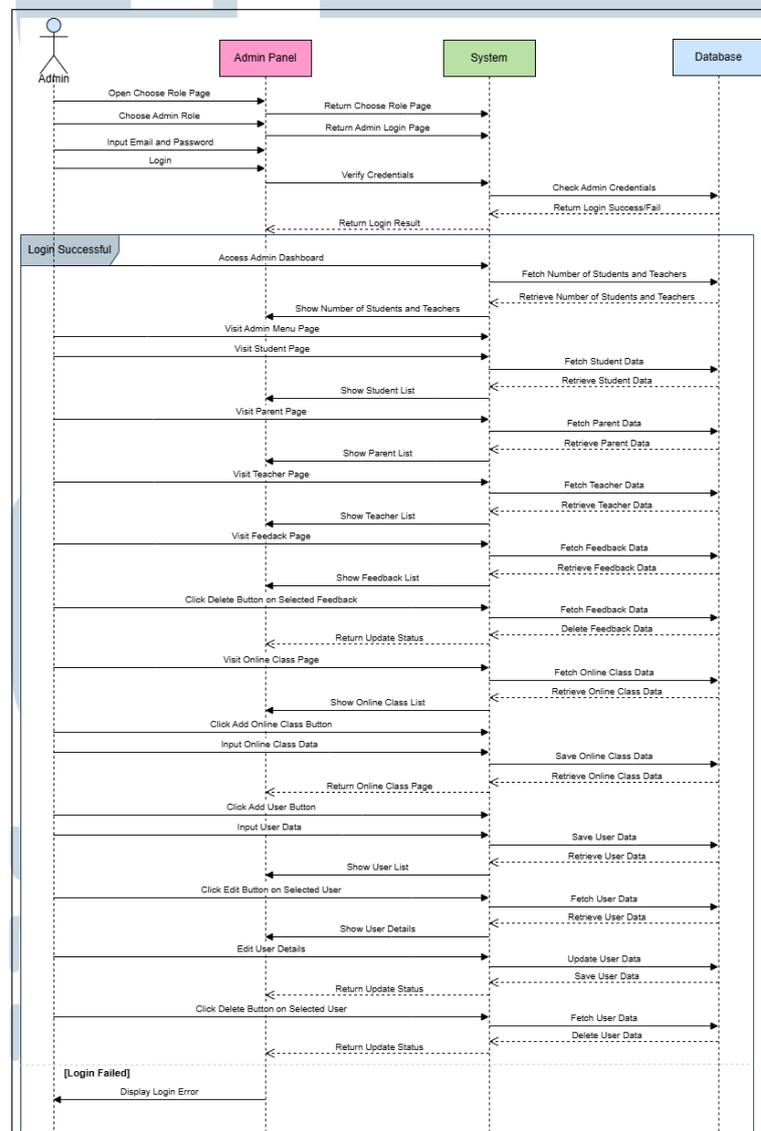
Gambar 3.4. Sequence Diagram Teacher Aplikasi LMS PT Matica Inovasi Edukasi

Guru juga dapat mengakses halaman catatan siswa (notes), menambahkan catatan baru, dan menyimpannya. Di samping itu, pengajar dapat mengakses halaman latihan, menambah latihan, dan membuka halaman video pembelajaran untuk tambah konten video. Setiap tindakan yang diambil oleh guru, baik melihat maupun menambah data, akan mendorong sistem untuk mengakses atau menyimpan informasi ke dalam basis data sesuai kebutuhan. Di akhir proses, guru pun bisa mengunjungi halaman berkas siswa untuk memeriksa daftar file yang sudah diunggah. Proses ini menjamin bahwa guru dapat menjalankan fungsinya secara

komprehensif dalam mengatur aktivitas pembelajaran siswa dalam sistem LMS yang dirancang.

B Admin

Sequence Diagram Admin menggambarkan urutan interaksi antara admin dan aplikasi LMS PT Matica Inovasi Edukasi, seperti yang terlihat pada Gambar 3.9. Sequence diagram ini menggambarkan alur interaksi antara aktor Admin dengan tiga komponen utama, yaitu Admin Panel, System dan Database, dalam rangka mengelola berbagai fitur pada aplikasi LMS.



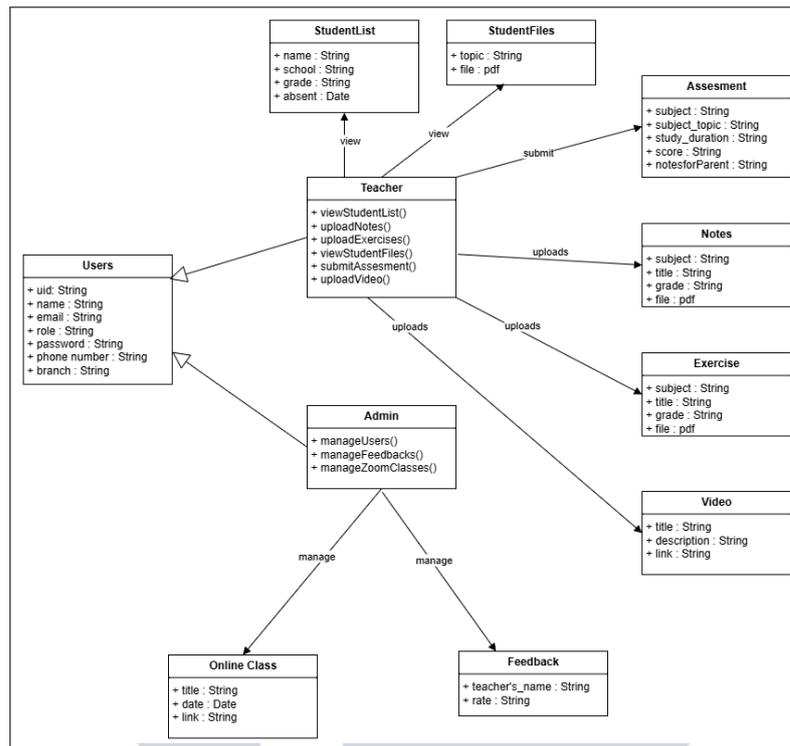
Gambar 3.5. Sequence Diagram Admin Aplikasi LMS PT Matica Inovasi Edukasi

Proses dimulai ketika Admin mengakses halaman pemilihan role dan memilih role sebagai admin, kemudian diarahkan ke halaman login untuk mengisi email dan password. Kredensial yang dimasukkan diperiksa oleh sistem dengan melakukan verifikasi ke database. Apabila data valid, halaman dashboard akan ditampilkan oleh sistem, tetapi jika tidak, pesan kesalahan login akan ditampilkan. Setelah berhasil masuk, Admin bisa mengakses beragam menu seperti Halaman Siswa, Halaman Orang Tua, dan Halaman Guru untuk melihat daftar pengguna berdasarkan kategori masing-masing. Informasi pengguna ditampilkan setelah diambil dari basis data. Selain itu, Admin juga dapat mengakses halaman Feedback untuk melihat serta menghapus tanggapan dari pengguna. Prosedur penghapusan dilaksanakan dengan mengumpulkan informasi yang relevan dan memperbaruinya di basis data. Di halaman kelas daring, Admin dapat melihat daftar kelas yang ada dan juga menambah kelas baru dengan mengisi informasi yang diperlukan. Data tersebut selanjutnya disimpan ke dalam database dan daftar kelas akan diperbaharui. Admin juga berhak untuk mengelola data pengguna, termasuk menambah pengguna baru, mengubah informasi pengguna yang ada, hingga menghapus pengguna. Seluruh interaksi tersebut melibatkan proses pengumpulan dan penyimpanan informasi ke dalam basis data.

3.4.4 Class Diagram

Dalam aplikasi *Learning Management System (LMS)* yang dikembangkan untuk PT Matica Inovasi Edukasi, *Class Diagram* digunakan sebagai representasi visual dari struktur statis sistem, dengan tujuan untuk memetakan entitas-entitas utama yang terlibat dalam proses pembelajaran digital. Diagram ini mencakup kelas-kelas inti seperti *Admin*, *Teacher*, *Student*, dan *Parent*, serta sejumlah entitas pendukung yang berperan dalam alur informasi dan pengelolaan data, seperti *Assessment*, *Notes*, *Exercise*, *Video*, *Online Class*, *Feedback*, dan *StudentFiles*. Masing-masing kelas dalam diagram ini terdiri dari atribut (field) yang merepresentasikan data yang disimpan oleh sistem dan metode (function) yang menggambarkan fungsionalitas atau tindakan yang dapat dilakukan oleh objek dari kelas tersebut.

Namun, karena pengembangan sistem ini dibagi ke dalam beberapa tanggung jawab tim, maka pembahasan *Class Diagram* dalam laporan ini akan difokuskan pada dua aktor utama yang menjadi lingkup pengerjaan penulis, yaitu *Admin* dan *Teacher*. *Class Diagram Teacher* dan *Admin* dapat dilihat pada Gambar 3.6.



Gambar 3.6. Class Diagram Aplikasi LMS PT Matica Inovasi Edukasi

A Teacher

Pada class diagram, *Teacher* merupakan subclass dari *Users* yang memiliki peran utama dalam pengelolaan aktivitas pembelajaran. *Teacher* dapat menjalankan berbagai metode, antara lain *viewStudentList()* untuk melihat daftar siswa, *uploadNotes()* untuk mengunggah catatan materi, *uploadExercises()* untuk menambahkan latihan berupa soal, *viewStudentFiles()* untuk mengakses file yang dikumpulkan siswa, *submitAssessment()* untuk memberikan penilaian siswa, serta *uploadVideo()* untuk membagikan materi pembelajaran dalam bentuk video. Class ini terhubung dengan class *Notes*, *Exercise*, dan *Video*, yang masing-masing menyimpan data materi pembelajaran, serta dengan class *Assessment* yang menampung informasi penilaian seperti subjek, durasi belajar, skor, dan catatan untuk orang tua. Selain itu, *Teacher* juga berinteraksi dengan class *StudentList* dan *StudentFiles* untuk memantau absensi siswa dan file tugas yang dikumpulkan.

B Admin

Admin juga merupakan subclass dari *Users* yang bertanggung jawab dalam mengelola keseluruhan sistem. *Admin* memiliki tiga metode utama, yaitu

manageUsers() untuk mengelola data pengguna seperti guru, siswa, dan orang tua; *manageFeedbacks()* untuk menangani tanggapan yang masuk dari pengguna; dan *manageZoomClasses()* untuk mengatur data kelas daring yang ditampilkan dalam class *Online Class*. Class Admin berinteraksi langsung dengan class *Feedback* dan *Online Class*, yang masing-masing menyimpan data komentar pengguna dan informasi kelas seperti judul, tanggal, dan tautan Zoom. Dengan peran ini, Admin bertindak sebagai pengelola utama yang memastikan sistem berjalan dengan baik dari sisi operasional dan komunikasi.

3.4.5 Struktur Tabel Database

Struktur database yang dibentuk dalam pengembangan Aplikasi Learning Management System (LMS) PT Matica Inovasi Edukasi terdiri dari tabel - tabel yang berisikan field yang digunakan untuk mendukung kebutuhan aplikasi. Setiap tabel memiliki atribut yang sesuai dengan kebutuhan aplikasi.

A Table teachers

Tabel 3.2 mendeskripsikan struktur tabel *teachers* yang berfungsi untuk menyimpan informasi seluruh pengguna yang terdaftar dalam sistem. Tabel ini berisi data seperti nama, cabang, alamat email, no telepon, kata sandi, role dan uid. Selain itu, tabel ini digunakan untuk mendefinisikan pengguna (*user*) yang akan digunakan untuk login pada Halaman Guru untuk mengakses LMS.

Tabel 3.2. Struktur Tabel *teachers*

Nama Atribut	Tipe Data
uid	string
name	string
branch	string
email	string
phone	string
password	string
role	string

Penjelasan Atribut Tabel *teachers*:

1. **uid**
Menyimpan nomor identitas unik untuk setiap pengguna dalam sistem.
2. **name**
Menyimpan nama lengkap pengguna.
3. **branch**
Menyimpan informasi cabang tempat pengguna terdaftar.
4. **email**
Menyimpan alamat email pengguna yang digunakan untuk login dan komunikasi.
5. **phone**
Menyimpan nomor telepon pengguna untuk keperluan kontak.
6. **password**
Menyimpan kata sandi akun pengguna untuk login ke dalam sistem.
7. **role**
Menyimpan peran pengguna dalam sistem, seperti admin, guru, siswa, atau orangtua.

B Students

Table 3.3 mendeskripsikan struktur tabel `students`, yang digunakan untuk menyimpan informasi siswa yang terdaftar dalam sistem. Informasi yang disimpan meliputi nama siswa, jenis kelamin, sekolah asal, jenjang kelas, serta referensi pengguna yang terkait. Tabel ini digunakan untuk menampilkan daftar siswa kepada guru dan orangtua melalui antarmuka aplikasi. Data dalam tabel ini juga digunakan dalam proses absensi, penilaian, serta pelacakan perkembangan akademik siswa.

Tabel 3.3. Struktur Tabel students

Nama Atribut	Tipe Data
branch	string
email	string
gender	string
name	string
phone	string
role	string
uid	string

Penjelasan Atribut Tabel students:

1. **id**
Menyimpan nomor identitas unik untuk setiap siswa dalam sistem.
2. **name**
Menyimpan nama lengkap siswa.
3. **gender**
Menyimpan jenis kelamin siswa, seperti Male atau Female.
4. **school**
Menyimpan nama sekolah tempat siswa belajar.
5. **grade**
Menyimpan informasi jenjang kelas siswa, seperti Grade 1, Grade 2, dan seterusnya.
6. **uid**
Menyimpan kode identitas pengguna yang terhubung dengan akun siswa, biasanya berasal dari sistem autentikasi.

C Parents

Tabel 3.4 mendeskripsikan struktur tabel parents, yang digunakan untuk menyimpan informasi akun orangtua siswa dalam sistem. Informasi yang disimpan meliputi nama, jenis kelamin, email, nomor telepon, cabang tempat terdaftar, serta daftar anak yang diasuh. Tabel ini berfungsi untuk menunjukkan informasi orangtua

di halaman akun orangtua, serta untuk mengaitkan mereka dengan akun siswa yang mereka asuh. Setiap atribut pada tabel `parents` menggambarkan satu akun orang tua, yang bisa terhubung dengan satu atau lebih siswa melalui atribut `children`.

Tabel 3.4. Struktur Tabel `parents`

Nama Atribut	Tipe Data
<code>branch</code>	string
<code>children</code>	array of string
<code>email</code>	string
<code>gender</code>	string
<code>name</code>	string
<code>phone</code>	string
<code>role</code>	string
<code>uid</code>	string

Penjelasan Atribut Tabel `parents`:

1. **`branch`**
Menyimpan nama cabang tempat orangtua atau anak terdaftar.
2. **`children`**
Menyimpan daftar UID siswa yang diasuh oleh orangtua tersebut.
3. **`email`**
Menyimpan alamat email orangtua.
4. **`gender`**
Menyimpan jenis kelamin orangtua, seperti `male` atau `female`.
5. **`name`**
Menyimpan nama lengkap orangtua.
6. **`phone`**
Menyimpan nomor telepon orangtua untuk keperluan kontak.
7. **`role`**
Menyimpan peran pengguna, yang dalam hal ini adalah `parent`.
8. **`uid`**
Menyimpan kode identitas unik pengguna dalam sistem autentikasi.

D Exercise

Tabel 3.5 mendeskripsikan struktur tabel `exercise`, yang digunakan untuk menyimpan data latihan soal yang diberikan kepada siswa. Informasi yang disimpan dalam tabel ini meliputi judul latihan, deskripsi, tingkat kesulitan, topik mata pelajaran, serta tautan file soal yang dapat diakses siswa. Tabel ini dimanfaatkan oleh guru untuk mengatur dan mendistribusikan soal latihan secara daring melalui aplikasi. Data yang tersimpan memungkinkan siswa untuk mengakses soal, memahami materi, dan menyelesaikannya sesuai petunjuk. Selain itu, atribut seperti kategori dan mata pelajaran membantu sistem dalam menyaring soal sesuai dengan kebutuhan pembelajaran.

Tabel 3.5. Struktur Tabel `exercise`

Nama Atribut	Tipe Data
<code>id</code>	string
<code>title</code>	string
<code>subject</code>	string
<code>grade</code>	string
<code>timestamp</code>	timestamp
<code>fileUrl</code>	string

Penjelasan Atribut Tabel `exercise`:

1. **`id`**
Menyimpan identitas unik untuk setiap latihan soal yang disimpan dalam sistem.
2. **`title`**
Menyimpan judul latihan soal yang akan ditampilkan kepada siswa.
3. **`subject`**
Menyimpan nama mata pelajaran yang berhubungan dengan latihan soal tersebut.
4. **`grade`**
Menyimpan informasi jenjang kelas yang sesuai dengan latihan soal.
5. **`timestamp`**
Menyimpan waktu saat latihan soal dibuat atau diunggah ke sistem.

6. **file_url**

Menyimpan tautan atau URL file soal yang dapat diakses atau diunduh oleh siswa.

E Notes

Table 3.6 mendeskripsikan struktur table *notes*, yang digunakan untuk menyimpan catatan atau materi pembelajaran yang dibagikan kepada siswa. Setiap data pada tabel ini berisi informasi berupa judul catatan, mata pelajaran yang terkait, jenjang kelas, waktu unggahan, serta tautan file yang dapat diakses. Tabel ini membantu guru dalam memberikan materi pembelajaran digital kepada siswa berdasarkan kelas dan mata pelajaran yang sesuai. Data dalam tabel ini akan muncul di halaman siswa untuk mempermudah akses ke informasi yang diberikan oleh guru.

Tabel 3.6. Struktur Tabel *notes*

Nama Atribut	Tipe Data
fileUrl	string
grade	string
subject	string
timestamp	timestamp
title	string

Penjelasan Atribut Tabel *notes*:

1. **fileUrl**

Menyimpan tautan menuju file catatan atau materi pembelajaran yang dapat diakses oleh siswa.

2. **grade**

Menyimpan jenjang kelas siswa yang menjadi target dari catatan tersebut.

3. **subject**

Menyimpan mata pelajaran yang terkait dengan isi dari catatan.

4. **timestamp**

Menyimpan waktu saat catatan diunggah ke dalam sistem.

5. **title**

Menyimpan judul dari catatan untuk membantu identifikasi isi materi.

F Student Files

Table 3.7 mendeskripsikan struktur tabel `studentfiles`, yang digunakan untuk menyimpan informasi file yang diunggah oleh orangtua siswa kepada guru melalui aplikasi. Informasi yang disimpan mencakup judul file, deskripsi isi, nama dan UID guru tujuan, waktu pengunggahan, serta tautan ke file tersebut. Tabel ini berfungsi untuk mempermudah pengiriman tugas, hasil kerja, atau dokumen penting dari orangtua siswa kepada guru secara langsung. Sistem akan memperlihatkan file yang relevan hanya kepada guru yang bersangkutan, sehingga komunikasi tetap terstruktur dan tepat.

Tabel 3.7. Struktur Tabel `studentfiles`

Nama Atribut	Tipe Data
<code>description</code>	string
<code>fileUrl</code>	string
<code>teacherName</code>	string
<code>teacherUid</code>	string
<code>timestamp</code>	timestamp
<code>title</code>	string

Penjelasan Atribut Tabel `studentfiles`:

- 1. `description`**
Menyimpan deskripsi singkat mengenai isi file yang diunggah.
- 2. `fileUrl`**
Menyimpan tautan menuju file yang dikirimkan oleh orangtua kepada guru.
- 3. `teacherName`**
Menyimpan nama guru tujuan yang menerima file dari orangtua.
- 4. `teacherUid`**
Menyimpan UID guru tujuan yang digunakan sistem untuk menyaring akses file.
- 5. `timestamp`**
Menyimpan waktu saat file diunggah ke sistem.
- 6. `title`**
Menyimpan judul file yang dikirimkan.

G Videos

Tabel 3.8 mendeskripsikan struktur tabel `videos`, yang digunakan untuk menyimpan informasi mengenai video pembelajaran yang dibagikan kepada siswa. Tabel ini mencatat informasi seperti judul video, mata pelajaran, jenjang kelas, durasi video, tanggal publikasi, serta tautan video yang dapat diakses oleh siswa. Tabel ini digunakan untuk menyajikan materi pembelajaran tambahan berupa video yang bisa ditonton secara mandiri oleh para siswa. Video ini mungkin berisi penjelasan tentang konsep, latihan soal, atau pembahasan materi sesuai dengan kurikulum di setiap tingkatan.

Tabel 3.8. Struktur Tabel `videos`

Nama Atribut	Tipe Data
<code>date</code>	<code>timestamp</code>
<code>duration</code>	<code>string</code>
<code>grade</code>	<code>string</code>
<code>link</code>	<code>string</code>
<code>subject</code>	<code>string</code>
<code>title</code>	<code>string</code>

Penjelasan Atribut Tabel `videos`:

1. **`date`**
Menyimpan tanggal dan waktu unggahan video ke dalam sistem.
2. **`duration`**
Menyimpan lama durasi video, biasanya dalam format menit.
3. **`grade`**
Menyimpan jenjang kelas yang menjadi target dari video pembelajaran tersebut.
4. **`link`**
Menyimpan tautan video yang dapat diakses oleh siswa.
5. **`subject`**
Menyimpan nama mata pelajaran yang terkait dengan isi video.
6. **`title`**
Menyimpan judul video pembelajaran yang ditampilkan kepada siswa.

H Assessments

Tabel 3.9 mendeskripsikan struktur tabel *assessment*, yang digunakan untuk mencatat laporan pembelajaran siswa, seperti status kehadiran, aktivitas pembelajaran, dan status asesmen. Tabel ini berisi informasi penting seperti nama siswa, jenjang kelas, sekolah, waktu kehadiran, serta deskripsi laporan yang dicatat oleh sistem. Tabel ini berperan sebagai catatan riwayat kegiatan siswa di kelas, yang dapat diakses oleh guru dan orang tua. Dengan merekam data ini secara rutin, sistem dapat menampilkan riwayat kemajuan dan aktivitas siswa dalam proses belajar.

Tabel 3.9. Struktur Tabel *assessment*

Nama Atribut	Tipe Data
date	string
description	string
grade	string
name	string
school	string
status	string
timestamp	timestamp
uid	string

Penjelasan Atribut Tabel *assessment*:

1. **date**
Menyimpan tanggal dan waktu kehadiran atau pencatatan aktivitas siswa.
2. **description**
Menyimpan deskripsi laporan pembelajaran, seperti kehadiran atau aktivitas yang dilakukan siswa.
3. **grade**
Menyimpan jenjang kelas siswa yang dilaporkan.
4. **name**
Menyimpan nama lengkap siswa.
5. **school**
Menyimpan nama sekolah tempat siswa terdaftar.

6. **status**

Menyimpan status aktivitas atau laporan, misalnya "Input Assessment".

7. **timestamp**

Menyimpan waktu sistem saat data laporan dimasukkan.

8. **uid**

Menyimpan kode identitas unik dari siswa yang bersangkutan.

I Feedback

Tabel 3.10 mendeskripsikan struktur tabel `feedback`, yang digunakan untuk menyimpan umpan balik dari siswa terhadap guru setelah sesi pembelajaran berlangsung. Setiap entri dalam tabel ini mencatat nama siswa, nama guru, isi umpan balik (`feedback`), rating penilaian, serta waktu pemberian umpan balik.

Tabel 3.10. Struktur Tabel `feedback`

Nama Atribut	Tipe Data
<code>feedback</code>	<code>string</code>
<code>rating</code>	<code>integer</code>
<code>studentName</code>	<code>string</code>
<code>teacherName</code>	<code>string</code>
<code>timestamp</code>	<code>timestamp</code>

Penjelasan Atribut Tabel `feedback`:

1. **`feedback`**

Menyimpan komentar atau pendapat siswa terhadap guru atau sesi pembelajaran.

2. **`rating`**

Menyimpan nilai penilaian dalam bentuk angka, biasanya dalam skala 1–5.

3. **`studentName`**

Menyimpan nama siswa yang memberikan umpan balik.

4. **`teacherName`**

Menyimpan nama guru yang menerima umpan balik.

5. **timestamp**

Menyimpan waktu saat umpan balik diberikan oleh siswa.

J **Online Class**

Tabel 3.11 mendeskripsikan struktur tabel `onlineclass`, yang digunakan untuk menyimpan informasi terkait jadwal kelas online yang akan diikuti oleh siswa. Data yang dicatat dalam tabel ini meliputi judul kelas, tanggal, hari, waktu pelaksanaan, serta tautan (`link`) menuju sesi online.

Tabel 3.11. Struktur Tabel `onlineclass`

Nama Atribut	Tipe Data
<code>date</code>	<code>string</code>
<code>day</code>	<code>string</code>
<code>link</code>	<code>string</code>
<code>time</code>	<code>string</code>
<code>title</code>	<code>string</code>

Penjelasan Atribut Tabel `onlineclass`:

1. **date**

Menyimpan tanggal pelaksanaan kelas online.

2. **day**

Menyimpan hari dalam format singkat, seperti `Mon`, `Tue`, `Wed`, dan seterusnya.

3. **link**

Menyimpan tautan menuju ruang kelas online (misalnya `Zoom`, `Google Meet`, atau `Canva Link`).

4. **time**

Menyimpan waktu pelaksanaan kelas dalam format jam.

5. **title**

Menyimpan judul atau nama kelas online yang akan diadakan.

3.4.6 Hasil Implementasi

Dalam proses rancang bangun aplikasi Learning Management System (LMS) PT. Matica Inovasi Edukasi, tabel yang dirancang dan diterapkan diharapkan mampu mendukung semua fungsionalitas yang diperlukan, baik untuk pengelolaan data oleh admin dan guru, maupun untuk pengguna seperti siswa dan orangtua.

A Splash Screen Onboarding

Bagian interface ini adalah tampilan splash screen dan onboarding di aplikasi LMS PT Matica Inovasi Edukasi. Splash screen muncul segera setelah aplikasi dibuka, berfungsi sebagai tampilan pertama yang mengenalkan identitas visual dari aplikasi seperti pada Gambar 3.7. Setelah itu, pengguna akan diarahkan ke halaman onboarding yang dirancang untuk memberikan ringkasan tentang fitur utama dan keuntungan dari aplikasi. Onboarding bertujuan untuk membantu pengguna baru memahami secara keseluruhan fungsi aplikasi sebelum memasuki halaman login.



Gambar 3.7. Tampilan Splash Screen

Setelah aplikasi dijalankan, pengguna akan diarahkan ke halaman splash screen terlebih dahulu. Jika ini adalah pertama kalinya aplikasi dibuka, maka sistem akan otomatis menampilkan halaman onboarding pertama seperti pada Gambar 3.8 sebagai pengantar sebelum masuk ke proses login.



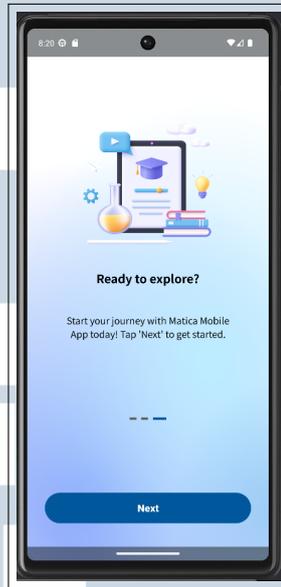
Gambar 3.8. Tampilan Onboarding 1

Setelah pengguna menyelesaikan halaman onboarding pertama, aplikasi akan secara otomatis menavigasi ke halaman onboarding kedua ditunjukkan pada Gambar 3.9 untuk melanjutkan penjelasan fitur berikutnya.



Gambar 3.9. Tampilan Onboarding 2

Setelah halaman onboarding kedua ditampilkan dan dilanjutkan, pengguna akan diarahkan ke halaman onboarding ketiga sebagai bagian akhir dari rangkaian onboarding aplikasi seperti pada Gambar 3.10.



Gambar 3.10. Tampilan Onboarding 3

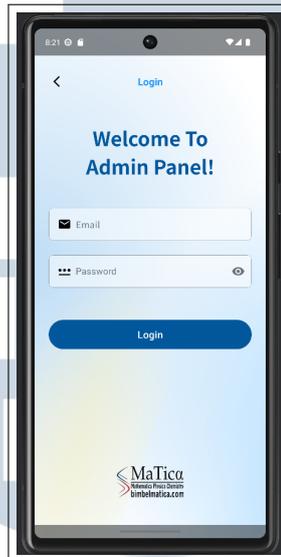
Setelah menyelesaikan halaman onboarding ketiga, user dapat menekan tombol Next untuk menuju halaman Choose Role, dapat dilihat pada Gambar 3.11, di mana user dapat memilih role sebagai admin, student, teacher, atau parent sesuai kebutuhan.



Gambar 3.11. Tampilan Pilih Role

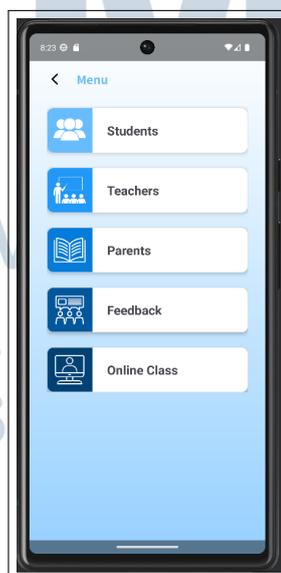
B Login dan Dashboard Admin

Bagian interface ini adalah halaman login untuk mengakses LMS yang digunakan oleh pengelola aplikasi LMS PT Matica Inovasi Edukasi. Pada Gambar 3.12, admin diminta untuk memasukkan username dan password sebagai langkah autentikasi sebelum dapat mengakses sistem.



Gambar 3.12. Tampilan Login Admin

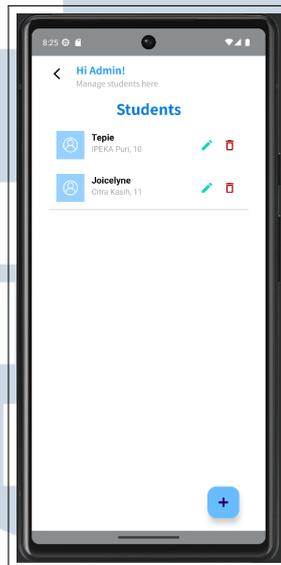
Halaman dashboard admin menampilkan ringkasan data berupa jumlah siswa, jumlah guru, dan jumlah mata pelajaran. Selain itu, terdapat tombol di bawah kanan untuk melihat menu seperti pada Gambar 3.13.



Gambar 3.13. Tampilan Menu Admin

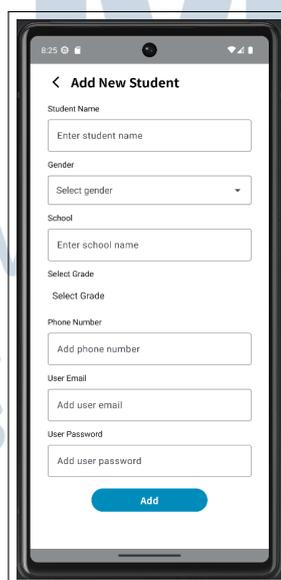
C Manajemen Akun Siswa

Pada halaman *Student List*, admin dapat melihat daftar seluruh siswa yang terdaftar dalam sistem. Setiap entri siswa dilengkapi dengan tombol Edit untuk memperbarui data dan tombol Delete untuk menghapus data siswa dari sistem yang ditunjukkan pada Gambar 3.14.



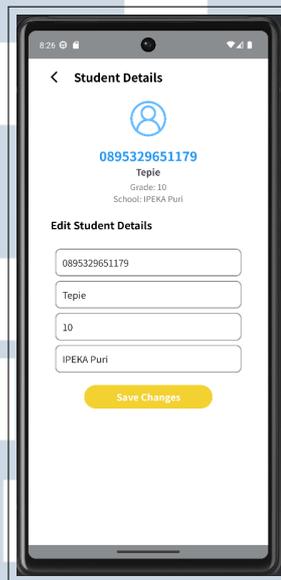
Gambar 3.14. Tampilan Daftar Siswa

Pada halaman *Student List*, admin dapat menambahkan siswa baru dengan mengisi data seperti nama, gender, sekolah, grade, nomor telepon, email, dan password (Gambar 3.15).



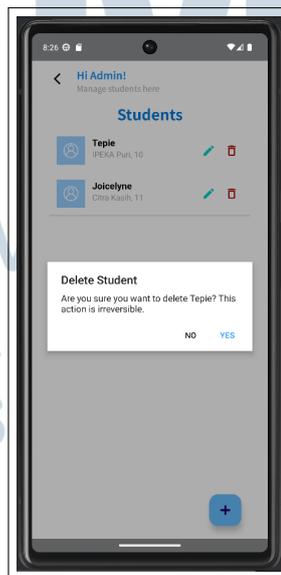
Gambar 3.15. Tampilan Tambah Siswa

Selain menambahkan data, admin juga dapat mengelola data siswa yang sudah ada melalui fitur Edit Student Details. Fitur ini memungkinkan admin untuk memperbarui informasi siswa secara langsung pada halaman detail siswa yang dipilih, seperti yang ditunjukkan pada Gambar 3.16.



Gambar 3.16. Tampilan Edit Info Siswa

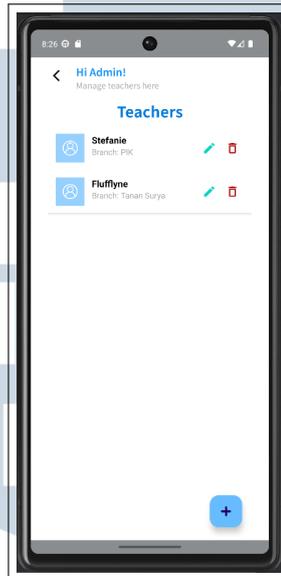
Admin juga dapat menghapus data siswa melalui tombol Delete yang tersedia pada daftar siswa. Saat tombol ditekan, akan muncul kotak konfirmasi untuk memastikan tindakan penghapusan sebelum data benar-benar dihapus dari sistem, sebagaimana terlihat pada Gambar 3.17.



Gambar 3.17. Tampilan Hapus Siswa

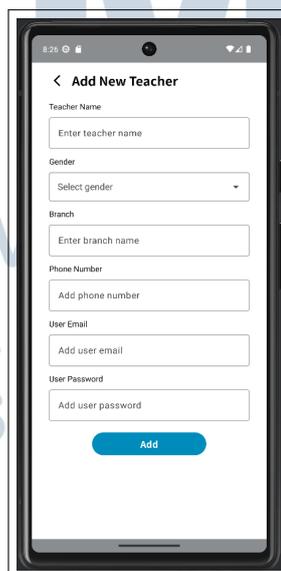
D Manajemen Akun Guru

Selain itu, tersedia halaman Teacher List yang menampilkan daftar seluruh pengguna dengan peran sebagai guru. Pada halaman ini, admin dapat melihat, menambahkan, mengedit, maupun menghapus data guru yang terdaftar dalam sistem, seperti yang ditunjukkan pada Gambar 3.18.



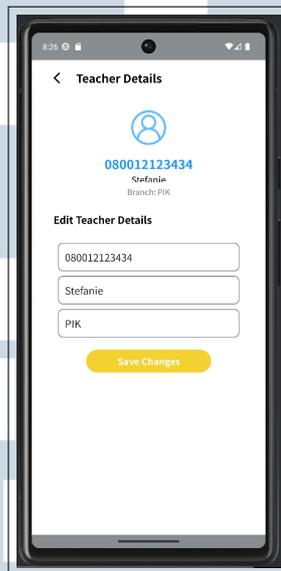
Gambar 3.18. Tampilan Daftar Guru

Admin dapat menambahkan guru baru melalui tombol Add New Teacher, lalu mengisi formulir data seperti nama, gender, cabang, email, telepon, dan password (Gambar 3.19).



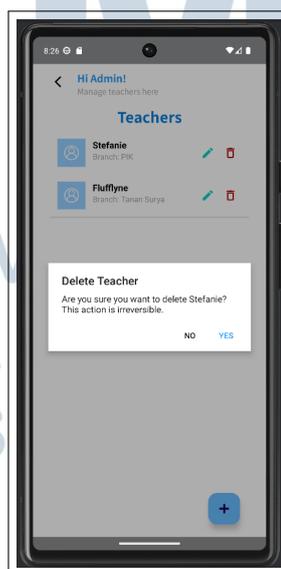
Gambar 3.19. Tampilan Tambah Guru

Selain menambahkan data, admin juga dapat memperbarui informasi guru melalui fitur Edit Teacher Details. Fitur ini memungkinkan admin untuk mengakses dan mengedit data guru yang telah terdaftar sesuai kebutuhan, sebagaimana ditunjukkan pada Gambar 3.20.



Gambar 3.20. Tampilan Edit Data Guru

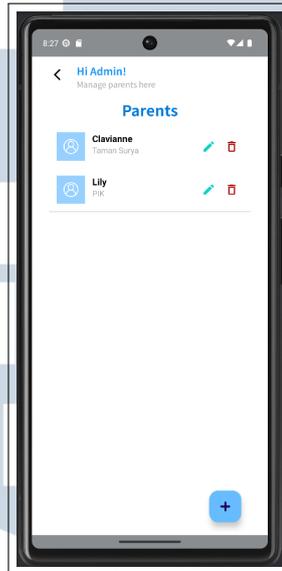
Admin juga memiliki opsi untuk menghapus data guru melalui fitur Delete Teacher. Saat tombol delete ditekan, sistem akan menampilkan kotak konfirmasi untuk memastikan tindakan penghapusan sebelum data guru dihapus secara permanen dari sistem, seperti terlihat pada Gambar 3.21.



Gambar 3.21. Tampilan Hapus Data Guru

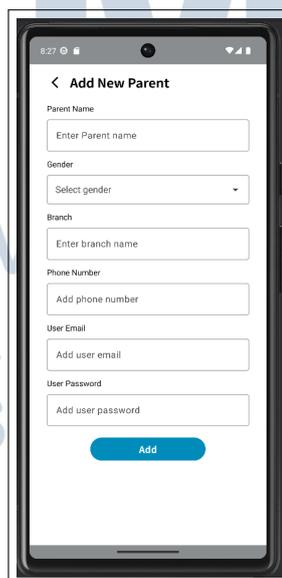
E Manajemen Akun Orangtua

Selain halaman Guru, tersedia juga halaman Parent List yang menampilkan daftar pengguna dengan peran sebagai orangtua. Halaman ini memungkinkan admin untuk melihat dan mengelola data orangtua yang terdaftar dalam sistem, seperti pada Gambar 3.22.



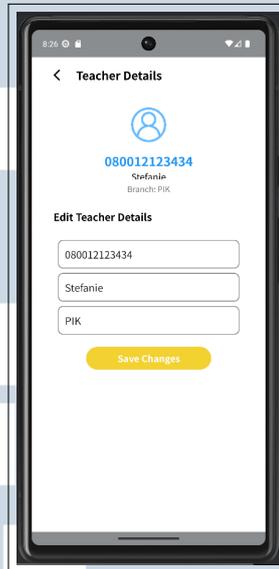
Gambar 3.22. Tampilan Daftar Orang Tua

Admin dapat menambahkan orangtua baru melalui tombol Add New Parent, lalu mengisi formulir data seperti nama, gender, cabang, email, telepon, dan password (Gambar 3.23).



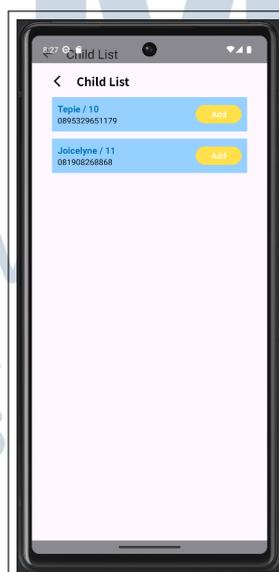
Gambar 3.23. Tampilan Tambah Orang Tua

Selain menambahkan data, admin juga dapat memperbarui informasi orangtua melalui fitur Edit Parent Details, yang memungkinkan pengeditan data secara langsung. Ilustrasi fitur ini ditunjukkan pada Gambar 3.24.



Gambar 3.24. Tampilan Edit Data Orang Tua

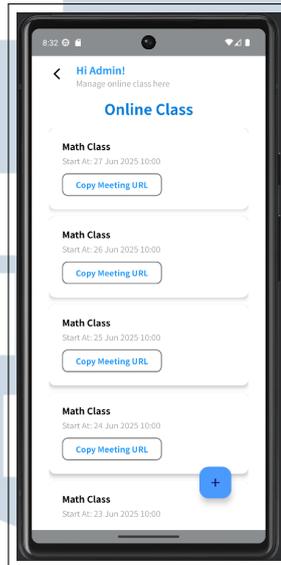
Selanjutnya, terdapat halaman Child List yang memungkinkan admin untuk menetapkan atau assign anak ke akun orangtua. Melalui halaman ini, admin dapat memilih anak yang ingin dihubungkan, kemudian memilih akun orangtua yang sesuai untuk membentuk relasi antara keduanya dalam sistem, seperti yang ditunjukkan pada Gambar 3.25.



Gambar 3.25. Tampilan Daftar Anak

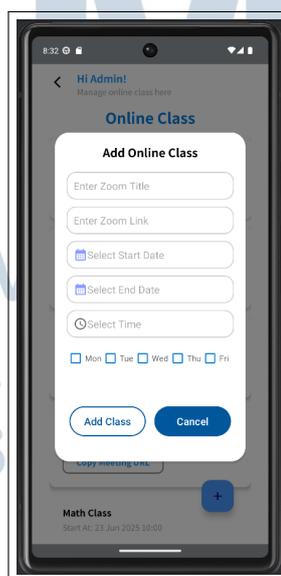
F Manajemen Kelas Daring dan Feedback

Terdapat juga halaman Online Class List yang menampilkan daftar sesi kelas online yang telah dijadwalkan. Halaman ini memungkinkan admin untuk memantau informasi seperti judul kelas, tanggal, hari, waktu, dan tautan menuju sesi kelas daring yang tersedia, sebagaimana terlihat pada Gambar 3.26.



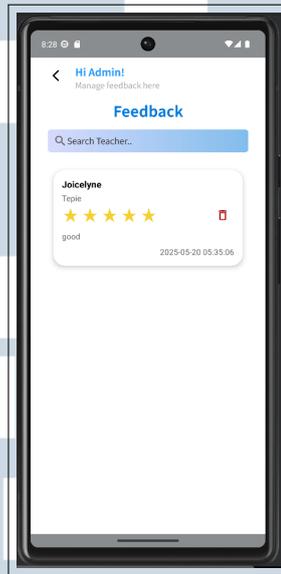
Gambar 3.26. Tampilan Daftar Kelas Daring

Admin dapat menambahkan sesi kelas online baru melalui tombol Add Online Class, lalu mengisi detail seperti judul, tanggal, hari, waktu, dan link (Gambar 3.27).



Gambar 3.27. Tampilan Tambah Kelas Daring

Selain itu, tersedia halaman Feedback List yang menampilkan kumpulan umpan balik dari siswa terhadap guru. Melalui halaman ini, admin dapat melihat komentar, nama siswa, nama guru, serta rating yang diberikan sebagai bahan evaluasi kualitas pembelajaran. Tampilan fitur ini ditunjukkan pada Gambar 3.28.



Gambar 3.28. Tampilan Feedback

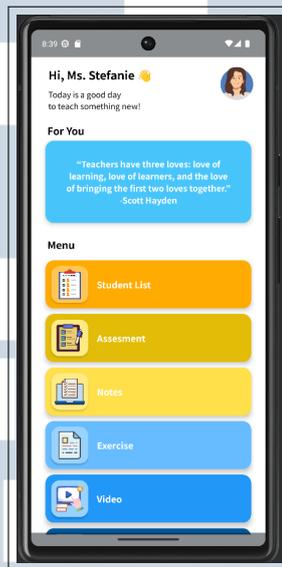
G Login dan Dashboard Guru

Guru dapat login dengan email dan password terdaftar, lalu diarahkan ke dashboard utama untuk mengakses fitur pembelajaran (Gambar 3.29).



Gambar 3.29. Tampilan Login Guru

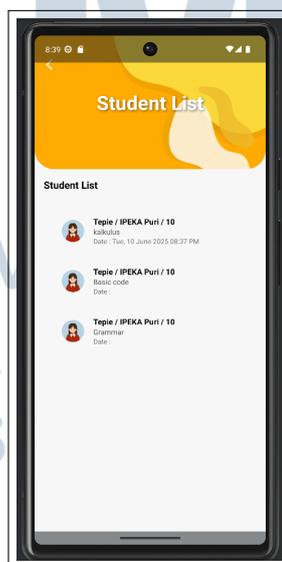
Setelah berhasil login, guru akan diarahkan ke halaman dashboard yang menampilkan menu utama seperti *Student List*, *Exercise*, *Notes*, *Video* dan *Student's Files*, yang mempermudah akses ke fitur-fitur pembelajaran yang tersedia, sebagaimana terlihat pada Gambar 3.30.



Gambar 3.30. Tampilan Dashboard Guru

H Student List

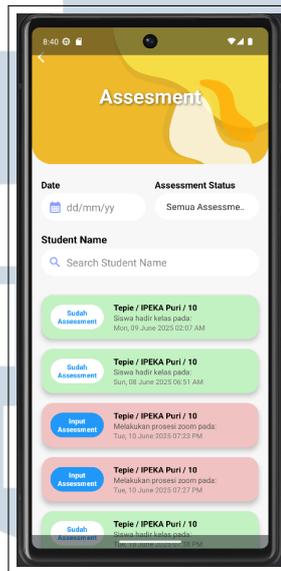
Guru dapat membuka halaman Student List untuk melihat siswa yang terhubung dan memantau perkembangan mereka (Gambar 3.31).



Gambar 3.31. Tampilan Daftar Siswa di Halaman Guru

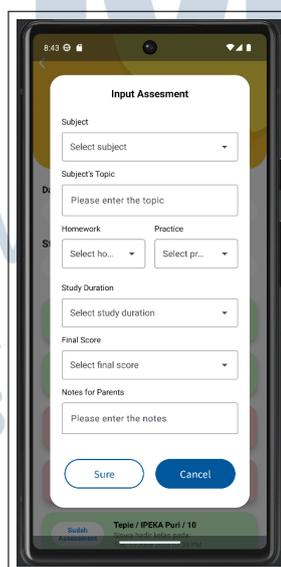
I Assessment

Guru juga dapat mengakses halaman Assessment List untuk melihat status asesmen dari masing-masing siswa. Informasi seperti tanggal, deskripsi, dan status asesmen ditampilkan secara jelas dan terstruktur, seperti yang dapat dilihat pada gambar 3.32.



Gambar 3.32. Tampilan Daftar Assessment

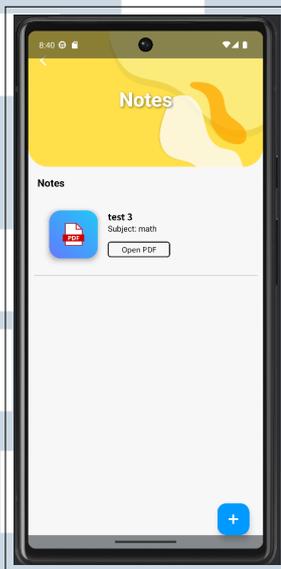
Guru juga dapat melakukan input asesmen melalui form khusus yang tersedia pada sistem, yang ditampilkan pada Gambar 3.33. Form ini mencakup informasi seperti mata pelajaran, topik, nilai akhir, durasi belajar, serta status tugas dan latihan siswa.



Gambar 3.33. Tampilan Input Assessment

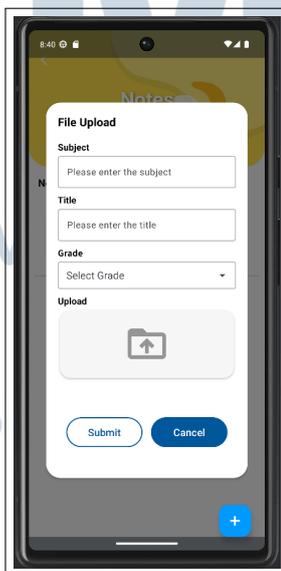
J Catatan (Notes)

Melalui fitur Notes, guru dapat melihat daftar catatan yang telah diunggah sebelumnya, seperti yang diperlihatkan pada Gambar 3.34, lengkap dengan informasi judul, mata pelajaran, jenjang kelas, dan waktu unggah.



Gambar 3.34. Tampilan Daftar Catatan (Notes)

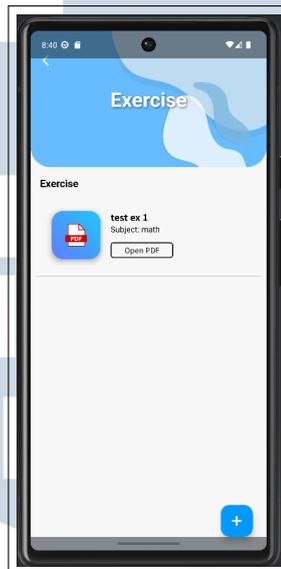
Guru juga memiliki akses untuk mengunggah catatan baru melalui fitur Notes. Proses ini ditampilkan pada Gambar 3.35, di mana guru dapat mengisi form berisi judul, mata pelajaran, jenjang kelas, dan mengunggah file catatan untuk dibagikan kepada siswa.



Gambar 3.35. Tampilan Unggah Catatan (Notes)

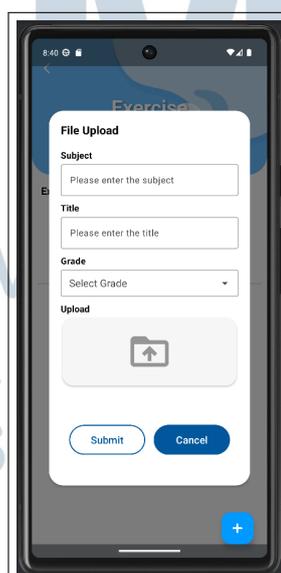
K Latihan Soal (Exercise)

Guru juga dapat mengakses daftar latihan soal (Exercise List) yang telah dibuat atau dibagikan sebelumnya. Melalui halaman ini, seperti yang ditunjukkan pada Gambar 3.36, guru dapat memantau judul latihan, mata pelajaran, jenjang kelas, dan tanggal unggah dengan mudah.



Gambar 3.36. Tampilan Daftar Latihan Soal (Exercise)

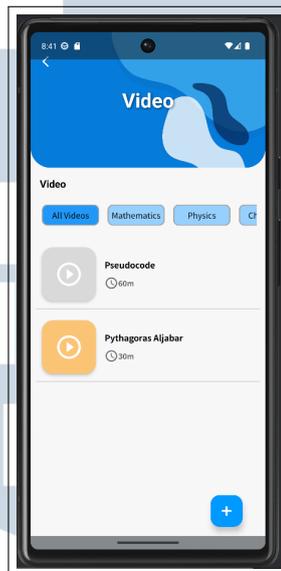
Guru juga dapat mengunggah latihan soal melalui form berisi judul, mata pelajaran, jenjang kelas, dan tautan file latihan, seperti ditampilkan pada Gambar 3.37.



Gambar 3.37. Tampilan Unggah Latihan Soal (Exercise)

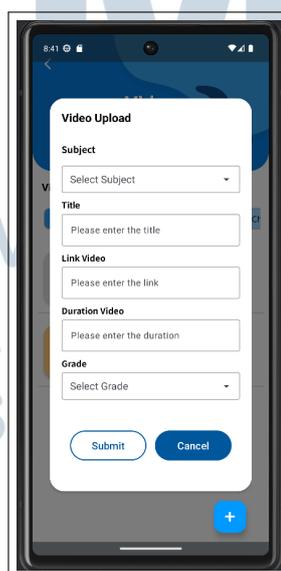
L Video

Guru juga dapat mengakses daftar video pembelajaran melalui fitur Videos. Pada halaman ini, guru dapat melihat video yang telah dibagikan beserta detail seperti judul, mata pelajaran, jenjang kelas, durasi, dan tanggal tayang, sebagaimana terlihat pada Gambar 3.38.



Gambar 3.38. Tampilan Daftar Video

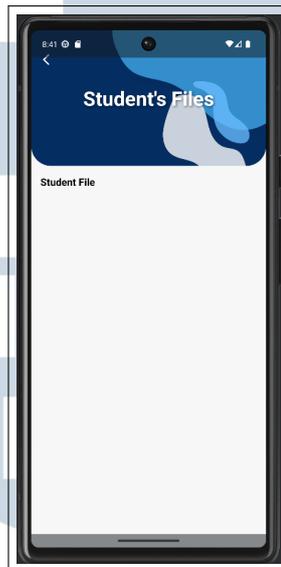
Guru juga dapat mengunggah tautan video pembelajaran dengan mengisi form yang mencakup judul, mata pelajaran, jenjang kelas, durasi, dan tanggal, seperti ditunjukkan pada Gambar 3.39.



Gambar 3.39. Tampilan Unggah Video

M File Siswa (Student's Files)

Guru juga dapat mengakses halaman *Student's Files*, yaitu daftar file yang dikirim oleh orang tua murid. Melalui halaman ini, seperti yang ditampilkan pada Gambar 3.40, guru dapat melihat nama file, deskripsi, serta informasi pengirim untuk setiap file yang diterima.



Gambar 3.40. Tampilan Daftar File Siswa

3.5 Pengujian Sistem

Pengujian sistem dilaksanakan dengan metode Black Box testing untuk menjamin bahwa semua fitur dan fungsi aplikasi berjalan dengan baik, sesuai dengan yang diinginkan. Metode ini berorientasi pada evaluasi antarmuka, pengguna dan hubungan antara sistem dengan pengguna tanpa melihat kode internal [18]. Pengujian Aplikasi terkait dilakukan oleh Supervisor Legi Hendra. Tabel 3.12 menyajikan hasil dari pengujian sistem yang telah dilakukan, dengan mencakup Setiap skenario yang diujikan, hasil yang diinginkan, dan hasil pengujian yang sebenarnya, serta kesimpulan dari setiap uji yang telah dilakukan.

Tabel 3.12. Hasil Pengujian Sistem

No.	Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Menjalankan aplikasi	Aplikasi menampilkan splash screen secara otomatis	Splash screen muncul selama beberapa detik sebelum berpindah ke onboarding	Sesuai
2	Menampilkan onboarding saat pertama kali membuka aplikasi	Onboarding muncul secara berurutan mulai dari halaman onboarding_1 hingga onboarding_3	Onboarding ditampilkan dengan urutan dan tampilan sesuai desain	Sesuai
3	Navigasi dari onboarding_3 ke halaman Choose Role	Setelah menekan tombol Next pada onboarding_3, pengguna diarahkan ke halaman Choose Role	Halaman Choose Role berhasil ditampilkan setelah menekan tombol Next	Sesuai
4	Pemilihan peran admin	Saat memilih peran sebagai admin, sistem menampilkan halaman login admin	Halaman login admin muncul setelah memilih role admin	Sesuai
5	Login admin dengan kredensial valid	Admin berhasil login dan diarahkan ke dashboard admin	Sistem menampilkan dashboard admin dengan informasi ringkasan	Sesuai
6	Navigasi dari dashboard ke halaman Student List	Tombol atau sidebar mengarahkan ke halaman daftar siswa	Halaman Student List berhasil ditampilkan	Sesuai
Lanjut pada halaman berikutnya				

Lanjutan Tabel 3.12

No.	Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
7	Menambahkan data siswa baru	Formulir Add New Student muncul dan data dapat disimpan	Siswa berhasil ditambahkan dan muncul di daftar siswa	Sesuai
8	Mengedit data siswa	Admin dapat memperbarui data siswa	Data siswa berhasil diperbarui dan tersimpan dengan benar	Sesuai
9	Menghapus data siswa	Admin dapat menghapus siswa dengan konfirmasi terlebih dahulu	Data siswa terhapus setelah konfirmasi dilakukan	Sesuai
10	Menampilkan daftar guru	Halaman Teacher List muncul dan menampilkan data guru yang terdaftar	Daftar guru berhasil ditampilkan	Sesuai
11	Menambahkan data guru baru	Formulir Add New Teacher muncul dan data dapat disimpan	Guru berhasil ditambahkan ke sistem	Sesuai
12	Mengedit data guru	Admin dapat mengubah informasi guru	Data guru berhasil diperbarui dengan benar	Sesuai
13	Menghapus data guru	Admin dapat menghapus data guru dengan konfirmasi	Guru berhasil dihapus dari daftar	Sesuai
14	Menampilkan daftar orangtua	Halaman Parent List muncul dan menampilkan data pengguna dengan peran orangtua	Daftar orangtua berhasil ditampilkan	Sesuai
Lanjut pada halaman berikutnya				

Lanjutan Tabel 3.12

No.	Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
15	Menambahkan data orangtua baru	Admin dapat mengisi formulir Add New Parent dan menyimpannya	Orangtua berhasil ditambahkan dan tampil dalam daftar	Sesuai
16	Mengedit data orangtua	Admin dapat memperbarui informasi orangtua	Perubahan disimpan dan ditampilkan dengan benar	Sesuai
17	Assign anak ke akun orangtua	Admin dapat memilih anak dan orangtua untuk dihubungkan	Anak berhasil di-assign ke akun orangtua	Sesuai
18	Login guru dengan kredensial valid	Guru dapat login dan diarahkan ke dashboard guru	Dashboard guru ditampilkan setelah login berhasil	Sesuai
19	Menampilkan daftar siswa guru	Halaman Student List di sisi guru menampilkan siswa yang diasuh	Daftar siswa tampil dengan benar	Sesuai
20	Input asesmen siswa oleh guru	Guru dapat mengisi form asesmen untuk siswa	Asesmen berhasil dikirim dan tersimpan ke sistem	Sesuai
21	Melihat catatan (notes) yang telah diunggah	Halaman Notes menampilkan daftar catatan guru	Daftar catatan tampil sesuai data yang diunggah	Sesuai
22	Upload catatan baru oleh guru	Guru dapat mengisi form dan mengunggah file catatan	File catatan berhasil diunggah dan muncul di daftar	Sesuai
23	Menampilkan daftar exercise	Guru dapat melihat latihan soal yang telah dibuat	Halaman Exercise List tampil sesuai data latihan soal	Sesuai

Lanjut pada halaman berikutnya

Lanjutan Tabel 3.12

No.	Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
24	Upload latihan soal baru oleh guru	Guru dapat mengisi form latihan dan mengunggahnya	Latihan soal berhasil diunggah dan muncul di sistem	Sesuai
25	Menampilkan daftar video pembelajaran	Halaman Video List menampilkan video yang tersedia	Daftar video tampil dengan detail lengkap	Sesuai
26	Upload tautan video oleh guru	Guru dapat mengisi form dan menyimpan link video pembelajaran	Video berhasil disimpan dan muncul di daftar	Sesuai
27	Akses Student's Files oleh guru	Guru dapat melihat file yang dikirim oleh orangtua	File tampil lengkap dan dapat diakses guru	Sesuai

3.6 Kendala dan Solusi yang Ditemukan

Selama periode kerja magang di PT Matica Inovasi Edukasi, sejumlah kendala telah diidentifikasi, baik pada sisi frontend maupun backend. Kendala-kendala menjadi sebuah tantangan yang perlu dihadapi agar proyek dapat berjalan dengan sesuai dengan strategi dan memenuhi harapan perusahaan.

3.6.1 Kendala

Beberapa kendala yang ditemukan selama pelaksanaan kerja magang adalah sebagai berikut:

1. Keterbatasan penggunaan layanan Firebase karena adanya pembatasan kuota dan biaya mengharuskan penggunaan Supabase sebagai storage alternatif. Hal ini menimbulkan tantangan tersendiri karena dokumentasi dan referensi komunitas Supabase tidak sebanyak Firebase, sehingga memperlambat proses integrasi dan debugging.
2. Kurangnya pengalaman tim dalam penggunaan Supabase berdampak pada lambatnya penanganan error dan keterbatasan optimalisasi fitur. Beberapa

fitur seperti preview file, pengaturan akses publik, dan pemantauan upload memerlukan eksplorasi manual dan tidak selalu terdokumentasi dengan jelas.

3.6.2 Solusi

Untuk menyelesaikan kendala-kendala yang ada, beberapa solusi yang diterapkan antara lain:

1. Tim melakukan eksplorasi lebih dalam terhadap dokumentasi resmi Supabase dan memanfaatkan forum komunitas seperti GitHub Discussions dan Stack Overflow. Selain itu, dilakukan pencatatan proses implementasi sebagai dokumentasi internal untuk mempercepat troubleshooting di masa mendatang.
2. Dilakukan pembagian tugas eksplorasi Supabase secara spesifik per fitur di antara anggota tim, serta dilakukan uji coba fitur secara modular. Tim juga membuat prototipe kecil (sandbox) untuk menguji fitur tertentu sebelum diterapkan ke sistem utama guna meminimalisir risiko kesalahan.

